



بررسی عملکرد الگوریتم های فرا ابتکاری در پیش بینی پیوند شبکه های پیچیده گراف (مقایسه میان سه الگوریتم جغرافیای زیستی، ژنتیک و ازدحام ذرات)

زهرا سبزی^{۱*}، ناصر شریفی^۲، امیر شیخ احمدی^۳

- ۱- دانشجوی دکتری رشته مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد سنندج، سنندج، ایران.
- ۲- دانشجوی دکتری رشته مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد سنندج، سنندج، ایران.
- ۳- استادیار گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد سنندج، سنندج، ایران.

*z.sabzi65@gmail.com

ارسال: شهریور ماه ۱۴۰۱ پذیرش: مهر ماه ۱۴۰۱

چکیده

امروزه تجزیه و تحلیل پیوندهای موجود در شبکه های اجتماعی یا هر شبکه پیچیده دیگر را به منظور درک تکامل آنها و پیش بینی پیوندهای احتمالی در آینده، انجام می دهند. مسئله پیش بینی پیوند دارای کاربردهای بسیار متنوعی در زمینه های مختلف علوم و مهندسی می باشد. به عنوان مثال، در زیر شاخه های مختلفی از بیو انفورماتیک، شبکه های برهمکنش پروتئین و شبکه های ژنتیکی، پیوندهایی بین گره ها وجود دارد که نشان دهنده یک رابطه متقابل میان آنها می باشد. در این مقاله به بررسی حل مساله چالش بر انگیز پیش بینی پیوند با استفاده از الگوریتم های بهینه سازی فرا ابتکاری پرداخته شده است. برای این منظور، نخست از طریق مطالعه و بررسی ادبیات تحقیق، به فرموله بندی مسئله پیش بینی پیوند به عنوان یک مسئله بهینه سازی برای پیش بینی لینک ها در هر نوع شبکه پرداخته شد. در ادامه نتایج تجربی سه الگوریتم جغرافیای زیستی (BBO)، ازدحام ذرات (PSO) و الگوریتم ژنتیک (GA) بر روی تعدادی از شبکه های واقعی مورد بررسی قرار گرفت. نتایج بدست آمده از این مطالعه ثابت کرد که استفاده از الگوریتم های فرا ابتکاری می تواند به عنوان یک رویکرد مناسب در حل مسائل پیش بینی پیوند در نظر گرفته شود.

کلمات کلیدی: پیش بینی پیوند، هوش ازدحامی، الگوریتم ازدحام ذرات، الگوریتم ژنتیک، الگوریتم جغرافیای زیستی.

۱- مقدمه

یکی از راه های پذیرفته شده به منظور نمایش تعاملات بین افراد در یک جامعه، استفاده از نظریه گراف^۱ می باشد که در آن مجموعه ای از رأس ها، توسط خانواده ای از زوج های مرتب که همان یال ها^۲ هستند به هم متصل می باشند. از این رو یک گره^۳ نشان دهنده یک فرد و یک یال نشان دهنده ارتباط میان افراد آن جامعه می باشد [۱]. شبکه های اجتماعی دارای یک ماهیت پویا است، به طوری که یال ها و رئوس جدیدی در طول زمان در گراف ظاهر می شوند [۲ و ۵]. درک تکامل شبکه های پویا که با تشکیل پیوندهای جدید

¹ Graph theory

² Edge

³ Node

رخ می دهد، یک کار نوظهور، برجسته و چالش برانگیز می باشد، که توجه محققان بسیاری را در حوزه های مختلف به خود جلب کرده است [۶]. یا اینحال، درک پویایی یک گراف به دلیل تعداد زیادی از پارامترهای قابل تغییر، یک مساله پیچیده و چالش بر انگیز می باشد. علاوه بر این، در طول زمان لبه های جدیدی به سرعت توسعه میابد، که نشان دهنده ایجاد تعاملات جدید بین افراد در یک شبکه اجتماعی است. لذا درک نحوه تغییر آنها در طول زمان یک موضوع اساسی است که هنوز به خوبی مورد بررسی قرار نگرفته است [۱ و ۷].

تجزیه و تحلیل پیوندهای موجود در شبکه های اجتماعی یا هر شبکه پیچیده دیگر به منظور درک تکامل آنها در آینده، مسئله جالب پیش بینی پیوند را شکل می دهد [۸-۹]. در مطالعه انجام شده توسط (Al Hasan et al. (2006 [۱۰]، پیش بینی پیوند^۱ به عنوان یکی از مشکلات اساسی در ارتباط با درک تعامل بین دو گره خاص در نظر گرفته شده و به اهمیت بررسی این مقوله تاکید شده است. پیش بینی پیوند روشی برای استنباط پیوندهای غایب از یک شبکه خاص میباشد که از طریق آن میتوان پیوندهای اضافی را که مستقیماً قابل مشاهده نیستند مورد ارزیابی قرار داد [۱۱]. پیش بینی پیوند در زمینه های مختلفی مانند بیوانفورماتیک^۲، تجارت الکترونیک^۳ و سایر مقوله ها بسیار مفید است. در بیوانفورماتیک، از روش های پیش بینی پیوند می توان برای یافتن روابط بین پروتئین ها و تشخیص برخی بیماری های مرتبط استفاده کرد [۱۲]. همچنین، پیش بینی پیوند در تجارت الکترونیک می تواند به ایجاد سیستم های توصیه ای^۴ در فروشگاه ها (پیش بینی خریداران یک محصول بر اساس بررسی خریداران سایر محصولات) بسیار کمک کند [۱]. علاوه بر این، پیش بینی پیوند می تواند به شناسایی گروه های ناشناخته مجرمان کمک کند [۱۳]. در کنار موارد ذکر شده، جستجوهای زیادی در شبکه های هم تالیفی^۵ انجام شده است، که از طریق آن ثابت شده که دو یا چند نویسنده دارای تالیفات مشابه، در شبکه همسایه یکدیگر هستند و احتمال همکاری آنها در آینده نزدیک بسیار بالا بوده و از طریق پیش بینی پیوند قابل شناسایی می باشد [۱]. تاکنون رویکردهای مختلفی به منظور حل مسائل مربوط به پیش بینی پیوند مورد استفاده قرار گرفته که بیشتر آنها بر اساس شاخص های مجاورت^۶ و شباهت^۷ عمل می کنند. از جمله شاخصهای شباهت میتوان به مواردی از قبیل رویکرد همسایه مشترک^۸، رویکرد ضریب جاکارد^۹، رویکرد آدامیک اداری^{۱۰}، رویکرد دل بستگی ترجیحی^{۱۱} و بسیاری دیگر اشاره نمود [۱]. نویسندگان (Liben-Nowell and Kleinberg, (2007 [۷]، یکی از روش های اولیه پیش بینی پیوند را پیشنهاد کردند که به وضوح در زمینه شبکه های اجتماعی قابل استفاده می باشد. آنها در مطالعه خود، هر گره در گراف را مربوط به یک فرد و پیوند بین دو گره را مربوط به ارتباط بین این افراد در نظر گرفتند. مدل یادگیری مورد استفاده در این نوع چیدمان، ابتدا شباهت بین چند گره را با شاخص های شباهت مبتنی بر گراف محاسبه می کند و سپس از روش رتبه بندی برتر^{۱۲} امتیازاتی را بر اساس شباهت بین گره ها در نظر می گیرد. در واقع، در این کار بر عملکرد معیارهای مختلف شباهت مبتنی بر گراف به منظور حل مساله پیش بینی پیوند تمرکز شده بود. با اینحال برخی مطالعات انجام شده در ادبیات تحقیق، بر روی روش های انتخاب ویژگی در حوزه های مرتبط با استفاده از الگوریتم های یادگیری ماشین^{۱۳} نظارت شده و روش های جبر خطی^{۱۴} تمرکز دارند [۱۴-۱۵]. در مطالعه (Al Hasan et al. (2006 [۱۰]، نویسندگان کار خود را به دو شیوه مختلف گسترش دادند. آنها در گام نخست پیشنهاد کردند که با استفاده از داده های خارج از محدوده توپولوژی گراف می توان به طور قابل توجهی نتایج بدست آمده از پیش بینی را ارتقاء داد. در ادامه، آنها انواع معیارهای

¹ Link prediction

² Bioinformatics

³ Ecommerce

⁴ Recommendation systems

⁵ Co-authorship

⁶ Proximity

⁷ Similarity

⁸ Common neighbor approach

⁹ Jaccard coefficient approach

¹⁰ Adamic-adar approach

¹¹ Preferential attachment approach

¹² The top ranking

¹³ Supervised machine learning algorithms

¹⁴ Linear algebraic

شباهت را در ترکیب با تکنیک های یادگیری تحت نظارت^۱ مورد استفاده قرار دادند تا از این طریق مساله پیش بینی پیوند را به عنوان یک مساله طبقه بندی باینری^۲ حل کنند. علاوه بر این، در مطالعه Tylenda et al. (2009) [۱۶] نشان داده شد که استفاده از برجسب های زمانی^۳ در اتصالات گذشته، می تواند عملکرد پیش بینی پیوند را به طور قابل توجهی توسعه دهد.

Song et al. (2009) [۱۵]، از ماتریس عامل بندی^۴ برای محاسبه شباهت تقریبی بین رئوس در یک شبکه اجتماعی واقعی استفاده کردند که دارای تقریباً دو میلیون راس و نود میلیون پیوند بود. با این حال هر الگوریتم مرسوم که هدف آن محاسبه شباهت های زوجی بین گره های یک گراف بزرگ باشد، نتایج چندان مناسبی را در پی نخواهد داشت [۱]. همچنین، در برخی از مطالعات، از رویکردهای مبتنی بر طبیعت^۵ برای حل مشکل پیش بینی پیوند استفاده شده است که به الگوریتم های تکاملی^۶ معروف هستند. الگوریتم های تکاملی از نظر فنی مسائل را با یک رویکرد آزمون و خطا حل میکنند و در خلال یک سری از تکرارها، به راه حل مناسب برای مسئله دست میابند. اساساً، تکنیک های محاسباتی تکاملی از پیشرفت مکرر برای توسعه در یک جمعیت استفاده میکنند. عمده تاً الگوریتم های بهینه سازی فراابتکاری را می توان به عنوان رویه ای در سطوح بالاتر تعریف کرد که با هدف کشف، تولید یا انتخاب یک راه حل مناسب برای یک مسئله بهینه سازی عمل میکنند [۱۷]، [۱۳]. در مطالعه Chen and Chen, (2014) [۱۸]، یک الگوریتم پیش بینی پیوند با استفاده از الگوریتم بهینه سازی کلونی مورچگان^۷ (ACO) مورد ارزیابی قرار گرفت. این الگوریتم با بهره برداری از هوش ازدحامی^۸، از مورچه های شبیه سازی شده برای مساله مورد نظر استفاده میکند. فرمون های ترشح شده و اطلاعات اکتشافی بدست آمده بر اساس مفاهیم پایه ای این الگوریتم، در لبه های گراف مورد نظر تخصیص می یابند. هر مورچه مسیر خود را بر اساس غلظت فرمون ترشح شده و اطلاعات اکتشافی موجود در پیوندها انتخاب می کند. در ادامه مسیرهای پیموده شده توسط مورچه ها ارزیابی شده و اطلاعات فرمون در هر لبه با توجه به میزان مسیری که وجود دارد سازماندهی مجدد می شود. فرمون هر پیوند به عنوان امتیاز نهایی شباهت بین گره ها مورد استفاده قرار می گیرد. نتایج تجربی بر روی تعدادی از شبکه های واقعی نشان می دهد که استفاده از الگوریتم ACO دقت پیش بینی را بهبود داده و در کنار کاهش پیچیدگی، زمان حل مساله را تقلیل می دهد [۱۸].

به طور کلی، اگر بتوان با دو چالش دقت پیش بینی و مقیاس پذیری^۹ در یک شبکه گراف مقابله کرد، پیش بینی پیوند به ابزاری قدرتمند برای کاربرد در دامنه های مختلف تبدیل می شود. این مطالعه با هدف یافتن راه حل ممکن برای مقابله با این مسائل در بستر شبکه های اجتماعی انجام شده است. برای این منظور، سه الگوریتم بهینه سازی جغرافیای زیستی (BBO)^{۱۰}، ازدحام ذرات (PSO)^{۱۱} و الگوریتم ژنتیک (GA)^{۱۲} به منظور پیش بینی پیوند با استفاده از دو مجموعه داده واقعی که از مراجع استخراج شده اند، به کار گرفته شدند. الگوریتم های مورد بررسی، الگوریتم های بهینه سازی فراابتکاری الهام گرفته از طبیعت هستند که در زمینه های مختلف مهندسی و اقتصادی به کار گرفته میشوند که نتایج مناسبی از عملکرد و کارآمدی را نشان داده اند. در پایان این مطالعه به منظور انتخاب یک روش قدرتمندتر، مقایسه ای بین الگوریتم های پیشنهادی انجام گرفت و نتایج بدست آمده از عملکرد این الگوریتم ها به همراه پیشنهاداتی برای مطالعات آینده در این زمینه ارائه گردید.

¹ Supervised learning

² Binary classification mission

³ Time stamps

⁴ Matrix factorization

⁵ Nature-inspired based

⁶ Evolutionary algorithms

⁷ Ant colony optimization (ACO)

⁸ Swarm intelligence

⁹ Scalability

¹⁰ Biogeography based optimization (BBO)

¹¹ Particle swarm optimization (PSO)

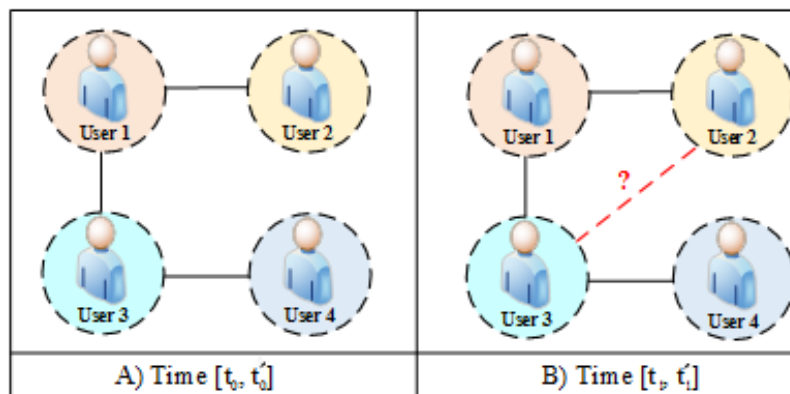
¹² Genetic algorithm (GA)

۲- روش شناسی تحقیق

۲-۱- فرمول بندی مساله پیش بینی پیوند

به منظور بیان مساله پیش‌بینی پیوند میتوان با در نظر گرفتن یک گراف وزن دار نشده و بدون جهت $G = (V, E)$ که ساختار توپولوژیکی یک شبکه را نشان می‌دهد و در آن هر یال $e = (u, v) \in E$ نشان دهنده یک برهمکنش بین u و v میباشد که در زمان خاص $t(e)$ انجام شده است. برای دو زمان t و $t' > t$ ، عبارت $G[t', t]$ زیر گراف^۱ جی را نشان میدهد که شامل همه ی لبه‌ها با برجسب زمانی بین t و t' میباشد. ابتدا فرض میشود که برای زمانهای t_0, t_1, t'_0, t'_1 شرط $t'_1 > t_1 > t'_0 > t_0$ برقرار میباشد. سپس وظیفه پیش‌بینی پیوند به این صورت است که برای شبکه $G[t_0, t'_0]$ ، لیستی از یال‌های موجود در $G[t_0, t'_0]$ را که به منظور ظاهر شدن در شبکه $G[t_1, t'_1]$ پیش‌بینی میشوند، تولید کند. همچنین $G[t_0, t'_0]$ به بازه آموزش و $G[t_1, t'_1]$ به بازه آزمایش اشاره دارد [۷].

اگر پیوندهایی که برای استنتاج یک شبکه اجتماعی مشاهده شده اند را به طور تصادفی انتخاب کنیم، برای گراف $G = (V, E)$ تعداد پیوندهایی برابر با $V^2 - E$ وجود خواهد داشت که می‌توان از بین آنها برخی را انتخاب نمود. تعداد پیوندهای E برابر است با $V^2 - d$ ، که d برای یک گراف متراکم مقداری ثابت بین ۱ و V میباشد. بنابراین، تعداد پیوندهای ثابتی برای انتخاب در اختیار بوده و احتمال انتخاب تصادفی مناسب برابر $O(1/c)$ میباشد. از سوی دیگر، اگر گراف G پراکنده باشد، تعداد پیوندهای E برابر با V خواهد بود. بنابراین، تعداد V^2 پیوند برای انتخاب از میان آنها و احتمال $O(1/V^2)$ برای یک انتخاب مناسب به طور تصادفی در اختیار میباشد. متأسفانه، شبکه‌های اجتماعی در ماهیت خود پراکنده هستند، بنابراین انتخاب تصادفی ایده چندان مناسبی به نظر نمیرسد [۱]. در شکل ۱، شمایی از مساله پیش‌بینی پیوند نشان داده شده است.



شکل ۱- شمایی از مساله پیش‌بینی پیوند برای یک شبکه اجتماعی در زمان $[t_0, t'_0]$ و $[t_1, t'_1]$ با پیوندهای پیش‌بینی شده (خط قرمز نشان دهنده پیوند پیش‌بینی شده می‌باشد) [۱۹]

۲-۲- الگوریتم بهینه سازی جغرافیای زیستی (BBO)

مانند هر الگوریتم تکاملی دیگر، در الگوریتم جغرافیای زیستی (BBO) نیز ابتدا یک مسئله بهینه سازی از نوع جمعیتی و از طریق راه حل‌های کاندید مطرح میشود. هر راه حل، از ویژگی‌ها یا متغیرهای مستقلی تشکیل شده است [۲۰]. در این الگوریتم، یک راه حل خوب همواره مربوط به یک زیستگاه بیولوژیکی است که برای زندگی مناسب بوده و راه حل‌های ضعیف مربوط به زیستگاهی است که برای زندگی مناسب نیستند. شاخص مناسب بودن سکونتگاه^۲ (HSI)، در این الگوریتم بیان‌کننده میزان مهاجرت دهی^۳ یا مهاجرت پذیری^۴ یک سکونتگاه را بیان میکند، به طوری که سکونتگاه‌های با شاخص HSI بالا دارای تناسب بالایی در بین

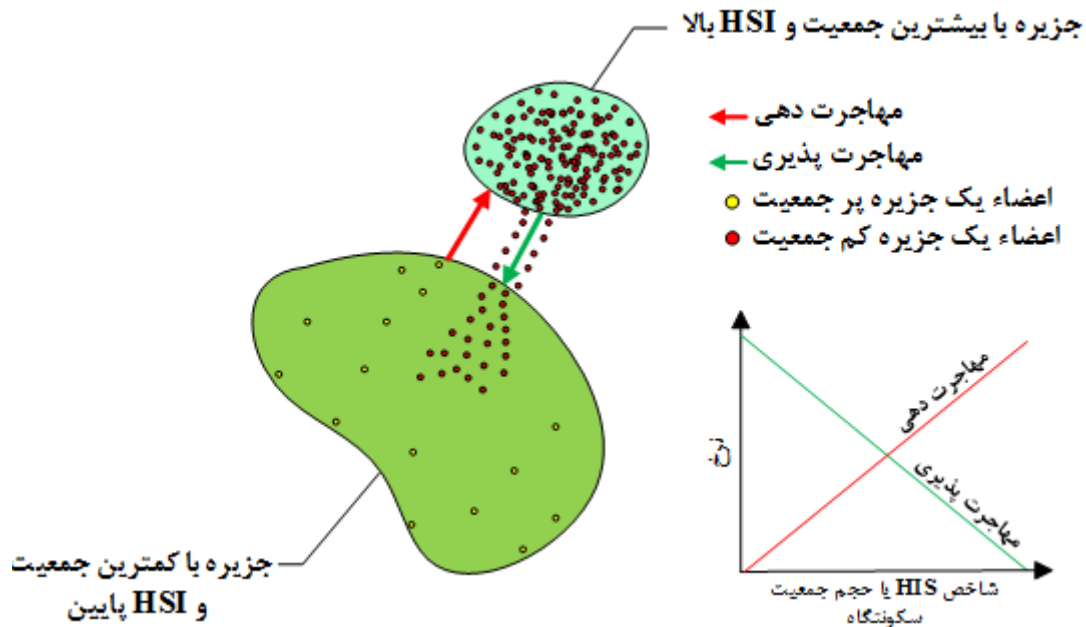
¹ Sub-graph

² Suitability Index Variable (SIV)

³ Emigration

⁴ Immigration

سکونتگاهها هستند و همین مساله سبب می شود تا همواره چنین سکونتگاههایی به دلیل جمعیت بالا، دارای مهاجرت پذیری پایین و مهاجرت دهی بالا باشند. در چنین حالتی، تمایل به مهاجرت از راه حل های با تناسب بالا به راه حل های کم تناسب و همچنین پذیرش ویژگی های مشترک در راه حل های کم تناسب افزایش میابد [۲۱]. در این الگوریتم عملگر های مهاجرت و جهش^۱ وظیفه اصلی اشتراک گذاری راه حل ها و ایجاد تنوع در آنها را بر عهده دارد و از این طریق الگوریتم را به پاسخ بهینه همگرا میکند. در شکل ۲، شمایی از ارتباط میان جمعیت زیستگاهها و میزان مهاجرت آنها نمایش داده شده است [۲۳]، [۲۲].



شکل ۲- شمایی از نسبت میان جمعیت زیستگاهها و میزان مهاجرت در آنها (جزیره با بیشترین جمعیت، همواره دارای شاخص HSI بالا بوده که همین مساله سبب افزایش نرخ مهاجرت دهی و کاهش نرخ مهاجرت پذیری در آن شده است، در مقابل جزیره با کمترین جمعیت دارای شاخص HSI پایین بوده و نرخ مهاجرت پذیری در آن بیشتر از نرخ مهاجرت دهی می باشد)

در الگوریتم BBO، اطلاعات بین راه حلها یا زیستگاههای مختلف از طریق دو متغیر نرخ مهاجرت پذیری (λ) و نرخ مهاجرت دهی (μ) اشتراک گذاری میشوند. نرخ مهاجرت هر راه حل به منظور به اشتراک گذاری یا تصمیم گیری تصادفی در خصوص مهاجرت یا عدم مهاجرت هر یک از ویژگی ها، در یک زمان مشخص مورد استفاده قرار میگیرد. همچنین، جهش یک تابع احتمالی است که می تواند ویژگی راه حل ها را تغییر داده و مانند هر الگوریتم فراابتکاری دیگر بر روی راه حل های مساله اعمال شود. هدف از جهش افزایش تنوع در بین جمعیت است [۲۴]. روابط مربوط به این متغیر ها به صورت زیر می باشد:

$$\lambda_i = I \left(1 - \frac{k(i)}{n} \right) \quad (1)$$

$$\mu_i = E \left(\frac{k(i)}{n} \right) \quad (2)$$

$$\lambda_i + \mu_i = E \quad (3)$$

همچنین در زیر شبه کد^۲ الگوریتم BBO ارائه شده است [۲۵]:

¹ Mutation

² Pseudocode

BBO Algorithm Pseudocode:

For each solution y_k set emigration μ_k proportional to

fitness with $\mu_k \in [0, 1]$.

for each solution y_k ($k = 1$ to N)

For each solution feature index s

Use $\{\mu_i\}$ to select the emigrating solution, denoted as y_j .

$z_k(s) \leftarrow y_j(s)$

End if

Next splution feature

Decide whethere to mutate z_k .

Next solution

$y \leftarrow z$.

که بر این اساس، امکان مهاجرت به جزیره z_k را می توان بوسیله شبه کد زیر تعریف نمود، که در آن $rand(0,1)$ یک عدد تصادفی است که به طور یکنواخت بین ۰ و ۱ توزیع میشود [۲۶]، [۲۳].

If $\lambda_k < rand(0,1)$

Migrate

Else

Do not migrate

۲-۳- الگوریتم ازدحام ذرات (PSO)

الگوریتم بهینه سازی ازدحام ذرات (PSO) اولین بار توسط Eberhart and Kennedy, (1995) [۲۷] معرفی شد. در سال های اخیر، از این الگوریتم به دلیل کارایی بالا در حل مسائل بهینه سازی پیچیده و سخت استفاده های بسیاری شده است. از سوی دیگر، الگوریتم PSO در حل مسائل غیرخطی، غیر قابل تمایز و چندوجهی بسیار قوی و سریع عمل می کند [۲۸]. در این الگوریتم، نخست مجموعه ای از راه حل های کاندید در یک فضای جستجو وجود می آیند و در ادامه با اندازه گیری مکرر تناسب این راه حل ها توسط تابع هدف بهینه شده، هر ذره به عنوان یک راه حل نامزد برای مسئله بهینه سازی در نظر گرفته میشود [۲۹]. در گام نخست، الگوریتم راه حل های کاندید را به طور تصادفی در یک فضای جستجو انتخاب می کند، به طوری که تمام راه حل های ممکن در ایجاد آن نقش داشته و هیچ گونه دانش قبلی از تابع هدف وجود ندارد. در ادامه، فرایند جستجو برای تعیین این که کدام راه حل نزدیک به حداکثر محلی یا جهانی است انجام میگیرد. با در نظر گرفتن x_i^k و v_i^k به عنوان موقعیت و سرعت ذره i در تکرار k ، می توان موقعیت هر ذره در تکرار بعدی را توسط رابطه زیر تعیین نمود:

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (4)$$

موقعیت ذرات x_i^{k+1} نسخه به روز شده از آخرین و بهترین موقعیت ذره به اضافه سرعت (شتاب) خود آن ذره می باشد. متغیر سرعت^۱، بر اساس سه پارامتر سرعت ذخیره شده قبلی، c_1r_1 و c_2r_2 تخمین زده می شود. از طریق سرعت ذرات، فرآیند بهینه سازی هدایت میشود که با استفاده از معادله زیر به روزسانی می گردد:

$$v_i^{k+1} = x \times [v_i^k + c_1r_1(p_i^k - x_i^k) + c_2r_2(g_i^k - x_i^k)] \quad (5)$$

که بر اساس رابطه فوق خواهیم داشت:

$$x = 2\eta / \left| 2 - \varphi - \sqrt{\varphi \times (\varphi - 4)} \right| \quad (6)$$

¹ Velocity

$$x \in [0,1]$$

$$\eta \in [0,1]$$

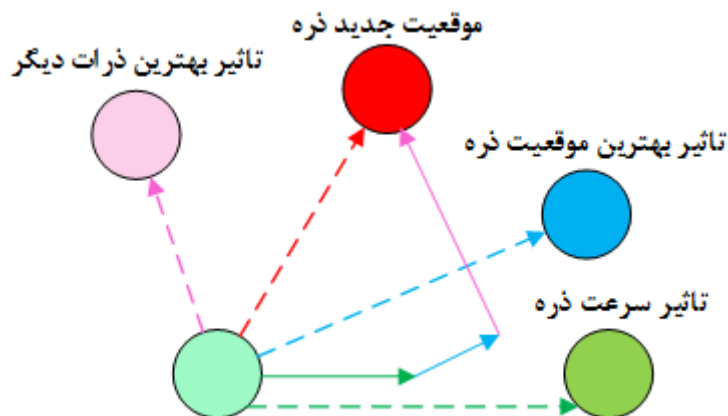
$$\phi_1 = c_1 r_1$$

$$\phi_2 = c_2 r_2$$

$$\phi = \phi_1 + \phi_2.$$

که در این روابط، بردارهای p_i^k و g_i^k به ترتیب $pbest$ و $gbest$ ذره i در تکرار k میباشد. ثابت‌های شتاب c_1 و c_2 تأثیر نسبی مکان‌های $pbest$ و $gbest$ را بر سرعت یک ذره کنترل می‌کنند. مقادیر کوچک برای c_1 و c_2 به هر ذره اجازه می‌دهد تا نقاط خوبی را که قبلاً کشف نشده اند، کشف کرده و مقادیر بزرگ از آنها، جستجوی فشرده تری را در مناطق نزدیک افزایش می‌دهد. همچنین، فاکتورهای اجتماعی r_1 و r_2 سبب ایجاد یک ماهیت تصادفی در الگوریتم می‌شود. علاوه بر این، ضریب وزن اینرسی (x) برای کنترل توانایی جستجوی حداقل‌های جهانی یا محلی^۲ مورد استفاده قرار میگیرد. مقادیر بزرگتر و کوچکتر از x به ترتیب باعث افزایش توانایی جستجوی جهانی و محلی شده و از سوی دیگر سبب میشود تا تأثیر سرعت قبلی ذرات بر سرعت فعلی آن‌ها کنترل شود. ضریب η ، متغیر دیگری است که وظیفه کنترل بهره برداری اکتشافی را در الگوریتم بر عهده دارد. با مقداری برابر صفر از این متغیر، نقطه حداقل جهانی سریعتر کشف شده و با مقداری برابر با ۱، درجه بالایی از اکتشاف با سرعت کمتر انجام می‌شود. همانطور که قبلاً ذکر شد، هر ذره دارای یک حافظه میباشد که به آن ذره کمک می‌کند موقعیت بعدی خود را به صورت بهتری در فضای جستجو دنبال کند. برای اینکه ذره به بهترین منطقه خود در فضا حرکت کند، از بهترین جزء گروه اجتماعی خود نیز استفاده می‌کند.

شکل ۳، چگونگی بروز رسانی موقعیت یک ذره را بر اساس بردارهای سرعت، حافظه و همچنین موقعیت سایر ذرات نمایش می‌دهد. در این فرایند، الگوریتم ابتدا موقعیت ذرات را مقداردهی اولیه می‌کند و سپس برنامه را هر چند بار که لازم است اجرا می‌کند تا بردار سرعت هر ذره محاسبه شود. هنگامی که سرعت برای هر ذره محاسبه شد، موقعیت هر ذره با اعمال سرعت آن، بهترین موقعیت در تکرارهای قبلی و بهترین موقعیت بدست آمده توسط سایر ذرات به روزرسانی می‌شود. این روند تا زمانی که شرایط توقف برآورده شود تکرار می‌شود [۳۱]، [۳۰].



شکل ۳- چگونگی به روزرسانی موقعیت ذرات در الگوریتم PSO

همچنین در زیر شبه کد الگوریتم PSO ارائه شده است [۳۲]:

¹ Social factors

² Global and local minimum

PSO Algorithm Pseudocode:

```

Initialize number of particle
Initialize each particle position and velocity randomly
Initialize  $c_1, c_2, r_1, r_2, k, k_{max}$ , and stop condition
While (not stop condition)
  For (all particle position)
    For (all input training patterns)
      apply input training pattern
      feed forward
      update fitness value  $epsom$ 
    End for
    IF ( $epsom \leq pbest$ )
      update  $pbest$  value
      update  $pbest$  vector
    End if
  End for
  For (all particle position)
    update  $gbest$  to the minimum  $pbest$  value
    update  $gbest$  vector
    updat particle velocity vector  $v_i^k$ 
    updat particle position vector  $x_i^k$ 
  End for
End while

```

۲-۴- الگوریتم ژنتیک (GA)

الگوریتم ژنتیک (GA) یک تکنیک بهینه سازی تکاملی بر اساس ایده های ژنتیک و انتخاب طبیعی^۱ است [۳۳]. این الگوریتم به طور ثابت شده ای عملکرد مناسبی در حل مسائل پیچیده یادگیری ماشین (ML)^۲ و بهینه سازی تولید دارا میباشد. در الگوریتم GA، ابتدا جمعیت اولیه ای از پاسخها مقاردهی اولیه می شوند. در ادامه، میزان تناسب پاسخ های اولیه مورد بررسی قرار گرفته و سه عملگر انتخاب^۳، تقاطع^۴ و جهش بر روی پاسخ ها اعمال میشود. دو عملگر تقاطع و جهش، به ترتیب برای اطمینان از همگرایی و اجتناب از بهینه محلی به کار می روند [۳۵]، [۳۴]. در ادامه فرایند، بهترین (مناسب ترین) راه حل از جمعیت به عنوان نسل اول انتخاب میشود. سپس، پاسخهایی که به عنوان والد انتخاب شده بودند با یکدیگر ترکیب می شوند تا از این طریق فرزندان با صفات والدینشان تولید شود. این فرآیند تا زمانی که الگوریتم به شرایط خاتمه یا نسلی از بهترین پاسخها دست یابد، تکرار میشود [۳۶]، [۳۵]. در شکل ۴، شمایی از فرایند عملکرد الگوریتم GA نمایش داده شده است.

¹ Natural selection² Machine learning (ML)³ Selection⁴ Crossover



شکل ۴- شمایی از فرایند عملکرد الگوریتم GA

همچنین در ادامه، شبه کد الگوریتم GA ارائه شده است [۳۷]:

GA Algorithm Pseudocode:

```

t = 0
Initialize(P(t=0))
Evaluate(P(t=0))
While is Not Terminated () do
Pp(t) = P(t).selectParents()
Pc(t) = Reproduction(Pp)
Mutate (Pc(t))
Evaluate (Pc(t))
P(t+1)=building Next Generation Form(Pc(t), P(t))
t=t+1
End

```

۳- پایگاه داده مورد استفاده

جدول ۱، مجموعه داده های مورد استفاده در این مطالعه را به همراه جزئیاتی در خصوص تعداد گره ها و لبه ها ارائه می دهد. مجموعه داده های به کار گرفته شده در این مطالعه شامل اطلاعات مربوط به شبکه هم تالیفی میان دانشمندان^۱ (NetScience) و شبکه تعامل پروتئین^۲ (Yeast) می باشد که از ادبیات تحقیق جمع اوری شده است [۳۸].

جدول ۱- جزئیات مجموعه داده های مورد استفاده در این مطالعه شامل تعداد گره ها و لبه ها

تعداد لبه ها	تعداد گره ها	نام مجموعه داده
۲۷۴۲	۱۵۸۹	شبکه هم تالیفی میان دانشمندان (NetScience)
۶۱۳۶	۲۳۷۵	شبکه تعامل پروتئین (Yeast)

۴- پیش بینی پیوند با استفاده از الگوریتم های فرا ابتکاری

به منظور مدیریت و رفع مشکل ابعاد شبکه های بزرگ، شبکه های مورد نظر به تعدادی جوامع کوچکتر تقسیم شدند. برای این منظور می توان جوامع کوچکتر را با استفاده از الگوریتم بهینه سازی مدولاریته حریص سریع^۳ که بر اساس الگوریتم چند مقیاسی

¹ Co-authorships network between scientists

² Network or protein interaction

³ Fast greedy modularity optimization algorithm

عمل می کند، شناسایی نمود [۳۹]. در برخی از مطالعات انجام شده در این زمینه مانند [۳۹-۴۰]، مدولاریت مجموع تفاوت میان بخشی از پیوندهای داخل یک پارتیشن با مقدار مورد انتظار این پیوندها تعریف شده است که توسط آن به یک پارتیشن دیگر پیوند داده شده است. بنابراین با قرار دادن تصادفی یالها به رابطه زیر خواهیم رسید:

$$Q_m = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \sigma(i, j) \quad (7)$$

که در این رابطه، A ماتریس مجاورت^۱، m تعداد یال ها یا قدرت کل برای یک شبکه وزنی، d_i درجه (یا قدرت) گره i ام می باشد. همچنین، تابع $\sigma(i, j)$ عدد یک را برمی گرداند اگر گره های i و j متعلق به یک جامعه باشند و در غیر این صورت این تابع برابر صفر خواهد بود.

یک شبکه بدون جهت، تا جای ممکن به جوامعی تقسیم می شود که بیشتر شبکه را پر می کند. در نتیجه، کل شبکه به تعداد اجزای متصل به دست آمده از شبکه قبل تقسیم می شود. این تقسیمات کل گراف سبب می شود تا شبکه به تعدادی زیرگراف تقسیم شود تا از این طریق بتوان به بررسی هر زیرگراف به طور جداگانه پرداخت.

به منظور پیاده سازی طراحی مجزا در گراف، هر زیرگراف توسط یک ماتریس مجاورت نشان داده می شود. بر اساس این ماتریس، تعدادی از واحد های مستقل به صورت تصادفی تولید می شوند که ماتریس های صفر و یک میباشند. با توجه به اینکه در واقعیت نمی توان آینده را دید، برای آزمایش درستی الگوریتم، ۹۰ درصد کل لبه ها از مجموعه داده مورد نظر جدا گردید تا به عنوان مجموعه آموزش^۲ در نظر گرفته شود. همچنین از ۱۰ درصد لبه های باقی مانده به عنوان مجموعه آزمایش^۳ و به منظور ارزیابی عملکرد روش ها توسعه داده شده در پیش بینی لینک مورد استفاده قرار گرفت. مجموعه لبه های باقیمانده در مجموعه تست که ۱۰ درصد کل لبه ها را تشکیل میدهد، مجاز به استفاده در فرایند آموزش نیستند.

به دلیل ماهیت معمولاً پراکنده شبکه های اجتماعی، مجموعه داده های لبه ها در یک شبکه معین نامتعادل می باشد. به منظور افزایش دقت در هنگام ارزیابی کیفیت الگوریتم، از یک معیار عملکرد به نام میانگین هندسی^۴ استفاده میشود که بیشترین تمرکز را بر روی کلاس اقلیت^۵ (کلاسهایی با داده های کمتر) دارد. میانگین هندسی به عنوان ریشه n ام حاصل ضرب n عدد تعریف می شود. برای مسئله پیش بینی پیوند، میانگین هندسی مطابق با رابطه زیر تعریف می شود:

$$G-Mean = \sqrt{TPR \times TNR} \quad (8)$$

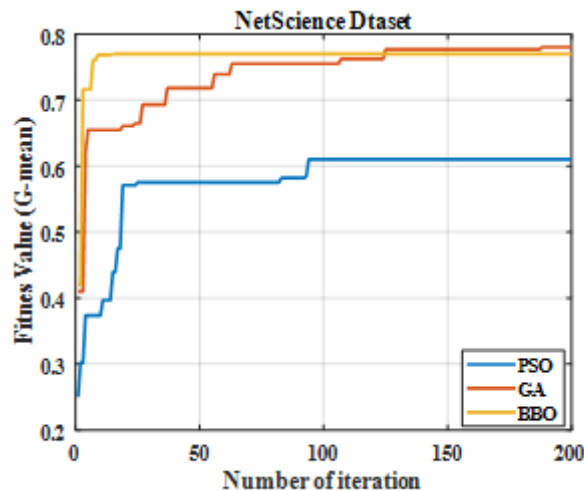
که در این رابطه TPR نرخ مثبت واقعی^۶ و TNR نرخ منفی واقعی^۷ می باشد. یکی از مهمترین بخش ها در پایه ریزی یک مساله بهینه سازی، تابع تناسب میباشد. تابع تناسب، مقایسه ای را بین بخش واقعی مجموعه آموزشی یک گراف و بخش پیش بینی شده آن انجام می دهد. در این راستا، مقدار میانگین جی^۸ (G-Mean) محاسبه شده، با بهترین مقدار بدست آمده در یک تکرار مشخص از الگوریتم مقایسه می شود. سپس، هر زیرگراف بهترین مقدار را برای تابع تناسب خود دریافت کرده و مجموع حاصلضرب ضریب وزنی اختصاص داده شده به هر زیرگراف که بر اساس اندازه آنها تعیین شده است، محاسبه می گردد. در هر تکرار از الگوریتم های بهینه سازی، این مقادیر بدست آمده مرتباً به روزسانی می شوند تا مقدار تناسب آنها در جهت حصول شرایط توقف مساله محاسبه شود [۱].

¹ Adjacency matrix
² Training set
³ Testing set
⁴ Geometric mean
⁵ Minority class
⁶ True positive rate
⁷ True negative rat
⁸ G-mean

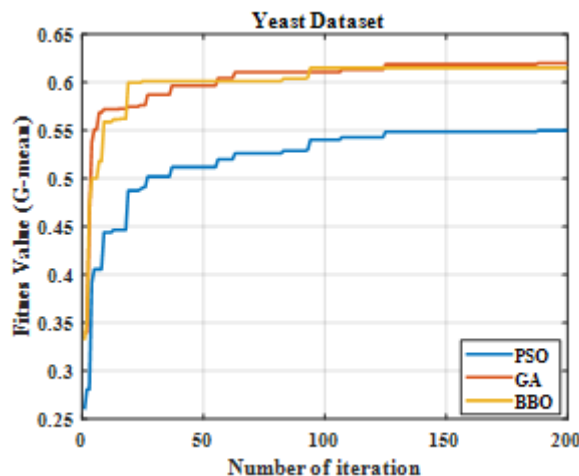
۵- بحث و نتیجه گیری

سه الگوریتم مورد بررسی در این مطالعه، با استفاده از دو مجموعه داده واقعی که از ادبیات تحقیق جمع آوری شده بودند، پیاده سازی و اجرا شدند. سخت افزار سیستم مورد استفاده به منظور توسعه الگوریتم های مورد بررسی، دارای پردازنده Intel-core™ i7-6700K با ۴ گیگابایت رم و سیستم عامل ۶۴ بیتی ویندوز ۸ میباشد. کد نویسی و پیاده سازی الگوریتم ها در محیط نرم افزار MATLAB R2021b انجام گرفت. به منظور بررسی دقیقتر الگوریتم ها، هر مجموعه داده ۵ بار مورد آزمایش قرار گرفت و در انتها میانگین بهترین مقدار G-Mean محاسبه گردید. هر اجرا دارای ۲۰۰ تکرار و ۵۰ عامل جستجو (اندازه جمعیت) است.

شکل های ۵ و ۶ به ترتیب بهترین منحنی های همگرایی را برای مجموعه داده های NetScience و Yeast نشان می دهند. شکل ۵ نشان می دهد که الگوریتم BBO سریعتر از دو الگوریتم GA و PSO برای مجموعه داده NetScience همگرا می شود. با این وجود، با اختلاف ناچیزی الگوریتم GA دارای مقدار G-mean نسبتا بالاتری در مقایسه با الگوریتم های BBO و GA میباشد. علاوه بر این، می توان از شکل ۶ مشاهده نمود که سرعت همگرایی الگوریتم GA برای مجموعه داده های Yeast نسبت به دو الگوریتم دیگر بیشتر می باشد. اگرچه BBO به سرعت الگوریتم ژنتیک همگرا نمی شود، اما به یک راه حل بهینه نزدیک به الگوریتم GA می رسد. با اینحال، الگوریتم PSO کندترین رفتار همگرایی و کمترین مقدار G-mean را به دست آورد. در جدول ۲، نتایج تجربی استفاده از الگوریتم های فرا ابتکاری بر روی دو مجموعه داده NetScience و Yeast ارائه شده است.



شکل ۵- منحنی های همگرایی برای مجموعه داده های NetScience



شکل ۶- منحنی های همگرایی برای مجموعه داده های Yeast

جدول ۲- نتایج تجربی استفاده از الگوریتم های فرا ابتکاری بر روی دو مجموعه داده Yeast و NetScience.

نام مجموعه داده	متغیر تحلیلی	PSO	GA	BBO
شبکه هم تالیفی میان دانشمندان (NetScience)	میانگین	۰.۵۷۶	۰.۷۴۴	۰.۷۶۵
	بهترین پاسخ	۰.۶۱۵	۰.۷۸۲	۰.۷۷۱
	انحراف استاندارد	۰.۰۶۵	۰.۰۵۷	۰.۰۳۵
شبکه تعامل پروتئین (Yeast)	میانگین	۰.۵۲۲	۰.۶۰۳	۰.۶۰۱
	بهترین پاسخ	۰.۵۵۴	۰.۶۲۳	۰.۶۱۵
	انحراف استاندارد	۰.۰۴۴	۰.۰۳۲	۰.۰۳۴

یکی از مزیت های مهم الگوریتم ژنتیک این است که به صورت موازی جمعیتی از نقاط را برای یافتن بهترین پاسخ جستجو می کنند. در نتیجه، این قابلیت را دارد که از گیر افتادن در راه حل بهینه محلی مانند رویکردهای سنتی، دور بماند. علاوه بر این، الگوریتم ژنتیک از قواعد انتخاب احتمالی^۱ به جای قواعد قطعی^۲ استفاده می کند و به جای کار کردن بر روی خود پارامترها، بر روی کروموزوم ها کار می کند که باعث ایجاد یک عملکرد مطلوب در زمان بهینه سازی مسائل می شود. الگوریتم BBO نیز میتواند به عنوان یک روش تکاملی قدرتمند در حل مسائل بهینه سازی مورد استفاده قرار گیرد که توانایی بالایی در بدست آوردن جواب بهینه مطلق دارا میباشد. بنابر این، این ویژگی ها در کنار نتایج بدست آمده از این مطالعه، ثابت میکند که دو الگوریتم GA و BBO دارای عملکرد بسیار مناسبی در حل مسائل پیش بینی لینک دارا می باشند.

۶- نتیجه گیری

در این مقاله با استفاده از سه الگوریتم جغرافیای زیستی (BBO)، ازدحام ذرات (PSO) و الگوریتم ژنتیک (GA) به بررسی راه حلی بالقوه برای حل مساله پیش بینی پیوند پرداخته شد. برای این منظور از دو مجموعه داده شبکه هم تالیفی میان دانشمندان (NetScience) و شبکه تعامل پروتئین (Yeast) که از ادبیات حقیق جمع اوری شده بودند به منظور توسعه و ارزیابی روش های مورد مطالعه استفاده گردید. شبکه های موجود در این داده ها به تعداد کوچکتی از جوامع تقسیم شدند و هر گره به عنوان یک ماتریس مجاورت نشان داده شد. در ادامه، برای یافتن بهترین راه حل در پایان تکرارهای الگوریتم، متغیر G-mean به عنوان تابع هزینه هر گره اندازه گیری و مورد ارزیابی قرار گرفت. نتایج بدست آمده نشان داد که الگوریتم های BBO و GA دارای سرعت همگرایی و پاسخ های مناسب تری نسبت به الگوریتم PSO می باشند. با این وجود، الگوریتم GA با اختلاف بسیار ناچیزی نسبت به الگوریتم BBO نتایج بهتری را در حل مساله پیش بینی لینک ارائه داد. به عنوان یک پیشنهاد برای مطالعات آینده، می توان مقایسه جامع تری بین الگوریتم های پیشنهادی و سایر الگوریتم های تکاملی ترتیب داد تا از این طریق به یک روش دقیق تر و کارآمدتر در حل مسائل پیش بینی لینک دست یافت.

۷- مراجع

1. R. Barham and I. Aljarah, "Link prediction based on whale optimization algorithm," in 2017 International Conference on New Trends in Computing Sciences (ICTCS), 2017, pp. 55-60.
2. A.-S. Ghadeer, I. Aljarah, and H. Alsawalqah, "Enhancing the Arabic sentiment analysis using different preprocessing operators," New Trends Inf. Technol, vol. 113, pp. 113-117, 2017.
3. L. M. Qaisi and I. Aljarah, "A twitter sentiment analysis for cloud providers: a case study of Azure vs. AWS," in 2016 7th International Conference on Computer Science and Information Technology (CSIT), 2016, pp. 1-6.
4. S. E. Shukri, R. I. Yaghi, I. Aljarah, and H. Alsawalqah, "Twitter sentiment analysis: A case study in the automotive industry," in 2015 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT), 2015, pp. 1-5.
5. H. Safdari, M. Contisciani, and C. De Bacco, "Reciprocity, community detection, and link prediction in dynamic networks," J. Phys. Complex., vol. 3, no. 1, p. 15010, 2022.

¹ Probabilistic selection rules

² Deterministic ones

6. P. S. Somula and R. Devulapalli, "Adaptive local neighborhood range based firefly algorithm for link prediction," *Int. J. Syst. Assur. Eng. Manag.*, 2021, doi: 10.1007/s13198-021-01291-9.
7. D. Liben- Nowell and J. Kleinberg, "The link- prediction problem for social networks," *J. Am. Soc. Inf. Sci. Technol.*, vol. 58, no. 7, pp. 1019–1031, 2007.
8. T. Nadu, "Structural Similarity Based Link Prediction in Social Networks Using Firefly Algorithm," pp. 560–564, 2017.
9. S. Pulipati and M. Ramakrishnan, "Topological and Attribute Link Prediction using Firefly algorithm," pp. 33–41, 2020.
10. M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *SDM06: workshop on link analysis, counter-terrorism and security*, 2006, vol. 30, pp. 798–805.
11. A. Kumari, R. K. Behera, K. S. Sahoo, A. Nayyar, A. Kumar Luhach, and S. Prakash Sahoo, "Supervised link prediction using structured- based feature extraction in social network," *Concurr. Comput. Pract. Exp.*, vol. 34, no. 13, p. e5839, 2022.
12. R. Barham, A. Sharieh, and A. Sleit, "Moth Flame Optimization Based on Golden Section Search and its Application for Link Prediction Problem," vol. 13, no. 1, pp. 10–27, 2019, doi: 10.5539/mas.v13n1p10.
13. X. H. Shi, Y. C. Liang, H. P. Lee, C. Lu, and Q. X. Wang, "Particle swarm optimization-based algorithms for TSP and generalized TSP," *Inf. Process. Lett.*, vol. 103, no. 5, pp. 169–176, 2007.
14. M. Fire, L. Tenenboim, O. Lesser, R. Puzis, L. Rokach, and Y. Elovici, "Link prediction in social networks using computationally efficient topological features," in *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing*, 2011, pp. 73–80.
15. H. H. Song, T. W. Cho, V. Dave, Y. Zhang, and L. Qiu, "Scalable proximity estimation and link prediction in online social networks," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*, 2009, pp. 322–335.
16. T. Tylenda, R. Angelova, and S. Bedathur, "Towards time-aware link prediction in evolving social networks," in *Proceedings of the 3rd workshop on social network mining and analysis*, 2009, pp. 1–10.
17. E. F. G. Goldbarg, G. R. de Souza, and M. C. Goldbarg, "Particle swarm for the traveling salesman problem," in *European Conference on Evolutionary Computation in Combinatorial Optimization*, 2006, pp. 99–110.
18. B. Chen and L. Chen, "A link prediction algorithm based on ant colony optimization," *Appl. Intell.*, vol. 41, no. 3, pp. 694–708, 2014.
19. F. Gao, K. Musial, C. Cooper, and S. Tsoka, "Link prediction methods and their accuracy for different social networks and network metrics," *Sci. Program.*, vol. 2015, 2015.
20. D. Simon, "Biogeography-based optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 6, pp. 702–713, 2008.
21. B. Santosa and A. L. Safitri, "Biogeography-based Optimization (BBO) Algorithm for Single Machine Total Weighted Tardiness Problem (SMTWTP)," *Procedia Manuf.*, vol. 4, pp. 552–557, 2015, doi: 10.1016/j.promfg.2015.11.075.
22. H. Ma, D. Simon, S. Member, P. Siarry, Z. Yang, and M. Fei, "Biogeography-Based Optimization : A 10-Year Review," vol. 1, no. 5, pp. 391–407, 2017.
23. W. L. Mao, Suprpto, C. W. Hung, and T. W. Chang, "Nonlinear system identification using BBO-based multilayer perceptron network method," *Microsyst. Technol.*, pp. 1–10, Apr. 2019, doi: 10.1007/s00542-019-04415-1.
24. S. K. Gupta, L. Kumar, M. K. Kar, S. Kumar, and G. Singh, "Active and Reactive Load Ability Improvement using BBO Algorithm with FACTS Devices," no. February, 2020, doi: 10.35940/ijitee.D1408.029420.
25. H. M. J. Mustafa, M. Ayob, D. Albashish, and S. Abu-Taleb, "Solving text clustering problem using a memetic differential evolution algorithm," *PLoS One*, vol. 15, no. 6, p. e0232816, 2020.
26. N. Mondal, S. Das, T. Banerjee, and M. C. Mandal, "Experimental study on drilling burr formation minimization and parameters optimization using BBO algorithm," *Mater. Today Proc.*, vol. 56, pp. 2586–2590, 2022.
27. R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the sixth international symposium on micro machine and human science*, 1995, pp. 39–43.
28. H. B. Duan and S. Q. Liu, "Non-linear dual-mode receding horizon control for multiple unmanned air vehicles formation flight based on chaotic particle swarm optimisation," *IET Control theory Appl.*, vol. 4, no. 11, pp. 2565–2578, 2010.
29. Jang-Ho Seo, Chang-Hwan Im, Sang-Yeop Kwak, Cheol-Gyun Lee, and Hyun-Kyo Jung, "An Improved Particle Swarm Optimization Algorithm Mimicking Territorial Dispute Between Groups for Multimodal Function Optimization Problems," *IEEE Trans. Magn.*, vol. 44, no. 6, pp. 1046–1049, Jun. 2008, doi:

10.1109/TMAG.2007.914855.

30. H. Liang, J. Zou, Z. Li, M. J. Khan, and Y. Lu, "Dynamic evaluation of drilling leakage risk based on fuzzy theory and PSO-SVR algorithm," *Futur. Gener. Comput. Syst.*, vol. 95, pp. 454–466, Jun. 2019, doi: 10.1016/j.future.2018.12.068.
31. A. Karkevandi-Talkhooncheh, S. Hajirezaie, A. Hemmati-Sarapardeh, M. M. Husein, K. Karan, and M. Sharifi, "Application of adaptive neuro fuzzy interface system optimized with evolutionary algorithms for modeling CO₂-crude oil minimum miscibility pressure," *Fuel*, vol. 205, pp. 34–45, 2017.
32. M. G. H. Omran, A. P. Engelbrecht, and A. Salman, "Particle swarm optimization for pattern recognition and image processing," in *Swarm intelligence in data mining*, Springer, 2006, pp. 125–151.
33. J. H. Holland, *Adaptation in Natural and Artificial Systems*. The MIT Press, 1992.
34. A. Al-fugara, M. Ahmadlou, A. R. Al-shabeeb, H. Al-amoush, and R. Al-adamat, "Spatial mapping of groundwater springs potentiality using grid search-based and genetic algorithm-based support vector regression," *Geocarto Int.*, vol. 0, no. 0, pp. 1–20, 2020, doi: 10.1080/10106049.2020.1716396.
35. S. K. Roy and D. De, "Genetic algorithm based internet of precision agricultural things (IopaT) for agriculture 4.0," *Internet of Things*, vol. 18, p. 100201, 2022.
36. O. Kramer, *Genetic Algorithm Essentials*, vol. 679. Cham: Springer International Publishing, 2017.
37. M. Ketcham and S. Vongpradhip, "Intelligent audio watermarking using genetic algorithm in DWT domain," *Int. J. Comput. Inf. Eng.*, vol. 1, no. 2, pp. 377–382, 2007.
38. L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, and T. Zhou, "Vital nodes identification in complex networks," *Phys. Rep.*, vol. 650, pp. 1–63, 2016.
39. E. Le Martelot and C. Hankin, "Multi-scale community detection using stability optimisation within greedy algorithms," *arXiv Prepr. arXiv1201.3307*, 2012.
40. N. Chiluka, N. Andrade, and J. Pouwelse, "A link prediction approach to recommendations in large-scale user-generated content systems," in *European Conference on Information Retrieval*, 2011, pp. 189–200.