



دانشگاه کاشان
University of Kashan

مجله محاسبات نرم

SOFT COMPUTING JOURNAL

تارنمای مجله: scj.kashanu.ac.ir



تجدید آرایش بهینه شبکه‌های توزیع به منظور کاهش تلفات و افزایش قابلیت اطمینان با استفاده از الگوریتم خفاش

سید عباس طاهر^{۱*}، استاد، سعید فاطمی^۲، دانشجوی دکتری، امید هنرفر^۳، دانشجوی دکتری
^{۱،۲،۳} دانشکده مهندسی برق و کامپیوتر، دانشگاه کاشان، کاشان، ایران.

چکیده

بازآرایی یکی از ساده‌ترین و کم‌هزینه‌ترین روش‌هایی است که به وسیله آن می‌توان بدون افزودن تجهیزات اضافی در شبکه، اهدافی مانند خودترمیمی، افزایش قابلیت اطمینان، کاهش تلفات و... را اجرایی کرد. با توجه به گسترش شبکه‌های توزیع، پیچیده‌تر شدن ارتباطات و افزایش پارامترها، و نیز با توجه به اینکه مدل‌سازی مسئله بازآرایی به وسیله معادلات ریاضی امری ناممکن است، استفاده از تکنیک‌های هوشمند در انجام بازآرایی اجتناب‌ناپذیر است. از سوی دیگر، تحلیل قطع یا وصل بودن هر سوئیچ می‌بایست بدون خطا بوده و متناسب با محدودیت‌های شبکه، همچون جریان قابل تحمل هر شاخه، ولتاژ مجاز شین‌ها و توان عبوری از خطوط انجام شود. با توجه به این نیاز است در بین الگوریتم‌های هوشمند، از الگوریتمی استفاده شود که از نظر فاکتورهای مختلف از قبیل دقت، سرعت همگرایی بتواند عملکردی بهینه داشته باشد. در این مطالعه، عملکرد الگوریتم‌های باینری خفاش، سنجاقک، ژنتیک، بهینه‌سازی ازدحام ذرات و فاخته در شبکه‌های توزیع شعاعی استاندارد ۱۶ و ۳۳ شینه و با هدف بازآرایی شبکه برای رسیدن به توپولوژی‌ای با کمترین تلفات و بیشترین قابلیت اطمینان با یکدیگر مقایسه شده است. نتایج حاصل برای این مسئله نشان می‌دهد که عملکرد الگوریتم باینری خفاش از نظر سرعت همگرایی نسبت به سایر الگوریتم‌ها مندرج در این مقاله بهتر است.

© ۱۳۹۹ - مجله محاسبات نرم، کليه حقوق محفوظ است.

اطلاعات مقاله

تاریخچه مقاله:

دریافت ۰۷ دی ماه ۱۳۹۸

پذیرش ۰۵ تیر ماه ۱۳۹۹

کلمات کلیدی:

الگوریتم‌های هوشمند

بازآرایی شبکه توزیع

قابلیت اطمینان

۱. مقدمه

با توجه به افزایش سریع تقاضای برق، محدودیت‌های زیست‌محیطی و بازار رقابتی برق، سیستم‌های انتقال و توزیع اغلب تحت شرایطی نزدیک شرایط نامی بارگذاری می‌شوند.

به تازگی ارزیابی کفایت سیستم‌های تولید و قابلیت اطمینان سیستم‌های انتقال از مهم‌ترین مطالعات برنامه‌ریزی سیستم محسوب می‌شود [۱]. در مجموع بیش از ۴۰٪ کل سرمایه‌گذاری انجام‌شده در شبکه برق، به بخش توزیع اختصاص دارد. این امر نشان‌دهنده اهمیت اقتصادی سیستم توزیع بوده و برای استفاده مناسب از آن نیازمند طراحی، ساخت، تعمیر، نگهداری و برنامه‌ریزی درست و دقیق است. به‌طور خلاصه، اغلب وقفه‌های ایجادشده در برق‌رسانی به مشترکان، ناشی از اختلالات شبکه‌های توزیع است [۲]. با توجه به این موضوع،

✦ نوع مقاله: پژوهشی

* نویسنده مسئول

پست‌های الکترونیک: sataher@kashanu.ac.ir (طاهر)

saeid.fatemi@grad.kashanu.ac.ir (فاطمی)

omid.honarfar@grad.kashanu.ac.ir (هنرفر)

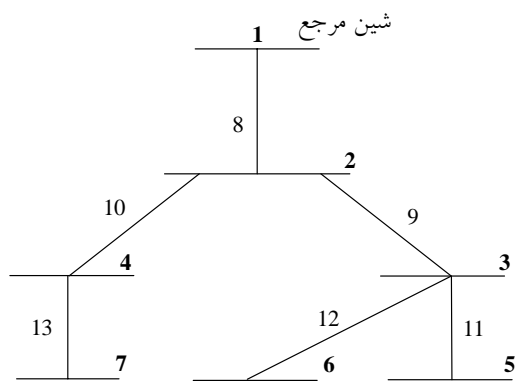
نحوه ارجاع به مقاله: طاهر، سید عباس، فاطمی، سعید، هنرفر، امید، «تجدید آرایش بهینه شبکه‌های توزیع به منظور کاهش تلفات و افزایش قابلیت اطمینان با استفاده از الگوریتم خفاش»، مجله محاسبات نرم، جلد ۸، شماره ۱، ص ۲۹-۴۲، بهار و تابستان ۱۳۹۸.

است که در آن سوئیچ باز، بسته و سوئیچ بسته، باز می‌شود تا شبکه ساختار شعاعی خود را حفظ کند. عیب این روش آن است که حالت‌های کمی را می‌تواند در نظر بگیرد. در [۶] روش‌هایی بر مبنای تعویض شاخه‌ها استفاده شده است که به‌عنوان روش‌های کلاسیک شناخته می‌شود. اما از آنجایی که شبکه توزیع شامل صدها سوئیچ است، در نظر گرفتن تمامی آرایش‌های موجود و امکان بررسی تمامی آن‌ها توسط روش‌های کلاسیک میسر نیست. لذا رفته‌رفته روش‌های هوشمند به دلیل دوری از پاسخ‌های محلی و در نظر گرفتن بهتر هدف‌های چندگانه مورد توجه قرار گرفتند. در [۷] یک روش متا اکتشافی^۱ مبتنی بر الگوریتم شیر مورچه^۲ برای یافتن تنظیمات بهینه شبکه توزیع ارائه شده است؛ این کار با تنظیم وضعیت سوئیچ‌ها انجام می‌شود. در [۸] تمرکز بر یافتن بهترین پیکربندی شبکه سیستم توزیع با تغییر تای^۳ کلید و سکشن‌لایزر توأم با شرط شعاعی ماندن شبکه است. هدف این مقاله بهبود پروفیل ولتاژ و به حداقل رساندن تلفات سیستم است. بهینه‌سازی سیستم با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات (PSO) برای یافتن بهترین الگوی سوئیچینگ انجام می‌شود. هدف در مقاله [۹]، کاهش تلفات در سیستم توزیع برای فیدرهای باراتا و فیدرهای Sulawesi با تنظیم ۱۰ سوئیچ با استفاده از الگوریتم مورچه‌ها اعمال شده است. در [۱۱] از الگوریتم اژدها برای حل مسائل تک و چند هدفی گسسته استفاده شده است. در [۱۲] از الگوریتم فاخته برای انجام محاسبات نرم استفاده شده است. به‌طور کلی، روش‌های مبتنی بر هوش مصنوعی مانند متغیرهای فازی [۱۳]، الگوریتم ژنتیک [۱۴]، الگوریتم جست‌وجوی کلونی مورچه‌ها [۱۵] و دیگر تکنیک‌های تکاملی به‌طور فزاینده‌ای برای مسئله بازآرایی سیستم توزیع استفاده می‌شوند. الگوریتم بهینه‌سازی خفاش، یک الگوریتم جدید الهام‌گرفته از رفتار خفاش برای یافتن غذاست. خفاش‌ها تنها پستانداران بال‌دار هستند که برای شکار طعمه از انعکاس صدا استفاده می‌کنند. الگوریتم خفاش یکی از

ارزیابی دقیق قابلیت اطمینان سیستم‌های توزیع در مراحل بهره‌برداری و برنامه‌ریزی یک سیستم قدرت اهمیت ویژه‌ای پیدا کرده است. معمولاً بازآرایی سیستم توزیع با اهداف کاهش تلفات، کاهش افت ولتاژ و افزایش قابلیت اطمینان انجام می‌شود [۳]. یک معیار مهم برای بازآرایی شبکه، شعاعی باقی ماندن یک شبکه شعاعی پس از بازآرایی است؛ که دلیل اصلی آن سهولت در هماهنگی رله‌های حفاظتی است. همچنین از دید بازایی شبکه، بازآرایی، مسیر مناسب تغذیه بارهای مختلف را چنان تعیین می‌کند که حداکثر بارهای شبکه با کمترین کلیدزنی، برقرار شود. با توجه به این، بازآرایی سیستم توزیع یک مسئله بهینه‌سازی است که توسط آن بهترین ترکیب ممکن از وضعیت کلیدها یافت می‌شود. روند حل مسئله بدین گونه است که یک تابع هدف (مانند تلفات شبکه) بهینه می‌شود. فرایند بهینه‌سازی محدودیت‌های مختلفی مانند حداکثر افت ولتاژ مجاز در خط، جریان مجاز خطوط، ظرفیت ترانسفورماتور و هر گونه محدودیتی را که در فازهای برنامه‌ریزی یا بهره‌برداری از شبکه وجود دارد، در نظر می‌گیرد. مفهوم بازآرایی شبکه توزیع به منظور کاهش تلفات ابتدا در سال ۱۹۷۵ توسط Merlin و Back معرفی شد. در این روش، ابتدا تمام کلیدهای شبکه بسته می‌شود و هر دفعه یک کلید باز می‌شود تا یک آرایش شعاعی جدید حاصل شود. در این فرایند، وضعیت کلیدها به‌گونه‌ای انتخاب می‌شود که تلفات خطوط در آرایش جدید کمینه شود. مزیت این ایده آن است که آرایش نهایی شبکه، مستقل از حالت اولیه کلیدها تعیین می‌شود، اما از طرفی همگرایی الگوریتم، تضمین نشده و به محاسبات بسیار زیادی نیاز دارد [۳ و ۴]. اکثر مسائل بازآرایی موجود در شبکه توزیع، قابلیت اطمینان را در نظر نمی‌گیرند چون با اضافه شدن قابلیت اطمینان به تابع هدف، مسئله بهینه‌سازی مشکل‌تر خواهد شد و لازم است تا با استفاده از الگوریتم‌های دقیق حل شوند. از سوی دیگر در شبکه‌های هوشمند عملکرد الگوریتم استفاده‌شده باید به‌نحوی باشد که زمان حل مسئله کم شده، کیفیت حل افزایش یافته و نیز از پاسخ‌های محلی اجتناب شود. در مراجع [۵] از تکنیک‌های مبتنی بر تغییر شاخه به‌منظور بازآرایی بهینه شبکه استفاده شده

1. Meta heuristic
2. Ant lion
3. Tie

PC از شبکه ۷ شینه نشان داده شده در شکل (۱) استفاده می‌شود. ماتریس PC این شبکه در شکل (۲) نشان داده شده است. سطر و ستون‌های این ماتریس نماینده شین‌های شبکه است. برای هر یک از شین‌های شبکه، ماتریس PC تعیین کننده دو مجموعه از شین‌های بالادستی و پایین دستی است. مجموعه شین‌های بالادستی شامل شین‌هایی است که شین مورد نظر را به شین مرجع متصل می‌کند. مجموعه شین‌های پایین دستی شامل شین‌هایی می‌شود که از طریق این شین تغذیه می‌شوند. برای یک شین مشخص، شین‌های بالادستی و پایین دستی شامل درایه‌های یک شده به ترتیب در ستون و سطر متناظر با آن شین در ماتریس PC است. برای مثال در شکل (۱) مجموعه شین‌های بالادستی برای شین ۳ عبارت است از {۱ و ۲ و ۳}. برای بررسی صحت ماتریس PC، همان طور که در شکل (۲) مشخص شده است، ستون متناظر شین ۳ سه درایه یک شده دارد که این سه درایه متناظر با شین‌های ۶، ۵ و نیز خود شین ۳ هستند. مجموعه شین‌های پایین دستی نیز به همین ترتیب با استفاده از سطر متناظر با شین ۷ تعیین می‌شود.



شکل (۱): شبکه ۷ شینه

شین بالا دستی

$$PC_7 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{matrix}$$

شکل (۲): ماتریس PC شبکه ۷ شینه

الگوریتم‌های فراابتکاری است که بر اساس اصل انعکاس صدای خفاش و دریافت صدا توسط این پرنده عمل می‌کند. این الگوریتم، الهام گرفته از صدای خفاش و جزء اولین الگوریتم‌هایی است که از تنظیم فرکانس استفاده می‌کنند. الگوریتم خفاش را یانگ در سال ۲۰۱۰ ارائه کرد [۱۶]. تاکنون الگوریتم خفاش برای مسائل بهینه‌سازی دودویی [۱۷] و برای مسائل بهینه‌سازی چندهدفی [۱۸] به کار گرفته شده است. با توجه به این، هدف این مقاله عبارت است از:

- بررسی عملکرد الگوریتم خفاش در بین چند الگوریتم شناخته شده دیگر و مقایسه سرعت و دقت همگرایی آن در یافتن پاسخ.

در بخش ۲، یک مدل قابلیت اطمینان، به منظور ارزیابی قابلیت اطمینان در نقاط بار، شرح داده شده و پس از آن در بخش ۳، نحوه محاسبه تلفات در شبکه‌های توزیع توضیح داده شده است. در بخش ۴، روش بازآرایی شبکه برق با قیود قابلیت اطمینان و تلفات توضیح داده می‌شود. سپس از چند الگوریتم مختلف برای پیدا کردن ساختار بهینه در جهت کاهش تابع هزینه در شبکه‌های توزیع ۳۳ شینه و ۱۶ شینه IEEE استفاده گردیده و در نهایت در بخش ۵ نتایج به دست آمده از الگوریتم‌های مختلف با الگوریتم خفاش مقایسه شده است. با توجه به نتایج به دست آمده، الگوریتم خفاش باینری از نظر دقت و سرعت به عنوان مناسب‌ترین الگوریتم معرفی می‌شود.

۲. قابلیت اطمینان

۱.۲. تعیین مسیر بار با شروع از شین مرجع

در این بخش، از مدل قابلیت اطمینان احتمالی تجهیزات مختلف به منظور محاسبه قابلیت اطمینان در نقطه بارهای مختلف استفاده می‌شود. اولین مرحله شامل تعیین کوتاه‌ترین مسیر بین هر بار و شین مرجع است. این امر با استفاده از ماتریس PC بیان شده در مرجع [۱۸] انجام می‌شود. یکی از ویژگی‌های منحصر به فرد ماتریس PC این است که در محاسبه قابلیت اطمینان شبکه‌های شعاعی، مسیر بین بار و شین مرجع را به آسانی تعیین می‌کند. برای نشان دادن ویژگی‌های ماتریس

۲.۲. محاسبه قابلیت اطمینان شبکه

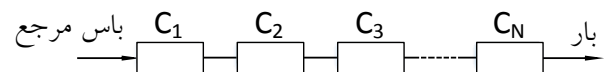
در مرحله دوم، مدل قابلیت اطمینان برای هر تجهیز به دست می‌آید. برای تعیین این مدل، از احتمال در دسترس پذیری هر تجهیز به صورت زیر استفاده می‌شود [۱۴]:

$$P = \frac{MTTF}{MTTF + MTTR} = \frac{\sum_i \frac{1}{\lambda_i}}{\sum_i \frac{1}{\lambda_i} + \sum_i \frac{1}{\mu_i}} \quad (1)$$

که P احتمال در دسترس پذیری، $MTTF$ متوسط زمان خرابی و $MTTR$ متوسط زمان تعمیر هر تجهیز است. نیز بیانگر نرخ خرابی و تعمیر هر تجهیز هستند. برای تعیین قابلیت اطمینان هر بار ابتدا مسیر بین بار و شین مرجع مشخص شده، سپس با استفاده از معادله (۱) و مطابق شکل (۳)، عدم قطعیت بار از معادله (۲) محاسبه می‌شود.

$$Q = P(U_i C_i) \quad (2)$$

در این معادله، Q عدم قطعیت در هر نقطه بار، C_i نشان‌دهنده تجهیزات موجود در مسیر بین بار و شین مرجع و $P(U_i C_i)$ نیز بیانگر مجموع عدم در دسترس پذیری تجهیزات شبکه است.



شکل (۳): فیدر نمونه در محاسبه عدم قطعیت [۱۹]

با توجه به اینکه در یک شبکه توزیع، نقطه بارهای مختلفی وجود دارد، می‌توان از مقدار متوسط عدم قطعیت برای تعیین قابلیت اطمینان شبکه استفاده کرد. بدین منظور پس از محاسبه عدم قطعیت از معادله (۳) متوسط عدم قطعیت شبکه به دست خواهد آمد و سپس با استفاده از معادله (۴) قابلیت اطمینان شبکه محاسبه می‌شود.

$$Q_{SA} = \frac{1}{L} \sum_{i=1}^L Q_{i(LoadPoint)} \quad (3)$$

در معادله فوق، L تعداد بارهای متصل شده به شبکه است. در نهایت برای محاسبه قابلیت اطمینان احتمالی شبکه، معادله (۴) به کار برده می‌شود.

$$R = 1 - Q \quad (4)$$

برای مثال، قابلیت اطمینان شبکه ۷ شینه نشان داده شده در شکل (۱) با در نظر گرفتن اطلاعات جدول (۱) محاسبه می‌شود.

جدول (۱): نرخ خروج و تعمیر اجزا شبکه ۷ شینه [۱۹]	
نرخ خرابی	۰/۰۲ خروج در سال
زمان تعمیر	۳۰ ساعت
نرخ نگهداری	۰/۲ خروج در سال
زمان نگهداری	۲۰ ساعت

با استفاده از معادله (۱)، دسترس پذیری هر تجهیز برابر ۰/۹۹۹۸۹۶ و عدم دسترس پذیری آن با استفاده از معادلات (۲) تا (۴) برابر با ۰/۰۰۰۱۰۳۷۶۶۷ به دست می‌آید. نتایج عدم قطعیت به دست آمده در شبکه ۷ شینه، در جدول (۲) نشان داده شده است.

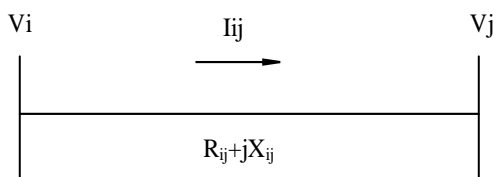
۳. محاسبه تلفات در شبکه‌های توزیع

در این مقاله برای محاسبه پخش بار شبکه، از روش جاروب پیشرو-پسرو توضیح داده شده در مرجع [۱۸] استفاده شده است. در این روش، ابتدا جریان هر بار با استفاده از معادله (۵) محاسبه شده و سپس با استفاده از KCL در هر شین جریان همه شاخه‌ها محاسبه می‌شود. پس از آن، طبق شکل (۴) با استفاده از معادله (۶) از طریق KVL با شروع از شین مرجع، ولتاژ شین‌ها محاسبه می‌شود. پس از همگرا شدن ولتاژ شین‌ها و به دست آمدن جریان خطوط شبکه، تلفات هر خط از معادله (۷) به دست می‌آید. در این حالت، تلفات شبکه برابر مجموع تلفات همه خطوط خواهد بود.

$$I_i = \frac{P_i^* - jQ_i^*}{V_i^*} \quad (5)$$

$$V_j = V_i - I_{ij} \times R_{ij} \quad (6)$$

$$P_{ij} = R_{ij} \times |I_{ij}|^2 \quad (7)$$



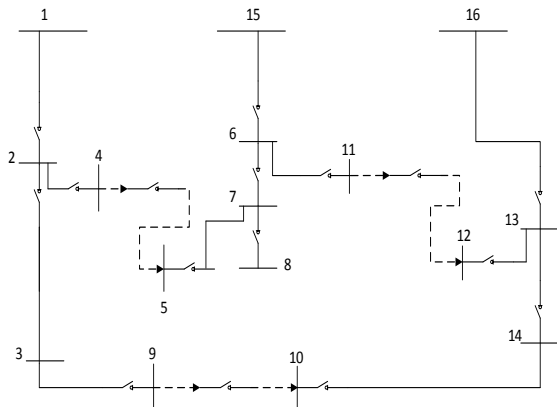
شکل (۴): محاسبه KVL بین دو شین [۱۸]

(۹) در معادلات فوق V_i, I_i, P_i و Q_i به ترتیب نشان‌دهنده

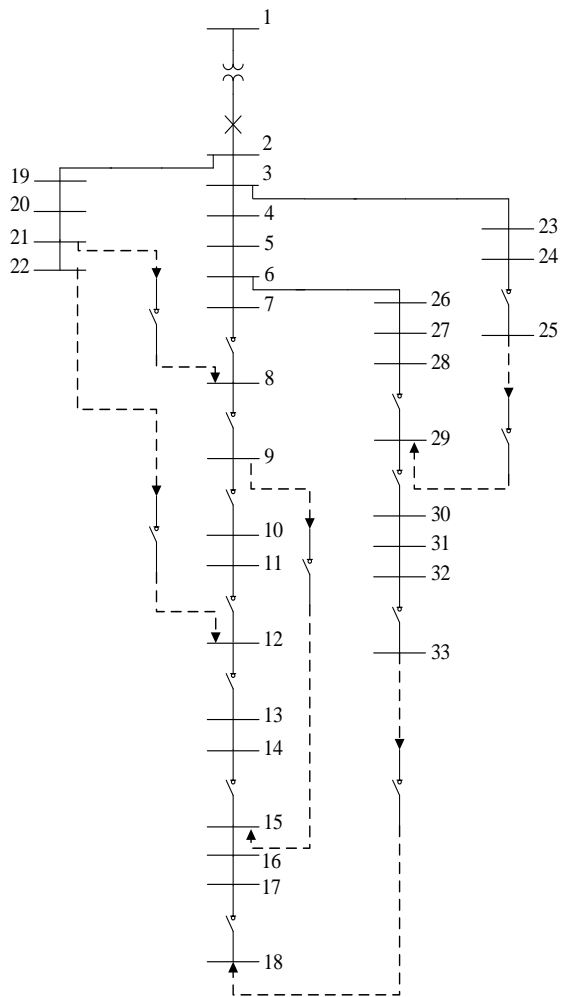
(۱۰) جریان، ولتاژ، توان اکتیو و توان راکتیو هر بار، I_{ij} و P_{ij}

(۱۱) به ترتیب بیانگر جریان و تلفات توان هر خط می‌باشند.

ساختار شبکه ۱۶ شینه و ۳۳ شینه به ترتیب در شکل‌های (۵) و (۶) نشان داده شده است.



شکل (۵): ساختار شبکه ۱۶ شینه [۲۰]



شکل (۶): ساختار شبکه ۳۳ شینه [۱۹]

۴. بازآرایی شبکه

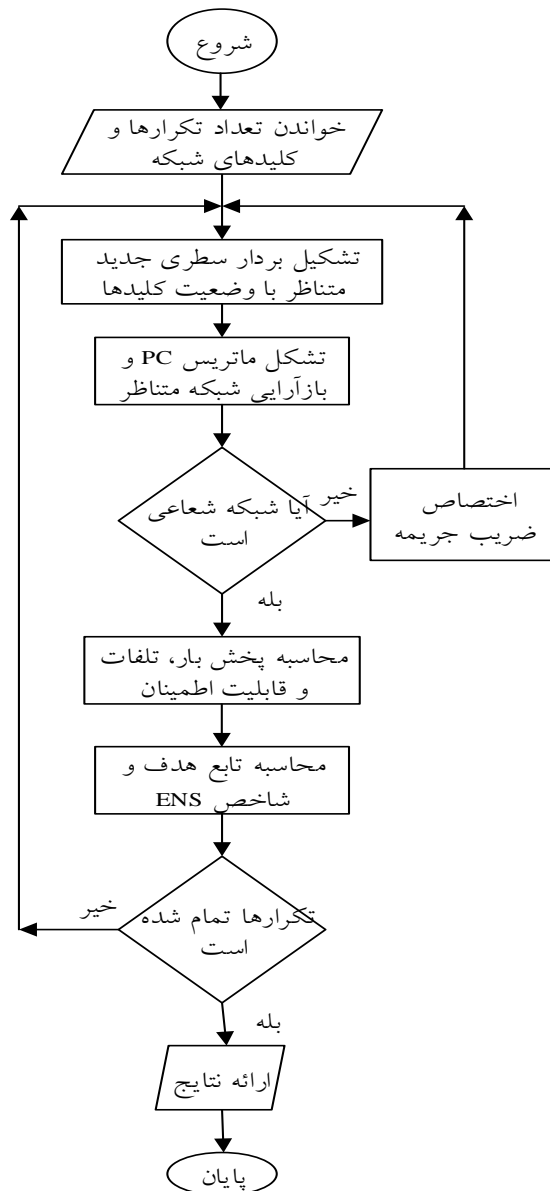
در این مقاله، بازآرایی با هدف کاهش تلفات و افزایش قابلیت اطمینان صورت گرفته است. آرایش بهینه کلیدها با استفاده از الگوریتم‌های ازدحام ذرات باینری، ژنتیک باینری، سنجاچک باینری، خفاش باینری و فاخته باینری و با توجه به توابع هدف بیان شده در معادلات (۸) و (۱۲) انجام می‌شود. در این خصوص، تنظیم پارامترها بر اساس روش سعی و خطا، و همچنین تنظیم الگوریتم‌ها بر اساس توابع استاندارد جمعی صورت گرفته است. روش یافتن آرایش بهینه بدین صورت است که ابتدا یک بردار سطری با اندازه‌ای برابر با تعداد کلیدهای شبکه، توسط الگوریتم هوشمند تولید می‌شود. این بردار تنها مقادیر صفر و یک را شامل می‌شود. مقدار صفر نشانگر باز بودن کلید و یک حاکی از بسته بودن کلید است. شرط اساسی در بازآرایی، شعاعی باقی ماندن شبکه و برقرار بودن تمام شین‌های شبکه می‌باشد؛ در غیر این صورت یک ساختار ناممکن حاصل می‌شود. با توجه به مطالب فوق، تابع هدف به صورت زیر تعریف می‌شود [۲۰]:

$$J_{(x)} = \begin{cases} C & \text{ساختار ناممکن} \\ w_1 Q_{SA} + w_2 P_{Loss} & \text{ساختار ممکن} \end{cases} \quad (۸)$$

که C (یک عدد خیلی بزرگ) ضریب جریمه ساختارهای ناممکن است. در این تابع هدف، ساختار ناممکن به ساختاری گفته می‌شود که در آن شبکه شعاعی نباشد و یا آنکه تمام شین‌های شبکه در آن ساختار برقرار نباشد که اگر چنین حالتی پیش آید، بدان حالت یک ضریب جریمه اطلاق خواهد شد، به صورتی که هزینه آن در تابع هزینه نسبت به ساختارهای ممکن بسیار بیشتر باشد. معادلات (۹) تا (۱۱) برای تعیین شاخص ENS استفاده می‌شوند. در این مقاله، روابط (۹) تا (۱۱) تنها در شبکه ۳۳ شینه و با فرض اینکه بیشینه بار شبکه برابر با ۳۷۱۵ کیلووات، ضریب بار برابر با $0.5 (LF=0.5)$ ، ضریب تقاضا برابر ۱ و ضریب تلفات برابر با ۰/۳۷۵ باشد، مورد استفاده قرار گرفته است.

جدول (۲): محاسبه عدم قطعیت شبکه ۷ شینه

شین بار	مسیر بین بار و شین مرجع	عدم قطعیت
۲	۲-۸-۱	۰/۰۰۰۳۱۱
۳	۳-۹-۲-۸-۱	۰/۰۰۰۵۱۸
۴	۴-۱۰-۲-۸-۱	۰/۰۰۰۵۱۸
۵	۵-۱۱-۳-۹-۲-۸-۱	۰/۰۰۰۷۲۶
۶	۶-۱۲-۳-۹-۲-۸-۱	۰/۰۰۰۷۲۶
۷	۷-۱۳-۴-۱۰-۲-۸-۱	۰/۰۰۰۷۲۶
عدم قطعیت شبکه		۰/۰۰۰۵۸۸



شکل (۷): فلوجارت حل مسئله

در شبکه ۱۶ شینه، تابع هدف به صورت زیر در نظر گرفته شده است:

$$J_{(x)} = \begin{cases} C & \text{اگر ساختار ممکن نباشد} \\ W_1 P_{Loss} & \text{اگر ساختار ممکن باشد} \end{cases} \quad (12)$$

در این شبکه، آرایش بهینه تنها با در نظر گرفتن قید تلفات محاسبه می شود. فلوجارت کلی حل مسئله در شکل (۷) آورده شده است. مراحل فلوجارت به صورت زیر است:

- در ابتدا تعداد جمعیت، تکرارهای الگوریتم و کلیدهای موجود در شبکه توسط کاربر وارد می شود.
- الگوریتم، برداری متشکل از صفر و یک متناظر با تعداد کلیدهای وارد شده تشکیل می دهد.
- با توجه به وضعیت کلیدها، شبکه بازآرایی شده و ماتریس PC آن تشکیل می شود.
- در این مرحله، شعاعی بودن شبکه بررسی شده و اگر شبکه در آرایش فعلی، شعاعی نباشد به مرحله ۲ می رویم.
- اگر شبکه شعاعی باشد، پخش بار پیشرو- پسرو انجام شده و قابلیت اطمینان شبکه و تلفات آن معین می شود.
- با استفاده از نتایج به دست آمده در مرحله قبل، تابع هدف محاسبه شده و شاخص ENS نیز به دست می آید.
- در این مرحله، اگر تعداد تکرارهای صورت گرفته تکمیل نشده باشد به مرحله ۲ می رویم؛ در غیر این صورت نتایج نهایی اعم از نمودار همگرایی، زمان حل مسئله و آرایش بهینه محاسبه شده به دست آمده و حل مسئله پایان می یابد.

۵. بررسی نتایج شبیه‌سازی

و بزرگ مهم است و باید مورد ارزیابی قرار گیرد. در این بین الگوریتم فاخته دارای کمترین زمان حل یا بالاترین سرعت همگرایی بوده و پس از آن الگوریتم خفاش قرار دارد. بیشترین زمان نیز مربوط به الگوریتم‌های سنجاچک و ازدحام ذرات است. نمودار متوسط همگرایی الگوریتم‌ها نیز در شکل (۸) نشان داده شده است. روش استخراج این نمودار بدین صورت است که نمودار متوسط همگرایی هر الگوریتم به‌ازای ۲۰ اجرا، که در هر اجرا تکرار نرم‌افزار برابر با ۱۶۰ است، به دست آمده است. نمودار همگرایی نشان‌دهنده آن است که هر الگوریتم پس از چند تکرار درونی به جواب بهینه خود دست می‌یابد. هرچه تعداد تکرار در یک الگوریتم بیشتر باشد، حجم محاسبات در آن الگوریتم نیز بیشتر خواهد بود. با توجه به نمودار شکل (۸) الگوریتم فاخته بیشتر حجم محاسباتی را به خود اختصاص می‌دهد که این در مسائل پیچیده و بزرگ نیازمند داشتن سیستم‌های محاسباتی با مشخصات فنی بالا است. از سوی دیگر، حجم محاسباتی بالا می‌تواند در سرعت عملکرد الگوریتم نیز تأثیر منفی داشته باشد. همچنین با توجه به این نمودار، الگوریتم‌های خفاش و ازدحام ذرات دارای بهترین نمودار همگرایی و کمترین حجم محاسبات هستند. لذا با توجه به پاسخ به‌دست‌آمده، نمودار همگرایی و سرعت حل مسئله بازآرایی، الگوریتم خفاش باینری عملکرد بهتری نسبت به سایر الگوریتم‌ها داشته است.

۲.۵. مطالعه موردی دوم - شبکه تست ۳۳ شینه IEEE

برای مقایسه بهتر و اطمینان بیشتر از صحت نتایج به‌دست‌آمده در قسمت قبل، شبکه تست ۳۳ شینه IEEE نیز با یک تابع هدف متفاوت مورد ارزیابی قرار می‌گیرد. این شبکه شامل ۱۶ کلید و ۳۳ شین می‌باشد که در آن ۳۲ شین مربوط به بار بوده و یک شین آن شین مرجع است. ولتاژ خط برابر با ۱۲/۶۶ کیلوولت و مجموع بارهای متصل به شبکه ۳۷۱۵ کیلووات است. تابع هدف در این بخش شامل حالت‌های مختلف معادله (۸) می‌باشد. در این بخش، جهت ارزیابی بیشتر و بهتر، تابع هدف با اهداف مختلفی در سیستم تعریف شده است.

در این بخش، دو شبکه استاندارد ۱۶ شینه و ۳۳ شینه IEEE مورد مطالعه قرار گرفته و نتایج شبیه‌سازی در جداول (۳) و (۴) ارائه شده است. بازآرایی توسط الگوریتم‌های باینری خفاش، ازدحام ذرات، ژنتیک، سنجاچک و فاخته انجام شده است. تابع هدف در شبکه ۱۶ شینه معادله (۱۲) و در شبکه ۳۳ شینه معادله (۸) است. برای مقایسه بهتر نتایج، نمودار همگرایی هر الگوریتم نیز آورده شده است. شبیه‌سازی‌های صورت‌گرفته با استفاده از یک لپ‌تاپ با مشخصات Asus RAM 8GB، Corei7 و Cache 3MB انجام شده است.

۱.۵. مطالعه موردی اول - شبکه تست ۱۶ شینه IEEE

شبکه ۱۶ شینه، شامل ۱۶ کلید و سه فیدر تغذیه است. در این شبکه، ۱۳ شین مربوط به بار و سه شین جهت تغذیه بارها وجود دارد که در اینجا به‌صورت یک شین مرجع در نظر گرفته شده‌اند. ولتاژ خط شبکه برابر با ۲۳ کیلوولت و مجموع بارهای متصل به سیستم ۷/۲۸ کیلووات است. تابع هدف در این بخش، تابع بیان‌شده در معادله (۱۲) است. بازآرایی در این شبکه تنها با هدف حداقل نمودن تلفات صورت می‌پذیرد. نتایج مربوط به این شبکه در جدول (۳) و نمودار همگرایی آن در شکل (۸) آورده شده است. در جدول (۳)، نتایج برای دو حالت مختلف شبکه نشان داده شده است. این دو حالت عبارت است از حالت پایه، که نشان‌دهنده ساختار شبکه پیش از بازآرایی و حالت به‌دست‌آمده با هدف حداقل نمودن تلفات، که نشان‌دهنده ساختار شبکه پس از بازآرایی شبکه با هدف مذکور است. با توجه به جدول (۳) مشاهده می‌شود که نتایج به‌دست‌آمده برای میزان تلفات توان، تلفات سالانه انرژی و هزینه محاسبه‌شده در تابع هزینه توسط همه الگوریتم‌ها برابر، و کمتر از حالت پایه است؛ و تمامی الگوریتم‌ها پاسخ یکسانی را ارائه می‌دهند. با وجود این نمی‌توان ادعا کرد که عملکرد تمام الگوریتم‌ها عالی و بدون نقص است؛ زیرا تا بدین جا تنها دقت الگوریتم‌ها مورد سنجش قرار گرفته است، این در حالی است که سرعت عمل نیز در حل مسائل پیچیده

جدول (۳): نتایج بازآرایی برای حالت‌های مختلف در شبکه ۱۶ شینه

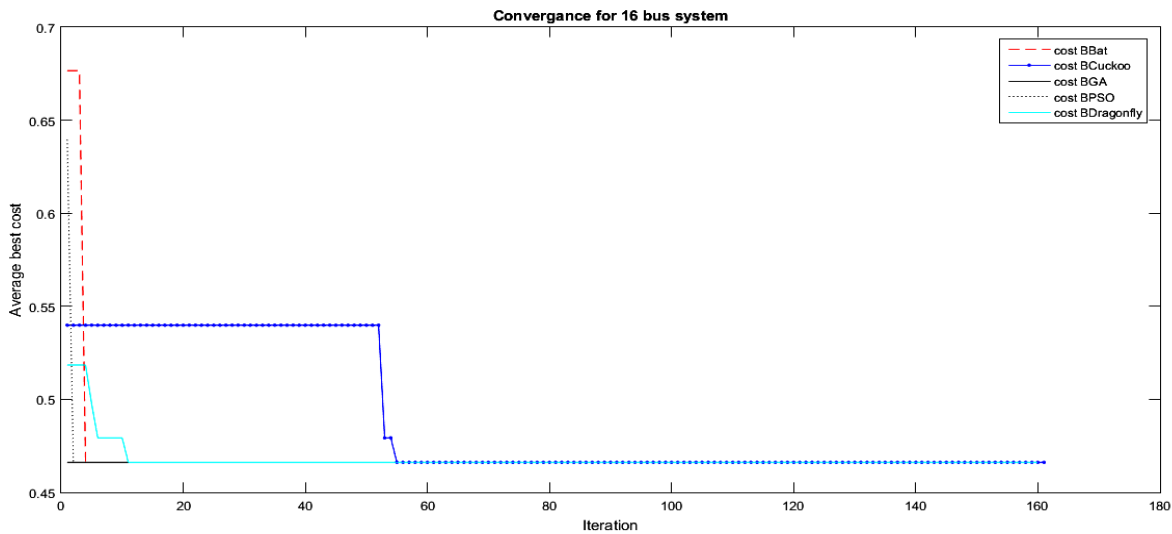
Item	Algorithms	Base case	Reconfiguration for min loss
Open switch	BCuckoo	۱۰-۹، ۱۲-۱۱، ۵-۴	۱۰-۹، ۱۱-۶، ۷-۵
	BBAT	۱۰-۹، ۱۲-۱۱، ۵-۴	۱۰-۹، ۱۱-۶، ۷-۵
	BGA	۱۰-۹، ۱۲-۱۱، ۵-۴	۱۰-۹، ۱۱-۶، ۷-۵
	BDragonfly	۱۰-۹، ۱۲-۱۱، ۵-۴	۱۰-۹، ۱۱-۶، ۷-۵
	BPSO [14]	۱۰-۹، ۱۲-۱۱، ۵-۴	۱۰-۹، ۱۱-۶، ۷-۵
Loss (Kw)	BCuckoo	۰/۵۱۱۴	۰/۴۶۶۱
	BBAT	۰/۵۱۱۴	۰/۴۶۶۱
	BGA	۰/۵۱۱۴	۰/۴۶۶۱
	BDragonfly	۰/۵۱۱۴	۰/۴۶۶۱
	BPSO [15]	۰/۵۱۱۴	۰/۴۶۶۱
Energy loss per year	BCuckoo	۱۶۷۹/۹۴۹	۱۵۳۱/۱۳۸۵
	BBAT	۱۶۷۹/۹۴۹	۱۵۳۱/۱۳۸۵
	BGA	۱۶۷۹/۹۴۹	۱۵۳۱/۱۳۸۵
	BDragonfly	۱۶۷۹/۹۴۹	۱۵۳۱/۱۳۸۵
	BPSO [15]	۱۶۷۹/۹۴۹	۱۵۳۱/۱۳۸۵
Cost	BCuckoo	۰/۵۱۱۴	۰/۴۶۶۱
	BBAT	۰/۵۱۱۴	۰/۴۶۶۱
	BGA	۰/۵۱۱۴	۰/۴۶۶۱
	BDragonfly	۰/۵۱۱۴	۰/۴۶۶۱
	BPSO [14]	۰/۵۱۱۴	۰/۴۶۶۱
CPU Time (sec)	BBAT	-----	۲۷
	BCuckoo	-----	۵۰
	BGA	-----	۱۶۰
	BDragonfly	-----	۱۸۰۰
	BPSO [14]	-----	۹۹۸

جدول (۴): نتایج بازآرایی برای حالت‌های مختلف در شبکه ۳۳ شینه

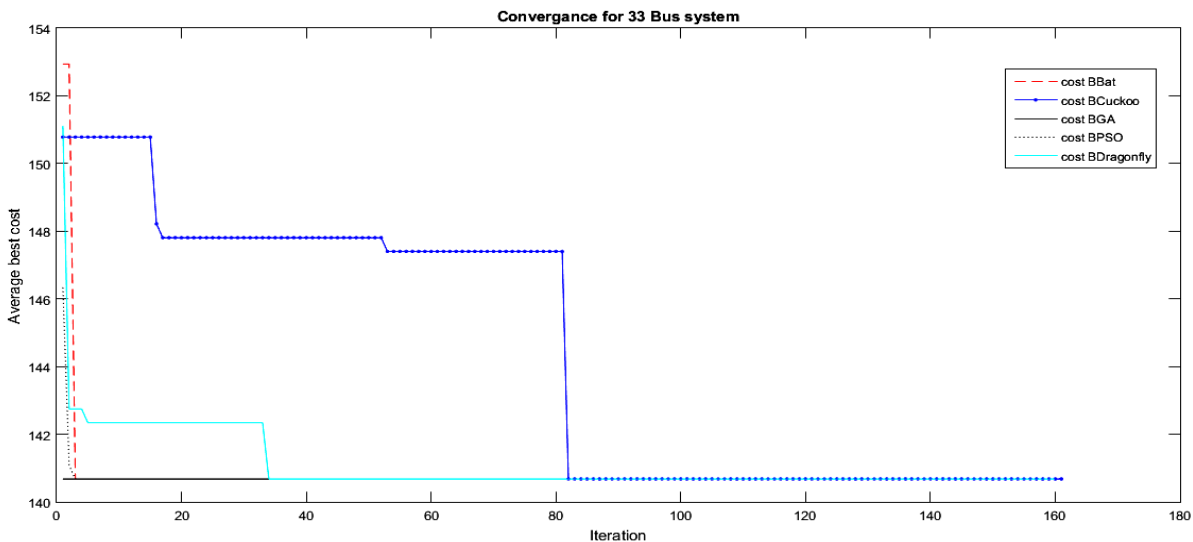
Item	Algorithms	Base case	Reconfiguration for max reliability	Reconfiguration for min loss	Reconfiguration for max reliability and min loss	
					W1=1000 W2=1	W1=10000 W2=1
Open switch	BCuckoo	۲۲-۱۲۱۵-۹، ۲۱-۸ ۲۹-۲۵، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۸، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲
	BPSO	۲۲-۱۲۱۵-۹، ۲۱-۸ ۲۹-۲۵، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۸، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲
	BBAT	۲۲-۱۲۱۵-۹، ۲۱-۸ ۲۹-۲۵، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۸، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲
	BGA	۲۲-۱۲۱۵-۹، ۲۱-۸ ۲۹-۲۵، ۳۳-۱۸	۱۵-۱۴، ۱۲-۱۱، ۲۱-۸ ۲۹-۲۸، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۸، ۳۳-۳۲
	BDragonfly	۲۲-۱۲۱۵-۹، ۲۱-۸ ۲۹-۲۵، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۸، ۳۳-۱۸	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲	۱۵-۱۴، ۱۰-۹، ۸-۷ ۲۹-۲۵، ۳۳-۳۲

ادامه جدول (۴): نتایج بازاریابی برای حالت‌های مختلف در شبکه ۳۳ شینه

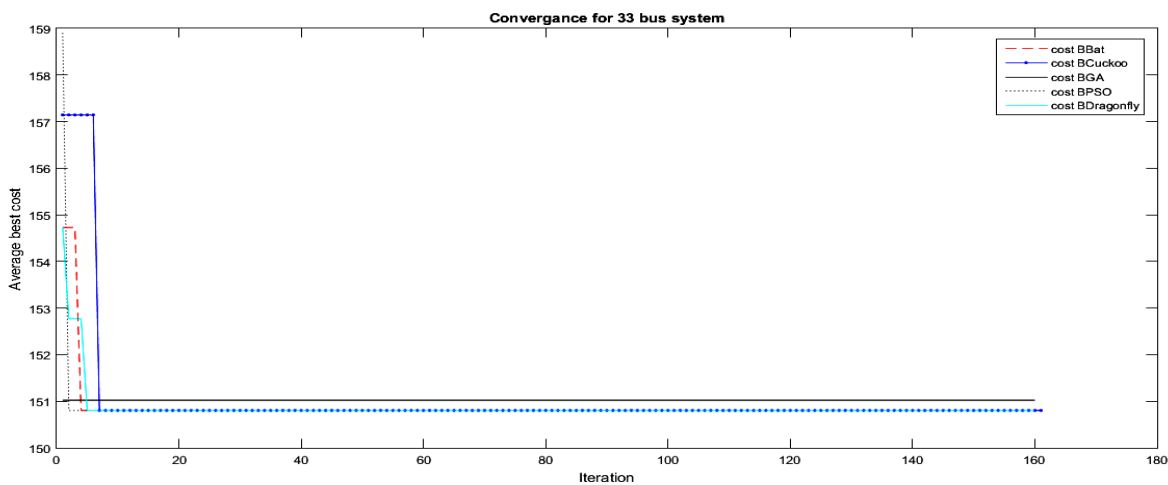
Item	Algorithms	Base case	Reconfiguration for max reliability	Reconfiguration for min loss	Reconfiguration for max reliability and min loss	
					W1=1000 W2=1	W1=10000 W2=1
Unreliability	BCuckoo	۰/۰۰۱۴۰۴۳	۰/۰۰۱۰۹۶۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰
	BBAT	۰/۰۰۱۴۰۴۳	۰/۰۰۱۰۹۶۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰
	BGA	۰/۰۰۱۴۰۴۳	۰/۰۰۱۱۰۴۲	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۰۴۲
	BDragonfly	۰/۰۰۱۴۰۴۳	۰/۰۰۱۰۹۶۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰
	BPSO [14]	۰/۰۰۱۴۰۴۳	۰/۰۰۱۰۹۶۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰	۰/۰۰۱۱۲۵۰
Down time (hours/year)	BCuckoo	۱۲/۶۴۱۰	۹/۶۰۱۰	۹/۸۵۵۰	۹/۸۵۵۰	۹/۸۵۵۰
	BBAT	۱۲/۶۴۱۰	۹/۶۰۱۰	۹/۸۵۵۰	۹/۸۵۵۰	۹/۸۵۵۰
	BGA	۱۲/۶۴۱۰	۹/۶۷۲۸	۹/۸۵۵۰	۹/۸۵۵۰	۹/۶۷۲۸
	BDragonfly	۱۲/۶۴۱۰	۹/۶۰۱۰	۹/۸۵۵۰	۹/۸۵۵۰	۹/۸۵۵۰
	BPSO [14]	۱۲/۶۴۱۰	۹/۶۰۱۰	۹/۸۵۵۰	۹/۸۵۵۰	۹/۸۵۵۰
ENS (Kwh)	BCuckoo	۱۲/۶۴۱۰	۱۷۸۳۴/۰۱۳۰	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷
	BBAT	۱۲/۶۴۱۰	۱۷۸۳۴/۰۱۳۰	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷
	BGA	۱۲/۶۴۱۰	۱۷۹۶۷/۳۹۸۲	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷	۱۷۹۶۷/۳۹۸۲
	BDragonfly	۱۲/۶۴۱۰	۱۷۸۳۴/۰۱۳۰	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷
	BPSO [14]	۱۲/۶۴۱۰	۱۷۸۳۴/۰۱۳۰	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷	۱۸۳۰۵/۶۷۸۷
Loss (Kw)	BCuckoo	۲۱۰/۹۸۶۸	۱۴۱/۹۱۸۷	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶
	BBAT	۲۱۰/۹۸۶۸	۱۴۱/۹۱۸۷	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶
	BGA	۲۱۰/۹۸۶۸	۱۴۶/۹۱۳۸	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶	۱۳۹/۹۱۰۶
	BDragonfly	۲۱۰/۹۸۶۸	۱۴۱/۹۱۸۷	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶
	BPSO [14]	۲۱۰/۹۸۶۸	۱۴۱/۹۱۸۷	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶	۱۳۹/۵۵۳۶
Energy loss per year	BCuckoo	۶۹۳۰۹۱/۷۸۲۴	۴۶۶۲۰۳/۲۴۴	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵
	BBAT	۶۹۳۰۹۱/۷۸۲۴	۴۶۶۲۰۳/۲۴۴	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵
	BGA	۶۹۳۰۹۱/۷۸۲۴	۴۸۲۶۱۲/۱۱۲۹	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵	۴۵۹۱۳۶/۳۵۶۹
	BDragonfly	۶۹۳۰۹۱/۷۸۲۴	۴۶۶۲۰۳/۲۴۴	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵
	BPSO [14]	۶۹۳۰۹۱/۷۸۲۴	۴۶۶۲۰۳/۲۴۴	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵	۴۵۸۴۳۳/۸۲۳۵
Cost	BCuckoo	-----	۰/۰۰۱۰۹۶۰	۱۳۹/۵۵۳۶	۱۴۰/۶۷۸۶	۱۵۰/۸۰۳۶
	BBAT	-----	۰/۰۰۱۰۹۶۰	۱۳۹/۵۵۳۶	۱۴۰/۶۷۸۶	۱۵۰/۸۰۳۶
	BGA	-----	۰/۰۰۱۱۰۴۲	۱۳۹/۵۵۳۶	۱۴۰/۶۷۸۶	۱۵۱/۰۲۲۷
	BDragonfly	-----	۰/۰۰۱۰۹۶۰	۱۳۹/۵۵۳۶	۱۴۰/۶۷۸۶	۱۵۰/۸۰۳۶
	BPSO [14]	-----	۰/۰۰۱۰۹۶۰	۱۳۹/۵۵۳۶	۱۴۰/۶۷۸۶	۱۵۰/۸۰۳۶
CPU Time (sec)	BCuckoo	-----	۱۹۰	۱۹۰	۱۹۰	۱۹۰
	BBAT	-----	۳۱۵	۳۱۵	۳۱۵	۳۱۵
	BGA	-----	۸۵۰	۸۵۰	۸۵۰	۸۵۰
	BDragonfly	-----	۱۸۰۰	۱۸۰۰	۱۸۰۰	۴۰۰۰
	BPSO [14]	-----	۷۳۰۰	۷۳۰۰	۷۳۰۰	۷۳۰۰



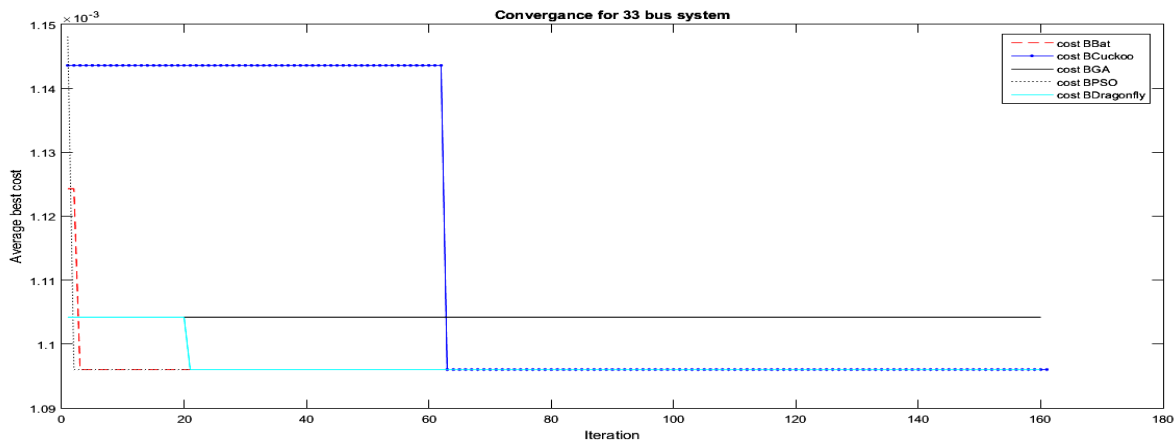
شکل (۸): نمودار همگرایی حل سیستم ۱۶ شینه IEEE



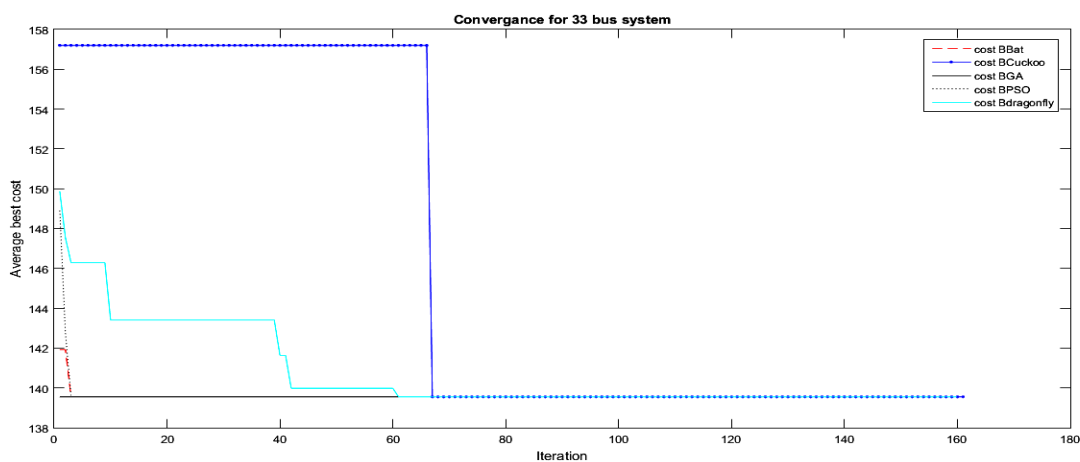
شکل (۹): نمودار همگرایی حل سیستم ۳۳ شینه IEEE با ضرایب $W_2=1, W_1=1000$



شکل (۱۰): نمودار همگرایی حل سیستم ۳۳ شینه IEEE با ضرایب $W_2=1, W_1=10000$



شکل (۱۱): نمودار همگرایی حل سیستم ۳۳ شینه IEEE با ضرایب $W_2=0$, $W_1=1$



شکل (۱۲): نمودار همگرایی حل سیستم ۳۳ شینه IEEE با ضرایب $W_2=1$, $W_1=0$

(۴)، در حالتی که تابع هدف بیشینه نمودن قابلیت اطمینان است، تمام الگوریتم‌ها پاسخ یکسانی ارائه داده‌اند اما در الگوریتم ژنتیک (درحالی‌که تنظیم پارامترها نسبت به دیگر حالت‌ها تفاوتی ندارد) جواب بهینه حاصل نشده است و در این الگوریتم، تلفات توان، تلفات سالانه انرژی، عدم قابلیت اطمینان، انرژی تغذیه نشده، ساعات خاموشی در سال و هزینه به دست آمده از تابع هدف، بیشتر از سایر الگوریتم‌هاست. با توجه به جدول (۴) و با مقایسه نتایج حالت پایه و نتایج به دست آمده با حالتی که در آن هدف، بیشینه نمودن قابلیت اطمینان است، مشخص است که الگوریتم ژنتیک توانسته به پاسخ بهینه نزدیک شود اما بدان دست نیافته است. در حالتی که هدف کمینه نمودن تلفات است، تمام الگوریتم‌ها پاسخ یکسانی را ارائه داده‌اند. با توجه به اینکه تغییر ساختار مسئله

نتایج شبیه‌سازی این شبکه در جدول (۴) و نمودار متوسط همگرایی آن (طبق شرایط بیان شده در قسمت قبل) در حالت‌های مختلف در شکل‌های (۹) تا (۱۲) آورده شده است. جدول (۴) نشان‌دهنده حالت پایه و نتایج بازآرایی با اهداف زیر است:

۱. بازآرایی با هدف بیشینه نمودن قابلیت اطمینان؛
۲. بازآرایی با هدف کمینه نمودن تلفات؛
۳. بازآرایی با هدف بیشینه نمودن قابلیت اطمینان و کمینه نمودن تلفات (که این حالت خود با ضرایب وزنی متفاوت انجام شده است).

در این قسمت، برای آنکه عملگر الگوریتم‌ها به‌طور دقیق‌تری ارزیابی شود، شبیه‌سازی با ضرایب وزنی متفاوت و توابع هدف متفاوتی صورت گرفته است. با توجه به جدول

مختلف در شبکه‌های متفاوت مورد بررسی قرار گرفت. با توجه به نتایج به دست آمده، مشاهده شد که الگوریتم خفاش باینری با توجه به فاکتورهای مذکور، عملکرد بهتری نسبت به سایر الگوریتم‌ها دارد و به عنوان یک الگوریتم بهینه‌سازی مطلوب در حل مسائلی که قابلیت مدل‌سازی ریاضی در آن‌ها وجود ندارد، پیشنهاد می‌شود. به عنوان زمینه تحقیق آتی، می‌توان عملکرد الگوریتم خفاش را با سایر الگوریتم‌های موجود که در این مقاله مورد بررسی واقع نشده است، و در حل مسائل دیگر بهینه‌سازی ارزیابی کرد.

ضمائم

جدول (۵): داده‌های شبکه نمونه ۳۳ شینه IEEE

From	To	R(Ohm)	X(Ohm)	End bus real load(kW)	End bus reactive load(kVAr)
۱	۲	۰/۰۹۲۲	۰/۰۴۷۷	۱۰۰	۶۰
۲	۳	۰/۴۹۳	۰/۲۵۱۱	۹۰	۴۰
۳	۴	۰/۳۶۶	۰/۱۸۶۴	۱۲۰	۸۰
۴	۵	۰/۳۸۱۱	۰/۱۹۴۱	۶۰	۳۰
۵	۶	۰/۸۱۹	۰/۷۰۷	۶۰	۲۰
۶	۷	۰/۱۸۷۲	۰/۶۱۸۸	۲۰۰	۱۰۰
۷	۸	۱/۷۱۱۴	۱/۲۳۵۱	۲۰۰	۱۰۰
۸	۹	۱/۰۳	۰/۷۴	۶۰	۲۰
۹	۱۰	۱/۰۴	۰/۷۴	۶۰	۲۰
۱۰	۱۱	۰/۱۹۶۶	۰/۰۶۵	۴۵	۳۰
۱۱	۱۲	۰/۳۷۴۴	۰/۱۲۳۸	۶۰	۳۵
۱۲	۱۳	۱/۴۶۸	۱/۱۵۵	۶۰	۳۵
۱۳	۱۴	۰/۵۴۱۶	۰/۷۱۲۹	۱۲۰	۸۰
۱۴	۱۵	۰/۵۹۱	۰/۵۲۶	۶۰	۱۰
۱۵	۱۶	۰/۷۴۶۳	۰/۵۴۵	۶۰	۲۰
۱۶	۱۷	۱/۲۸۹	۱/۷۲۱	۶۰	۲۰
۱۷	۱۸	۰/۷۳۲	۰/۵۷۴	۹۰	۴۰
۲	۱۹	۰/۱۶۴	۰/۱۵۶۵	۹۰	۴۰
۱۹	۲۰	۱/۵۰۴۲	۱/۳۵۵۴	۹۰	۴۰
۲۰	۲۱	۰/۴۰۹۵	۰/۴۷۸۴	۹۰	۴۰
۲۱	۲۲	۰/۷۰۸۹	۰/۹۳۷۳	۹۰	۴۰
۳	۲۳	۰/۴۵۱۲	۰/۳۰۸۳	۹۰	۵۰
۲۳	۲۴	۰/۸۹۸	۰/۷۰۹۱	۴۲۰	۲۰۰
۲۴	۲۵	۰/۸۹۶	۰/۷۰۱۱	۴۲۰	۲۰۰
۶	۲۶	۰/۲۰۳	۰/۱۰۳۴	۶۰	۲۵
۲۶	۲۷	۰/۲۸۴۲	۰/۱۴۴۷	۶۰	۲۵
۲۷	۲۸	۱/۰۵۹	۰/۹۳۳۷	۶۰	۲۰

(تغییر تابع هدف) سبب تغییر در دقت به دست آمده از الگوریتم‌ها (الگوریتم ژنتیک) شد، حالت بعدی که در آن هدف کمینه نمودن تلفات و بیشینه نمودن قابلیت اطمینان است، با دو ضریب وزنی متفاوت در نظر گرفته می‌شود تا بتوان تأثیر تغییر ساختار مسئله را بر عملکرد الگوریتم‌ها در حالتی که تأثیر قابلیت اطمینان متفاوت است مشاهده کرد. با توجه به جدول (۴)، در حالتی که تابع هدف بیشینه نمودن قابلیت اطمینان و کمینه نمودن تلفات است، تمام الگوریتم‌ها پاسخ یکسانی را برای فاکتورهای مختلف ارائه کرده‌اند اما در الگوریتم ژنتیک در حالتی که ضریب وزنی WI بیشتر است پاسخ بهینه به دست نیامده است. این امر نشان‌دهنده آن است که قابلیت اطمینان در الگوریتم ژنتیک نسبت به سایر الگوریتم‌ها کمتر است؛ زیرا با تغییر در ساختار مسئله، پاسخ الگوریتم از حالت بهینه خارج شده است. همان طور که از جدول (۴) مشخص است، از نظر زمان حل نیز همچنان الگوریتم‌های فاخته و خفاش کمترین زمان را نسبت به سایر الگوریتم‌ها دارند. همان گونه که از شکل‌های (۹) تا (۱۲) مشخص می‌شود، نمودار همگرایی الگوریتم‌های خفاش و ازدحام ذرات، در این حالت نیز نسبت به سایر الگوریتم‌ها بهتر است. لذا در شبکه ۳۳ شینه نیز با توجه به پاسخ به دست آمده، سرعت حل و نمودار همگرایی، الگوریتم خفاش باینری در حل مسئله بازآرایی، عملکرد بهتری نسبت به سایر الگوریتم‌ها دارد.

۶. نتیجه گیری

در این مطالعه، برای حل مسئله بازآرایی بهینه شبکه‌های ۱۶ شینه و ۳۳ شینه استاندارد IEEE، به عنوان یک مسئله غیرقابل مدل‌سازی با معادلات ریاضی، و با هدف کاهش تلفات و افزایش قابلیت اطمینان از الگوریتم‌های تکاملی مختلف استفاده شد. برای حل مسئله مذکور، عملکرد الگوریتم‌های باینری خفاش، ازدحام ذرات، ژنتیک، سنجاقک و فاخته با یکدیگر مقایسه شدند. در این مقایسه، موارد متعددی از قبیل زمان، دقت حل مسئله و نمودار همگرایی، در توابع هدف

جدول (۶): داده‌های شبکه نمونه ۱۶ شینه IEEE

From	To	R(Ohm)	X(Ohm)	End bus real load(kW)	End bus reactive load(kVAr)
۱	۲	۰/۰۷۵	۰/۱	۲	۱/۶
۲	۳	۰/۰۹	۰/۱۸	۲	-۰/۴
۲	۴	۰/۰۸	۰/۱۱	۳	۰/۴
۷	۵	۰/۱۱	۰/۱۱	۰/۶	-۰/۵
۱۵	۶	۰/۱۱	۰/۱۱	۴	۲/۷
۶	۷	۰/۰۸	۰/۱۱	۵	۱/۸
۷	۸	۰/۰۸	۰/۱۱	۴/۵	-۱/۷
۳	۹	۰/۰۴	۰/۰۴	۱/۵	۱/۲
۱۴	۱۰	۰/۰۴	۰/۰۴	۲/۱	-۰/۸
۶	۱۱	۰/۱۱	۰/۱۱	۱	۰/۹
۱۳	۱۲	۰/۰۹	۰/۱۲	۱	-۱/۱
۱۶	۱۳	۰/۱۱	۰/۱۱	۱	۰/۹
۱۳	۱۴	۰/۰۸	۰/۱۱	۱	۰/۹
۴	۵	۰/۰۴	۰/۰۴	۰	۰
۱۱	۱۲	۰/۰۴	۰/۰۴	۰	۰
۹	۱۰	۰/۰۹	۰/۱۲	۰	۰

ادامه جدول (۵): داده‌های شبکه نمونه ۳۳ شینه IEEE

From	To	R(Ohm)	X(Ohm)	End bus real load(kW)	End bus reactive load(kVAr)
۲۸	۲۹	۰/۸۰۴۲	۰/۷۰۰۶	۱۲۰	۷۰
۲۹	۳۰	۰/۵۰۷۵	۰/۲۵۸۵	۲۰۰	۶۰۰
۳۰	۳۱	۰/۹۷۴۴	۰/۹۶۳	۱۵۰	۷۰
۳۱	۳۲	۰/۳۱۰۵	۰/۳۶۱۹	۲۱۰	۱۰۰
۳۲	۳۳	۰/۳۴۱	۰/۵۳۰۲	۶۰	۴۰
۲۱	۸	۲	۲	۰	۰
۹	۱۵	۲	۲	۰	۰
۱۲	۲۲	۲	۲	۰	۰
۱۸	۳۳	۰/۵	۰/۵	۰	۰
۲۵	۲۹	۰/۵	۰/۵	۰	۰

مراجع

- [1] Power System Outage Task Force, Final report on the August 14, 2003 blackout in the United States and Canada: Causes and recommendations, Canada, Apr. 2004.
- [2] Billinton, R. and Allan, R.N., Reliability Evaluation of Power Systems, 2nd ed. New York: Plenum, 1996.
- [3] Baran, M.E. and Wu, F.F., "Network reconfiguration in distribution systems for loss reduction and load balancing", IEEE Trans. Power Delivery, vol. 4, no. 2, pp. 1401-1407, Apr. 1989.
- [4] Taher, S.A. and Karimi, M.H., Optimal reconfiguration and DG allocation in balanced and unbalanced distribution systems. Ain Shams Engineering Journal, vol. 5, no. 3, pp. 735-749, 2014.
- [5] Kashem, M.A., Jasmon, G.B. and Ganapathy, V., "A new approach of distribution system reconfiguration for loss minimization", International Journal of Electrical Power and Energy Systems, vol. 22, no. 4, pp. 269-276, May 2000.
- [6] Abur, A., "A modified linear programming method for distribution system reconfiguration", International Journal of Electrical Power and Energy Systems, vol. 18, no. 7, pp. 469-474, Oct. 1996.
- [7] Mosbah, M. Arif, S., Mohammedi R.D. and Zine, R., "Optimal Algerian Distribution Network Reconfiguration Using Antlion Algorithm for Active Power Losses", 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS), Tebessa, pp. 1-6, 2018.
- [8] Shetty, V.J. and Ankaliki, S.G., "Electrical Distribution System Power Loss Reduction and Voltage Profile Enhancement by Network Reconfiguration Using PSO", 2019 Fifth International Conference on Electrical Energy Systems (ICEES), Chennai, India, pp. 1-4, 2019.
- [9] Daud, J.G., Kondo, M. and Patabo, M., "Reconfiguration Distribution Network with Ant Colony", 2018 International Conference on Applied Science and Technology (iCAST), Manado, Indonesia, pp. 349-353, 2018.
- [10] Song, Y., Wang, G., Johans, A. and Wang, "Distribution network reconfiguration for loss reduction using fuzzy controlled evolutionary programming", Proc. IEE Generation Transmission and Distribution, vol. 144, no. 4 pp. 345-350, Jul. 1997.
- [11] Mirjalili, S.A., "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems", Neural Comput & Applic, April 2015.
- [12] Rajabioun, R., "Cuckoo Optimization Algorithm", Applied Soft Computing, vol. 11, pp. 5508-5518, 2011.
- [13] Nara, K., Shiose, A., "Implementation of genetic algorithm for distribution systems loss minimum Reconfiguration", IEEE Trans Power Systems, vol. 7, no. 3, pp. 1044-1051, August 1992.
- [14] Ching-Tzong, Su, Chang-Fu, Chang, Ji-Pyng, Chiou, "Distribution network reconfiguration for loss reduction by ant colony search algorithm", Electric power system research, vol. 75, no. 2-3, pp. 190-199, 2005.
- [15] Yang, X.S., A New Metaheuristic Bat-Inspired Algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NISCO 2010) (Eds. J. R. Gonzalez et al.), Studies in Computational Intelligence, Springer Berlin, 284, Springer, pp. 65-74, 2010.
- [16] Mirjalili, S.A., Mirjalili, S.M., Xin-She Yang, "Binary bat algorithm", Neural Comput & Applic, 2013.
- [17] Ang, X.S., Bat Algorithm for Multiobjective Optimization, Int. J. Bio-Inspired Computation, vol. 3, no.

- 5, pp. 267-274, 2011.
- [18] Eltantawy, A. and Salama, M., "A Novel Zooming Algorithm for Distribution Load Flow Analysis for Smart Grid", IEEE Trans. Smart Grid, vol. 5, no. 4, pp. 1704-1711, 2014.
- [19] Amanulla, B., Chakrabarti, S. and Singh, S.N., "Reconfiguration of Power Distribution Systems Considering Reliability and Power Loss", IEEE Trans. Power Delivery, vol. 27, no. 2, pp. 918-926, April 2012.
- [20] Zhu, J.Z., "Optimal reconfiguration of electrical distribution network using the refined genetic algorithm", Electric Power Systems Research, vol. 62, pp. 37-42, 2002.