

A novel metaheuristic algorithm and its discrete form for influence maximizing in complex networks

Vahideh Sahargahi^{*}, Vahid Majidnezhad^{**}, Saeid Taghavi Afshord^{**}, Yasser Jafari^{**}

^{*}Department of Computer Engineering, Shabestar Branch, Islamic Azad University, Shabestar, Iran

^{**}Assistant Professor, Department of Computer Engineering, Shabestar branch, Islamic Azad University, Shabestar, Iran

Abstract

In light of the No Free Lunch (NFL) theorem, which establishes the inherent limitations of meta-heuristic algorithms in universally efficient problem solving, the ongoing quest for enhanced diversity and efficiency prompts the introduction of novel algorithms each year. This research presents the IWOOGSA meta-heuristic algorithm, a pioneering solution tailored for addressing continuous optimization challenges. IWOOGSA ingeniously amalgamates principles from both the invasive weed optimization algorithm and the gravitational search algorithm, capitalizing on their synergies. The algorithm's key innovation lies in its dual-pronged sample generation strategy: a subset of samples follows a normal distribution, while others emulate the planetary motion-inspired velocities and accelerations from the gravitational search algorithm. Furthermore, a selective transfer of certain samples from distinct classes contributes to the evolution of succeeding generations. Expanding upon this foundation, a discrete variant of IWOOGSA, termed DIWOOGSA, emerges to tackle discrete optimization problems. The efficacy of DIWOOGSA is demonstrated through its application to the intricate influence maximization problem. DIWOOGSA distinguishes itself with an astute population initialization strategy and the integration of a local search operator to expedite convergence. Empirical validation encompasses a rigorous assessment of IWOOGSA against established benchmark functions, composite functions, and real-world engineering structural design problems. Remarkably, the IWOOGSA algorithm asserts its superiority, eclipsing both contemporary and traditional methods. This ascendancy is statistically affirmed through the utilization of the Friedman test rank, positioning IWOOGSA as the premier choice. Also, DIWOOGSA algorithm is evaluated by considering different networks for influence maximization problem, and it shows acceptable results in terms of influence and computational time in comparison to conventional algorithms.

Keywords: IWOOGSA, Invasive weed optimization algorithm, Gravitational search algorithm

یک الگوریتم فراابتکاری پیوسته جدید و گسسته‌سازی آن جهت بهینه‌سازی نفوذ در شبکه‌های پیچیده

وحیده سحرگاهی*، وحید مجیدنژاد**، سعید تقوی افشرد*، یاسر جعفری**

* دکتر، گروه مهندسی کامپیوتر، واحد شبستر، دانشگاه آزاد اسلامی، شبستر، ایران

** استادیار، گروه مهندسی کامپیوتر، واحد شبستر، دانشگاه آزاد اسلامی، شبستر، ایران

تاریخ پذیرش: ۱۴۰۲/۰۲/۱۷

تاریخ دریافت: ۱۴۰۱/۰۵/۲۷

نوع مقاله: پژوهشی

چکیده

طبق نظریه ناهار مجانی (NFL) هیچ الگوریتم فرا اکتشافی موجود قادر به حل همه نوع مسائل به طور کارآمد نیست، بنابراین هر ساله الگوریتم‌های جدیدی جهت تنوع بخشی پیشنهاد می‌شوند. در این مقاله، الگوریتم فراابتکاری جدیدی به نام IWOOGSA، برای مسائل بهینه‌سازی پیوسته پیشنهاد شده است که ترکیبی از الگوریتم‌های بهینه‌سازی علف‌های هرز و جستجوی گرانشی است. در IWOOGSA والد‌ها به دو صورت تکثیر می‌شوند و از هر دسته نمونه‌هایی برای انتقال به نسل جدید انتخاب می‌گردد. بخشی از تکثیر با توزیع نرمال صورت می‌گیرد و بخشی دیگر بر مبنای روابط سرعت و شتاب حرکت سیارات در الگوریتم جستجوی گرانشی انجام می‌شوند. یک مدل گسسته جدید از IWOOGSA به نام DIWOOGSA برای حل مسأله‌های بهینه‌سازی گسسته پیشنهاد شده است و کارایی آن بر روی یک چالش حیاتی تحت عنوان بهینه‌سازی نفوذ ارزیابی شده است. در DIWOOGSA از رویکرد هوشمندانه‌ای برای مقداردهی اولیه جمعیت استفاده شده و برای همگرایی سریعتر الگوریتم، یک عملگر جستجوی محلی پیشنهاد شده است. در حالت پیوسته الگوریتم IWOOGSA با توابع بنچمارک استاندارد و کامپوزیت و ۳ مسأله مهندسی رایج ارزیابی شده است. نتایج پیاده‌سازی ثابت می‌کند که الگوریتم IWOOGSA در مقایسه با روش‌های اخیر و متداول بسیار رقابتی بوده و با توجه به نتایج رتبه‌بندی آزمون فریدمن، توانسته است رتبه اول را کسب نماید. در حالت گسسته نیز الگوریتم DIWOOGSA با در نظر گرفتن شبکه‌های مختلف ارتباطاتی بین محققان برای مسأله بهینه‌سازی نفوذ مورد ارزیابی قرار گرفته و در مقایسه با الگوریتم‌های رایج در این زمینه از نظر میزان نفوذ و زمان اجرا نتایج قابل قبولی را کسب کرده است.

کلمه‌های کلیدی: الگوریتم بهینه‌سازی علف‌های هرز، الگوریتم‌های جستجوی گرانشی، الگوریتم IWOOGSA

CSA^۵ [5]، الگوریتم بهینه‌سازی آزمایشی دو شکاف یانگ (YDSE) [۶]، الگوریتم بهینه‌سازی کوآتی (COA) [۷] و الگوریتم سینوسی کسینوس تقویت شده با بهره برداری (EBSCA) [۸] توسط محققان مختلف پیشنهاد شده‌اند. انواع الگوریتم‌های فراابتکاری به دسته‌های الگوریتم‌های تکاملی [9-12]، الگوریتم‌های شبیه‌سازی ساختارهای فیزیکی [13] [41]، الگوریتم‌های هوش جمعی [14-18] و الگوریتم‌های مبتنی بر رفتار انسان [19-22] تقسیم‌بندی می‌شوند [13]. اکثر این الگوریتم‌ها در کنار عملکرد مناسب خود دارای معیاب نیز می‌باشند. زمان اجرای طولانی، نرسیدن به بهینه سراسری در برخی از اوقات و سرعت همگرایی پایین از معیاب الگوریتم زنتیک از دسته الگوریتم‌های تکاملی است [۲۰]. از معیاب الگوریتم وال [۲۱] از دسته الگوریتم‌های هوش جمعی می‌توان به افتادن در بهینه محلی، سرعت همگرایی و

مسائل پیچیده، اعم از مسائل مهندسی و یا مسائلی همانند زمانبندی کارها و بهینه‌سازی صرفاً توسط دانش و روش‌های ریاضی و الگوریتم‌های متداول قابلیت حل ندارند و یا زمان بالایی برای حل این مسائل نیاز می‌باشد. در این بین علم هوش مصنوعی دارای ابزارهای مناسب جهت حل این مسائل می‌باشد. یکی از ابزارهای کارآمد برای حل مسائل بهینه‌سازی در علم هوش مصنوعی الگوریتم‌های فراابتکاری هستند. در سال‌های اخیر الگوریتم‌های فراابتکاری بسیاری از جمله الگوریتم گرده افشانی گل^۱ CMFPA [1]، الگوریتم کرم شب تاب ترکیبی با جاذبه گروه بندی HFA- GA^۲ [2]، الگوریتم سه درختی^۳ TriTSA [3]، الگوریتم جستجوی تعاملی بیزی^۴ BISA [4]، الگوریتم جستجوی همکاری

^۱ flower pollination algorithm

^۲ Hybrid firefly algorithm with grouping attraction

^۳ Triple Tree-Seed Algorithm

^۴ Bayesian Interactive Search Algorithm

^۵ Cooperation search algorithm

طبق نظریه نهار مجانی (NFL^۶) [۳۴] منطقاً اثبات شده است که هیچ فرابتنکاری برای حل تمام مسائل بهینه‌سازی صحیح جواب نمی‌دهد. به عبارت دیگر، یک فرابتنکار بخصوص، ممکن است نتایج امیدبخشی را برای حل یک سری از مسائل نشان دهد، اما همین الگوریتم ممکن است کارآیی ضعیفی را برای تعداد دیگری از مسائل نشان دهد. بنابراین، هر ساله محققان در زمینه بهینه‌سازی در تلاش هستند تا تکنیک‌های بهینه‌سازی بهتری را ایجاد کنند.

زمان اجرا طولانی تر، مشکل همگرایی در صورت تولید نامناسب جمعیت اولیه از معایب الگوریتم GSA است [۳۷]. از سوی دیگر، از معایب IWO می‌توان به دقت حل پایین و گیر کردن در بهینه‌های محلی و همگرایی اولیه اشاره کرد [۳۲]. در سال‌های اخیر، بسیاری از محققان، سعی در ترکیب الگوریتم‌ها برای غلبه بر کاستی‌های فردی آن‌ها دارند. در این مقاله سعی کرده ایم از قابلیت‌های الگوریتم GSA، در بهبود عملکرد الگوریتم علف‌های هرز جهت حل مسائل بهینه‌سازی استفاده نماییم. در روش ترکیبی پیشنهادی IWOGSA برای تکثیر والد‌ها علاوه بر توزیع نرمال از روابط سرعت و شتاب حرکت سیارات در الگوریتم جستجوی گرانشی استفاده شده است و انتخاب والد‌ها در هر نسل از هر دو مجموعه فرزندان تولید شده و خود والد‌ها صورت می‌گیرد. روش پیشنهادی با در نظر گرفتن ۲۹ تابع بنچمارک استاندارد با روش‌های رایج و جدید موجود در این زمینه مقایسه شده است.

همچنین در این مقاله یک مدل گسسته جدید از الگوریتم پیشنهادی IWOGSA به نام DIWOGSA برای حل مسأله‌های بهینه‌سازی گسسته پیشنهاد شده است و کارایی آن بر روی یک چالش حیاتی تحت عنوان بیشینه‌سازی نفوذ ارزیابی شده است. مسأله بیشینه‌سازی نفوذ انتخاب مجموعه‌ای از گره‌های اولیه در شبکه‌های پیچیده برای گسترش نفوذ تا حد ممکن است. این مسأله به طور گسترده مورد مطالعه بسیاری از محققان قرار گرفته است. روش‌های حریصانه برای مسأله بیشینه‌سازی نفوذ نتایج خوبی را ارائه می‌دهد. اما این الگوریتم‌ها از مسائل مربوط به کارایی زمان و مقیاس‌پذیری رنج می‌برند. در مقابل الگوریتم‌های حریصانه، الگوریتم‌های به حداکثر رساندن تأثیر فرااکتشافی کارایی را بهبود

دقت پایین اشاره کرد [۲۲]. دقت حل کم، توانایی کم در جستجوی محلی و سرعت همگرایی پایین [۲۳] از معایب الگوریتم گرگ خاکستری [۱۲] و نرخ همگرایی آهسته، محاسبات بسیار از معایب الگوریتم تکامل تفاضلی [۲۴] است.

الگوریتم علف‌های هرز (IWO^۶) از الگوریتم‌های اکتشافی مبتنی بر جمعیت می‌باشد که توسط مهرآیین و لوکاس ارائه شده است [23]. این الگوریتم از روند رشد علف‌های هرز در طبیعت الهام می‌گیرد. به طور طبیعی، علف‌های هرز به شدت رشد می‌کنند و این رشد شدید یک تهدید جدی برای گیاهان مفید است. خصوصیات مهم علف‌های هرز، پایداری و سازگاری زیاد آن‌ها در طبیعت است که اساس بهینه‌سازی الگوریتم IWO است.

الگوریتم علف‌های هرز در مطالعات قبلی بهبود یافته که از آن جمله می‌توان به موارد زیر مانند IWO گسسته [24]، ترکیب الگوریتم علف‌های هرز با الگوریتم تکامل دیفرانسیل [25]، بکارگیری IWO برای الگوریتم‌های بهینه‌سازی چند هدفه [26]، بهبود یافته الگوریتم IWO [27, 28]، ترکیب الگوریتم PSO و الگوریتم بهینه‌سازی علف‌های هرز [29]، ترکیب الگوریتم بهینه‌سازی علف‌های هرز و الگوریتم کرم شب‌تاب [30] اشاره کرد.

الگوریتم جستجوی گرانشی از دسته الگوریتم‌های شبیه‌سازی ساختارهای فیزیکی از سوی خانم راشدی معرفی شد [۳۱]. این الگوریتم با شبیه‌سازی قوانینی شبیه به قانون گرانش و حرکت نیوتن در محیطی با زمان گسسته در فضای جستجو، طراحی شده است. الگوریتم GSA دارای ویژگی‌های مثبت زیادی همچون همگرایی سریع، عدم توقف در بهینه‌های محلی، کاهش حجم محاسباتی نسبت به الگوریتم‌های تکاملی و عدم نیاز به حافظه در مقایسه با دیگر الگوریتم‌های خانواده هوش جمعی، بستر جدیدی از تحقیقات را فرا روی محققان قرار داده است. از این رو، با توجه به زمینه‌های کاربردی مورد استفاده، نسخه‌های متفاوتی از الگوریتم GSA ارائه شده است که می‌توان به الگوریتم جستجوی گرانشی باینری (BGSA) [32] و الگوریتم جستجوی گرانشی نخبه‌گرایی پیشرفته (AEGSA) [۳۳] اشاره کرد.

^۶ Invasive Weed Optimization

^۷ No Free lunch theorem

پذیری بسیار بالای آن‌ها در طبیعت می‌باشد که این ویژگی مبنای بهینه‌سازی در الگوریتم IWO قرار گرفته است.

الگوریتم فراابتکاری علف‌های هرز یکی از الگوریتم‌های بهینه‌سازی جدید و توانمند است که بر اساس تقلید از قابلیت تطابق‌پذیری و تصادفی بودن کولونی علف‌های هرز، بهینه عمومی یک تابع ریاضی را پیدا می‌کند. علف هرز پدیده‌ای است که در جستجوی بهینگی و یافتن بهترین محیط برای زندگی بوده و به سرعت خود را با شرایط محیطی وفق داده و در مقابل تغییرات مقاوم می‌باشد. در ابتدا علف هرز به دنبال تولید تعداد زیاد فرزندان بوده که موجب افزایش کمیت و همچنین پوشش محیط در دسترس خود می‌شود، سپس به دلیل محدودیت ظرفیت، با افزایش کیفیت به رشد به صورت رقابتی ادامه می‌دهد و رفتار حریصانه خواهد داشت. به طور کلی هدف یافتن بهترین محیط برای زندگی علف‌های هرز می‌باشد. مراحل الگوریتم علف‌های هرز:

- پخش دانه در فضای مورد نظر
- رشد دانه‌ها با توجه به مطلوبیت (زاد و ولد و پراکندگی محیطی)
- ادامه حیات علف‌هایی با مطلوبیت بیشتر (حذف رقابتی)

۲-۲- الگوریتم جستجوی گرانشی

الگوریتم جستجوی گرانشی در سال ۲۰۰۹ از سوی خانم راشدی معرفی شد [31]. این الگوریتم با شبیه‌سازی قوانینی شبیه به قانون گرانش و حرکت نیوتن در محیطی با زمان گسسته در فضای جستجو، طراحی شده است.

مطابق قانون نیوتن هر جسم به جسم دیگر نیرویی وارد می‌کند که این نیرو با جرم خود جسم و معکوس مجذور فاصله تا جسم دیگر متناسب است. بنابراین، جرم‌های بزرگتر نیروی بیشتری بر جرم‌های کوچکتر وارد می‌کنند و برعکس جرم‌های کوچکتر نیروی کمتری بر جرم‌های بزرگتر وارد می‌کنند.

با توجه به قوانین فیزیکی هر جسم بر اثر نیرویی که به آن وارد می‌شود، دارای شتاب و در نتیجه سرعتی هم جهت با آن نیرو می‌شود. بنابراین، جرم‌های کوچکتر به علت آنکه نیروی اعمالی بر آن‌ها بسیار بیشتر است، به سمت جرم‌های بزرگتر حرکت می‌کنند. بدین ترتیب، اگر نقاط بهینه را محل اجرام بزرگتر در نظر بگیریم، با استفاده از قانون گرانش می‌توانیم این اطلاعات را به بقیه اجرام منتقل کنیم و آن‌ها را به سمت نقاط بهینه سوق دهیم.

مراحل مختلف الگوریتم جستجوی گرانشی عبارتست از:

۱. مشخص کردن فضای جستجو

می‌بخشند، اما نمی‌توانند نتایج دقیق را تضمین کنند. در این مقاله مدل گسسته از الگوریتم IWO به نام DIWOGSA برای مساله بهینه‌سازی نفوذ پیشنهاد شده است. روش پیشنهادی با در نظر گرفتن شبکه‌های مختلف با روش‌های رایج به نام EO, GSA, IWO, LSHADE-SPACMA و SHADE, FEP, GA, PSO, SSA, WOA مورد مقایسه قرار خواهد گرفت.

در ادامه پس از بیان توضیح مختصری از الگوریتم علف‌های هرز و جستجوی گرانشی ترکیبی از این دو الگوریتم برای حل مسائل بهینه‌سازی در قسمت دوم بیان خواهد شد. روش پیشنهادی با جزئیات کامل در قسمت سوم بیان شده است. قسمت چهارم حاوی ارزیابی روش پیشنهادی با در نظر گرفتن توابع بنچمارک استاندارد می‌باشد. در قسمت پنجم نحوه به کارگیری الگوریتم پیشنهادی برای حل سه مساله مهندسی رایج به نام طراحی مجرای فشار، طراحی پرتو جوش و مینیمم‌سازی وزن یک مطرح شده است. در قسمت ششم مدل گسسته الگوریتم IWO برای حل مساله بهینه‌سازی نفوذ پیشنهاد شده و مورد ارزیابی قرار گرفته و با روش‌های رایج هوشمند و حریصانه به نام Single Discount, DPSO, CELF, Degree, Betweenness Centrality و PageRank مقایسه شده است. در قسمت هفتم نتیجه‌گیری در ادامه کارهای آتی بیان خواهند شد.

۲- علف‌های هرز و جستجوی گرانشی

در این قسمت به بیان مختصری از الگوریتم‌های جستجوی گرانشی و الگوریتم علف‌های هرز می‌پردازیم و در ادامه یک روش ترکیبی از این دو روش برای غلبه بر کاستی‌های فردی آن‌ها در حل مسائل بهینه‌سازی مطرح می‌گردد.

۲-۱- الگوریتم علف‌های هرز

الگوریتم علف‌های هرز (IWO) یکی از الگوریتم‌های برجسته در حل مسائل بهینه‌سازی می‌باشد [23]. این الگوریتم را می‌توان با استفاده از تمهیدات خاص در شرایط گسسته، پیوسته و باینری بکار برده و نتایج بهتری از آن در مقایسه با سایر روش‌های بهینه‌سازی بدست آورد. الگوریتم علف‌های هرز در روش بهینه‌سازی از عملکرد رشد علف‌های هرز در طبیعت الهام گرفته است. در طبیعت علف‌های هرز رشد شدیدی دارند و این رشد شدید تهدیدی جدی برای گیاهان مفید می‌باشد. یکی از ویژگی‌های مهم علف‌های هرز پایداری و تطابق

۲. رقم‌دهی تصادفی
۳. ارزیابی برازش عامل‌ها
۴. بروز کردن $G(t)$, $best(t)$, $worst(t)$ و $M_i(t)$ برای $i=1, 2, \dots, N$
۵. محاسبه نیروی کلی در مسیرهای مختلف
۶. محاسبه سرعت و شتاب
۷. بروز کردن وضعیت و موقعیت عامل‌ها
۸. تکرار مراحل c تا g تا زمان رسیدن به معیار توقف
۹. پایان

$$a_i^d(t) = \frac{\left(\sum_{j \in X_{best}, j \neq i} rand_j \right) (G_0 \exp(-\alpha * t / (I_{max}))) \frac{M_i(t) \times M_j(t)}{R_{ij}(t) + \epsilon}}{M_i(t)} \quad (3)$$

که در آن $X_i^d(t)$ و $X_j^d(t)$ بردار راه‌حل‌های i ام و j ام است. $M_i(t)$ و $M_j(t)$ مقادیر محاسبه شده با رابطه ۲ می‌باشند. مقدار ϵ یک مقدار ثابت کوچک، $R_{ij}(t)$ فاصله اقلیدسی بین دو راه حل i و j ، G_0 و α مقادیر ثابت و $rand_j$ عددی تصادفی است. X_{best} مجموعه‌ای از راه‌حل‌های مطلوب از اعضای جمعیت است. تعداد اعضای این مجموعه در ابتدا برابر با اعضای جمعیت N بوده و با گذر زمان کاهش می‌یابد و در آخر فقط یک راه‌حل وجود دارد که به عنوان راه حل مطلوب شناخته می‌شود.

در ادامه مقادیر دو متغیر σ_{t1} و σ_{t2} به صورت مجزا با توجه به مقادیر اولیه و نهایی هر کدام با رابطه ۴-۵ به دست می‌آیند.

$$\sigma_{t1} = \left[\frac{(I_{max} - t)}{(I_{max})} \right]^2 \times (\sigma_{1initial} - \sigma_{1final}) + \sigma_{1final} \quad (4)$$

$$\sigma_{t2} = \left[\frac{(I_{max} - t)}{(I_{max})} \right]^2 \times (\sigma_{2initial} - \sigma_{2final}) + \sigma_{2final} \quad (5)$$

که در آن I_{max} بیشترین تعداد تکرار، t دور جاری، $\sigma_{1initial}$ و σ_{1final} مقدار اولیه و نهایی انحراف معیار اول و $\sigma_{2initial}$ و σ_{2final} مقدار اولیه و نهایی انحراف معیار دوم است.

مقدار تکثیر هر والد با بر اساس میزان شایستگی آن با رابطه ۶ مشخص می‌گردد.

$$S_i(t) = S_{min}(t) + (S_{max}(t) - S_{min}(t)) \times m_i(t) \quad (6)$$

در این مرحله هر والد با توجه به تعداد تکثیر محاسبه شده برای آن تکثیر خواهد شد. به طوری که راه‌حل‌های فرزند در حوالی والد خود به دو صورت توزیع می‌شوند:

a. ایجاد راه‌حل‌های فرزند با توزیع نرمال با احتمال

σ_{t2} با رابطه زیر صورت می‌گیرد و راه‌حل‌های جدید ایجاد شده در لیست جداگانه‌ای به نام I population با رابطه ۷-۸ قرار می‌گیرد.

$$V_i(t+1) = V_i(t) \quad (7)$$

۳- الگوریتم پیشنهادی IWOGSA

روش IWOGSA بهبود الگوریتم علف‌های هرز بر مبنای الگوریتم جستجوی گرانشی می‌باشد. در الگوریتم علف‌های هرز تکثیر دانه‌ها با توزیع نرمال صورت می‌گیرد ما برای بهبود این مرحله از الگوریتم GSA استفاده کرده‌ایم به طوری که تکثیر والد در روش پیشنهادی با دو روش مختلف توزیع نرمال و توزیع بر مبنای مکان مجموعه‌ای از راه‌حل‌های مطلوب با توجه به شایستگی آن‌ها صورت می‌گیرد. مراحل الگوریتم IWOGSA با بیان جزئیات هر مرحله در ادامه مطرح می‌گردد.

۳-۱- جمعیت اولیه

جمعیت اولیه به صورت تصادفی به تعداد اعضای جمعیت (N) تعریف شده به صورت برداری از متغیرهای مساله با توجه به ابعاد آن‌ها ایجاد می‌گردد.

۳-۲- تکثیر

برای تکثیر هر راه حل در ابتدا با توجه به برازندگی هر راه‌حل مقدار $M_i(t)$ برای هر راه‌حل با نرمال سازی مقادیر $m_i(t)$ با رابطه ۱-۲ محاسبه می‌شود.

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (1)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (2)$$

که در $fit_i(t)$ مقدار شایستگی راه‌حل i در دور t را نشان می‌دهد، $worst(t)$, $best(t)$ به ترتیب بهترین و بدترین مقدار شایستگی در

۱۲ بر اساس میزان شایستگی آن‌ها انتخاب می‌شوند به طوریکه مجموع این دانه‌ها از اندازه جمعیت مورد نظر بیشتر نشود.

$$NM = \text{round}(\min(\sigma_{t2} * N, \text{size}(\text{populationI}))) \quad (11)$$

$$NP = N - NM \quad (12)$$

که در آن N ماکزیمم اعضای جمعیت هر نسل، populationI جمعیت جدید ایجاد شده با توزیع نرمال، NM تعداد اعضای انتخابی از جمعیت جدید تولید شده با توزیع نرمال، NP تعداد اعضای انتخابی از مجموعه جمعیت فعلی و جمعیت جدید ایجاد شده بر مبنای روابط مبتنی بر شتاب و حرکت سیارات است.

۳-۴- شرط پایان

برای الگوریتم‌های فراابتکاری شرایط پایان متفاوتی را می‌توان تعریف کرد که از آن جمله می‌توان به تعداد تکرارهای خاص، محدودیت در تعداد دفعات اجرای تابع شایستگی، رسیدن به دقتی خاص، زمان خاص، عدم تغییر در نتیجه راه‌حل بعد از تعداد تکراری مشخص و یا ترکیبی از این‌ها اشاره کرد. شرط پایانی در الگوریتم پیشنهادی تعداد تکرار مشخص است. شبه کد روش پیشنهادی در الگوریتم ۱ و فلوجارت روش پیشنهادی در شکل ۱ نشان داده شده است.

$$\text{populationI} = [\text{populationI}; x_i(t) + \text{rand}_i \times \sigma_{t1}] \quad (8)$$

b. ایجاد راه‌حل‌های فرزند بر مبنای حرکت سیارات در الگوریتم جستجوی گرانشی (با توجه به سرعت و شتاب محاسبه شده) با احتمال $1 - \sigma_{t2}$ با رابطه ۹-۱۰ صورت گرفته و راه‌حل‌های ایجاد شده به اعضای جمعیت فعلی اضافه می‌گردد.

$$V_i(t+1) = \text{rand} \times V_i(t) + a_i(t) \quad (9)$$

$$\text{population} = [\text{population}; x_i(t) + V_i(t+1)] \quad (10)$$

قابل ذکر است که برای تعیین اینکه چه تعداد از راه‌حل‌ها با هر یک از روش‌ها ایجاد گردد، ما از یک توزیع نرمال دیگری به نام σ_{t2} استفاده کرده‌ایم. این مرحله برای آن است که در ابتدا تعداد بیشتری از دانه‌های فرزند با توزیع نرمال ایجاد شده و درصد بیشتری از آن‌ها به مراحل بعد انتقال یابند و در ادامه این تعداد کمتر گردد.

۳-۳- انتخاب جمعیت

انتخاب تعداد بیشتری از راه‌حل‌های ایجاد شده بر مبنای الگوریتم جستجوی گرانشی و جمعیت فعلی (NP) با رابطه ۱۱ و تعداد کمی از راه‌حل‌های ایجاد شده با توزیع نرمال (NM) با رابطه

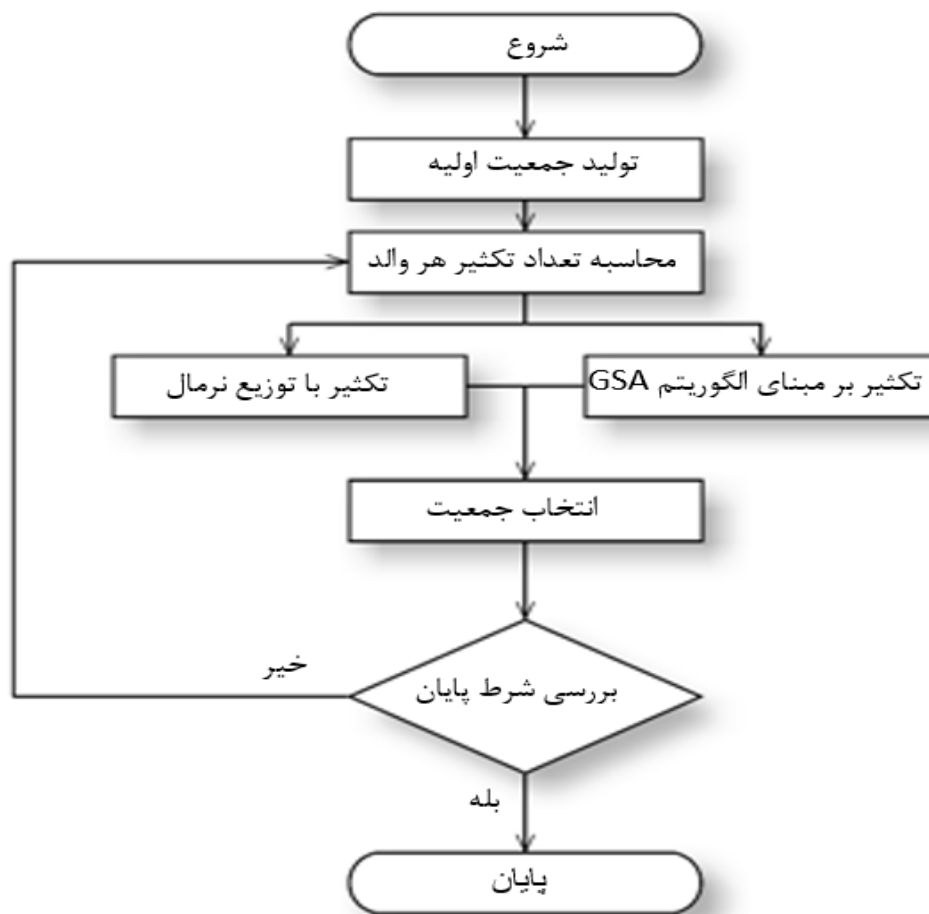
الگوریتم ۱: شبه کد الگوریتم IWOGSA

Input: maximum number of iteration MaxIt, number of population N, minimum and maximum number of offspring for each parents as S_{\min} and S_{\max} , $\sigma_{1\text{initial}}$ و $\sigma_{1\text{final}}$, $\sigma_{2\text{initial}}$ and $\sigma_{2\text{final}}$ for (Eq. 4-5) and alpha, G0 are constants with values equal 20, 100

1. Initialize population of solutions, set parameters;
2. Iter = 1;
3. Calculate the fitness of each solutions
4. while (Iter < MaxIt)
5. Compute the best and worst fitness in the population
6. Compute the σ_1 and the σ_2 using (Eq. 4-5)
7. Compute m, M, a for each solution in the population using (Eq. 1-3)
8. i=1
9. populationI = [31];
10. for i=1 to nPop
11. S_i → Compute the number of offsprings for solution_i using (Eq. 6)
12. for i=1: S_i
13. p=rand;
14. if p < σ_2
15. Produce offspring using (Eq.8)
16. Adjust the velocity of the new offspring using (Eq. 7)
17. Add offspring to the populationI
18. else
19. Produce offspring using (Eq. 9)
20. Update velocity using (Eq. 10)

```

21.      Add offspring to the population
22.      end
23.      end
24.      end
25.      Compute NM, NP using (Eq. 11-12);
26.      Select the number of NP offspring from the best members of population
           And the number of NM offsprings from the best offsprings of
           populationI as new_population
27.      population=new_population
28.      Iter=Iter+1;
29.end
30.output best solution from population
    
```



شکل ۱: فلوچارت الگوریتم IWOGSA

برداری و توانایی روش در فرار از بهینه محلی مورد ارزیابی قرار داد [۱۳، ۱۵، ۳۱، ۳۵].

این توابع به چهار دسته اصلی تقسیم‌بندی می‌شوند، توابع تک هدفه، توابع چندهدفه، توابع چندهدفه با ابعاد ثابت و توابع کامپوزیت. توابع (F1-F7) از دسته توابع تک هدفه هستند. این توابع ریاضی دارای یک راه‌حل واحد می‌باشند که برای ارزیابی بهره‌برداری روش‌های بهینه‌سازی طراحی شده‌اند. توابع تک هدفه به همراه ابعاد به کارگرفته شده آن‌ها در جدول نشان داده شده‌اند. توابع (F8-F13) از دسته توابع چندهدفه برای ارزیابی کاوش روش‌های بهینه‌سازی به

۴- ارزیابی روش پیشنهادی

برای بررسی کارایی الگوریتم پیشنهادی IWOGSA، این روش را با ۲۹ تابع بنچمارک مختلف ارزیابی کرده‌ایم. این توابع جزو توابع استاندارد بنچمارک می‌باشند که با استفاده از آن‌ها می‌توان روش‌های بهینه‌سازی را با در نظر گرفتن معیارهای مختلفی از جمله اکتشاف، بهره‌برداری و قدرت ایجاد تعادل بین اکتشاف و بهره

گزارش فنی CEC2005 یافت [36]. جزییات این دو دسته از توابع نیز به ترتیب در جدول و جدول آورده شده است. نمودار دو بعدی مربوط به نمونه‌هایی از این نوع توابع در شکل ۲ آورده شده است. این نمودارهای دو بعدی بر اساس مقادیر مختلف متغیرهای مربوط به هر تابع در بازه هر یک از آن‌ها که در جداول ۱-۴ بیان گردیده، رسم شده است.

کارگرفته می‌شوند. این توابع دارای بیش از یک راه‌حل بهینه هستند. جدول شامل جزییات این دسته از توابع می‌باشد. توابع (F14-F23) جزو توابع چندهدفه با ابعاد ثابت می‌باشند. که همانند دسته قبلی ولی با ابعاد کم و ثابت می‌باشند. توابع کامپوزیت (F24-F29) با داشتن تعداد زیادی از بهینه‌های محلی و پیچیدگی یک دامنه جستجوی واقعی را تقلید می‌کند. فلذا این توابع جزو توابع چالش برانگیز هستند. جزییات بیشتر در مورد عملکرد ترکیب را می‌توان در

جدول ۱. توابع آزمون تک هدفه

Function	D	Range	F min
$F_1(X) = \sum_{i=1}^n x_i^2$	30	[-100,100]	0
$F_2(X) = \sum_{i=1}^n X_i + \prod_{i=1}^n X_i $	30	[-10,10]	0
$F_3(X) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	30	[-100,100]	0
$F_4(X) = \max_i \{ x_i , 1 \leq i \leq n \}$	30	[-30,30]	0
$F_5(X) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[-100,100]	0
$F_6(X) = \sum_{i=1}^n ([x_i + 0.5])^2$	30	[-100,100]	0
$F_7(X) = \sum_{i=1}^n ix_i^4 + \text{random}[0.1]$	30	[-1.28,1.28]	0

جدول ۲. توابع آزمون چندهدفه

Function	D	Range	f _{min}
$F_8(x) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	30	[-500,500]	$-418.9829 \times d$
$F_9(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5.12,5.12]	0
$F_{10}(x) = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	30	[-32,32]	0
$F_{11}(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30	[-600,600]	0
$F_{12}(x) = \frac{\pi}{n} \{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \} + \sum_{i=1}^n u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i+1}{4} u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m x_i > a \\ 0 - a < x_i < a \\ k(-x_i - a)^m x_i < -a \end{cases}$	30	[-50,50]	0
$F_{13}(x) = 0.1 \{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	30	[-50,50]	0

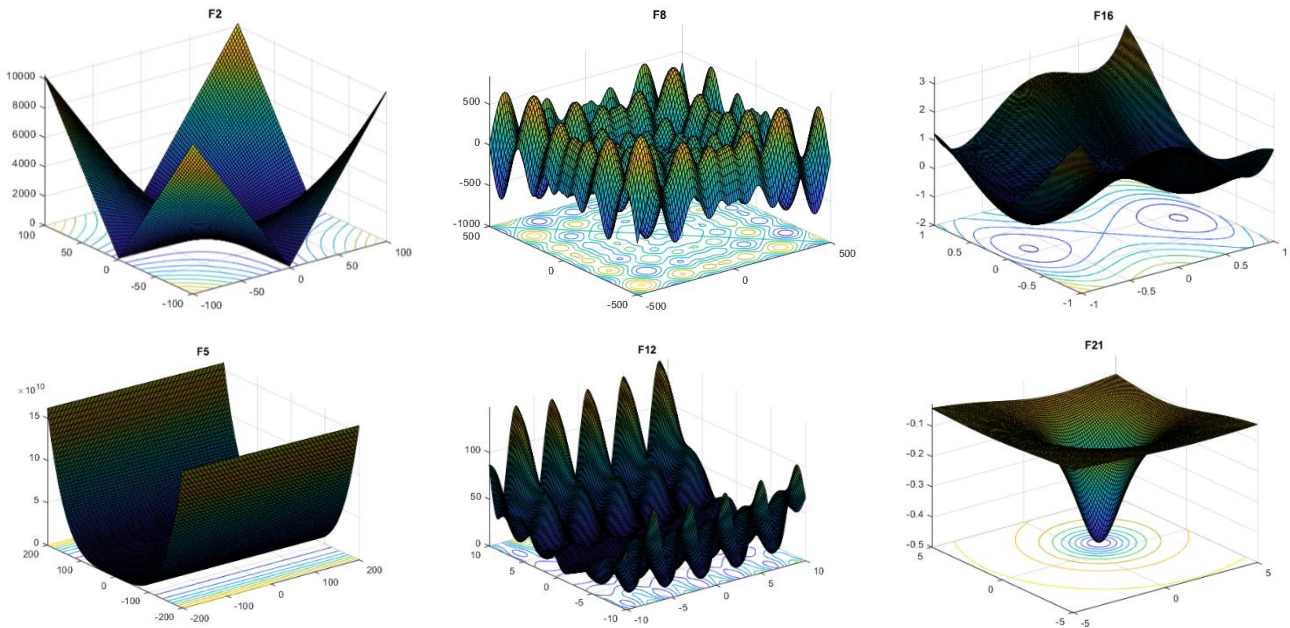
جدول ۳. توابع آزمون چندهدفه با ابعاد ثابت

Function	D	Range	f_{min}
$F_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum (x_i - a_{ij})^6} \right)^{-1}$	2	[-65,65]	0.998
$F_{15}(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_i(b_1^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$	4	[-5,5]	0.00030
$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[-5,5]	-1.0316
$F_{17}(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	2	[-5,5]	0.398
$F_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 + 36x_1x_2 + 27x_2^2)]$	2	[-2,2]	3
$F_{19}(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{i=1}^3 a_{ij}(x_j - p_{ij})^2)$	3	[1,3]	-3.86
$F_{20}(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{i=1}^6 a_{ij}(x_j - p_{ij})^2)$	6	[0,1]	-3.32
$F_{21}(x) = -\sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	-10.1532
$F_{22}(x) = -\sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	-10.4028
$F_{23}(x) = -\sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	-10.5363

جدول ۴. توابع آزمون composite

Function	D	Range	f_{min}
F_{24} (CF1) $f_1, f_2, \dots, f_{10} =$ Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [5 / 100, 5 / 100, 5 / 100, \dots, 5 / 100]$	10	[-5,5]	0
F_{25} (CF2) $f_1, f_2, f_3, \dots, f_{10} =$ Ciewank's Function $[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [5 / 100, 5 / 100, 5 / 100, \dots, 5 / 100]$	10	[-5,5]	0
F_{26} (CF3) $f_1, f_2, f_3, \dots, f_{10} =$ Griewank's Function $[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [1, 1, 1, \dots, 1]$	10	[-5,5]	0
F_{27} (CF4) $f_1, f_2 =$ Ackley's Function, $f_3, f_4 =$ Rastrigin's Function, $f_5, f_6 =$ Weierstrass Function, $f_7, f_8 =$ Griewank's Function, $f_9, f_{10} =$ Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [5 / 32, 5 / 32, 1, 1, 5 / 0.5, 5 / 0.5, 5 / 100, 5 / 100, 5 / 100, 5 / 100]$	10	[-5,5]	0
F_{28} (CF5) $f_1, f_2 =$ Rastrigin's Function, $f_3, f_4 =$ Weierstrass Function, $f_5, f_6 =$ Griewank's Function, $f_7, f_8 =$ Ackley's Function, $f_9, f_{10} =$ Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [1 / 5, 1 / 5, 5 / 0.5, 5 / 0.5, 5 / 100, 5 / 100, 5 / 32, 5 / 32, 5 / 100, 5 / 100]$	10	[-5,5]	0
F_{29} (CF6) $f_1, f_2 =$ Rastrigin's Function, $f_3, f_4 =$ Weierstrass Function, $f_5, f_6 =$ Griewank's Function, $f_7, f_8 =$ Ackley's Function, $f_9, f_{10} =$ Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]$	10	[-5,5]	0

$$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] = [0.1 * 1 / 5, 0.2 * 1 / 5, 0.3 * 5 / 0.5, 0.4 * 5 / 0.5, 0.5 * 5 / 100, 0.6 * 5 / 100, 0.7 * 5 / 32, 0.8 * 5 / 32, 0.9 * 5 / 100, 1 * 5 / 100]$$



شکل ۲: نمونه‌هایی از شکل‌های دو بعدی توابع بنچمارک (a) توابع تک هدفه، (b) توابع چندهدفه، (c) توابع چندهدفه با ابعاد ثابت

روش‌ها با هم برابر بوده و همانند مقاله [13] به ترتیب برابر با ۳۰ و ۵۰۰ در نظر گرفته شده‌اند.

جدول ۵: پارامترهای مربوط به الگوریتم پیشنهادی و الگوریتم‌های مورد مقایسه با توجه به مقاله اصلی آن‌ها

Algorithm	Parameters
IWOGSA	alpha, G0 are constants with values equal 20, 100 $\sigma_{2initial} = 0.01, \sigma_{2final} = 0.03$ $S_{min} = 0, S_{max} = 10$
IWO [23]	$S_{min} = 0, S_{max} = 10, Exponent = 2,$ $\sigma_{initial} = 1, \sigma_{final} = 0.001$
EO [13]	$a_1 = 2, a_2 = 1, GP = 0.5$
WOA [35]	$a_1 = [2 \ 0]; a_2 = [-2 \ -1]; b = 1$
PSO [13]	Cognitive = 2, social = 2 Inertial weight = [0.9 0.4]
GA [39]	mutation probability = 0.05, Crossover probability = 0.8, α = [-0.5, 1.5]
GSA [31]	Rnorm = 2, Rpower = 1, alpha = 20, and G0 = 100
SSA [42]	Leader position update probability = 0.5
LSHADE [40]	Pbest = 0.11, Arc rate = 1.4, Memory_size(H) = 5

مقادیر در نظر گرفته شده برای هر یک از پارامترهای مساله در جدول آورده شده است. پارامتر سیگما اولیه و نهایی برای روش IWOGSA برای هر یک از توابع مورد استفاده متفاوت بوده و به صورت دستی بهترین مقادیر برای آن‌ها به دست آمده است. پارامترهای مربوط به سایر الگوریتم‌ها مطابق با مقاله اصلی آن‌ها [13, 23, 31, 35, 37-41] در نظر گرفته شده است. برای بررسی روش پیشنهادی برنامه همانند الگوریتم‌های ارایه شده در مقاله‌های [13, 31, 35] بار اجرا شده و مقادیر میانگین و انحراف معیار محاسبه و برای مقایسه با روش‌های موجود مطرح می‌گردد. برای بررسی نتایج روش پیشنهادی، این روش با روش‌های بهینه‌سازی ازدحام ذرات (PSO) [37]، الگوریتم علف‌های هرز (IWO) [23]، بهینه‌سازی تعادل (EO) [13]، الگوریتم ازدحام سالپ (SSA) [42] و الگوریتم وال (WOA) [35] به عنوان روش‌های مبتنی بر SI، الگوریتم جستجوی گرانشی (GSA) [31] به عنوان روش‌های مبتنی بر فیزیک، روش برنامه‌نویسی تکاملی سریع (FEP) [38] و الگوریتم ژنتیک (GA) [39] به عنوان الگوریتم‌های تکاملی و دو روش SHADE [40] و LSHADE-SPACMA [41] به عنوان بهینه‌سازهایی با عملکرد بالا و برندگان رقابت‌های CEC انتخاب شده‌اند. برای مقایسه منصفانه تعداد جمعیت اولیه و تعداد تکرارها در همه

پیشنهادی با این دسته از توابع در مقایسه با روش‌های موجود در قسمت چهارم در جدول ۶ آورده شده است. الگوریتم IWOGSA تابع F24 در مقایسه با سایر روش‌ها به رتبه اول، در تابع F26 و F28 به رتبه دوم و در تابع F25 و F29 به رتبه سوم دست یافته است. در حالت کلی می‌توان گفت که روش پیشنهادی توانسته است در این دسته از توابع با پیچیدگی زیاد در میان ۱۱ روش مطرح عملکرد مناسبی را از خود نشان دهد.

۴-۲- مقایسه روش‌ها

برای مقایسه روش پیشنهادی با روش‌های موجود مطرح از روش رتبه‌بندی فریدمن استفاده شده است. آزمون فریدمن یک آزمون ناپارامتری، معادل آنالیز واریانس با اندازه‌های تکراری (درون گروهی است) که از آن برای مقایسه میانگین رتبه‌ها در بین k متغیر (گروه) استفاده می‌شود. مطابق رتبه‌بندی انجام شده با روش فریدمن روش پیشنهادی در مقایسه با روش‌های موجود مطرح شده دارای بالاترین رتبه با در نظر گرفتن ۲۹ تابع آزمون مختلف بوده است. رتبه محاسبه شده برای هر روش در سطر آخر جدول ۶ آورده شده است. مطابق رتبه محاسبه شده می‌توان میزان برتری روش IWOGSA را نسبت به سایر روش‌ها با رابطه ۱۳ به دست آورد.

$$(13) \quad \frac{(\text{new_value} - \text{original_value})}{|\text{original_value}| * 100}$$

که در آن new_value رتبه IWOGSA و original_value رتبه مربوط به هر کدام از روش‌های دیگری برای محاسبه میزان برتری روش IWOGSA نسبت به آن روش خواهد بود.

مطابق محاسبات انجام شده برای ۲۹ تابع ریاضی استاندارد IWOGSA قادر به پیشی گرفتن از GSA, IWO, EO, WOA, SSA, PSO, GA, FEP, SHADE و LSHADE-SPACMA به ترتیب برابر با ۵۴٪، ۴۰٪، ۱۲٪، ۳۰٪، ۴۳٪، ۴۲٪، ۵۱٪، ۴۴٪، ۱۷٪، و ۲۵٪ بوده است.

<i>LSHADE-SPACMA</i> [41]	Learning rate (c) = 0.8, threshold = $10e - 8$ $H = 5, P_{best} = 0.11, \text{Arc rate} = 1.4$ Probability variable (F_{cp}) = 0.5
---------------------------	---

۴-۱- نتایج ارزیابی

توابع (F1-F7) از دسته توابع تک هدفه، توابع ریاضی دارای یک راه‌حل واحد برای ارزیابی توانایی اکتشاف روش‌های بهینه‌سازی طراحی شده‌اند. نتایج ارزیابی روش پیشنهادی با این توابع نیز در جدول ۶ آورده شده است. مطابق نتایج به دست آمده روش IWOGSA در تمامی این توابع نسبت به الگوریتم IWO عملکرد بهتری را داشته است و در مقایسه با GSA جز در توابع F1، F4 و F6 در بقیه توابع نتایج بهتری را کسب کرده است.

بر خلاف توابع تک هدفه، توابع چندهدفه (F8-F23) شامل بهینه‌های محلی بسیاری است که تعداد آن‌ها بطور نمایی با اندازه مسئله افزایش می‌یابد (تعداد متغیرهای طراحی). بنابراین، برای ارزیابی توانایی اکتشافی یک الگوریتم بهینه‌سازی، این نوع توابع می‌تواند بسیار مفید باشد. مطابق نتایج به دست آمده و مطرح شده در قسمت دوم و سوم جدول ۶ روش IWOGSA توانسته است در توابع F18، F19، F22 و F23 نسبت به بقیه روش‌ها به نتیجه برابر یا بهتر دست یابد و در توابع F10، F15 و F20 با اختلافی کم به عنوان دومین بهترین روش در مقایسه با سایر روش‌ها می‌باشد. در تابع F13 و F21 نیز روش IWOGSA به عنوان سومین بهترین در میان ۱۱ روش مطرح شده می‌باشد. در این دسته از توابع نیز الگوریتم IWOGSA نسبت به الگوریتم IWO در تمامی توابع و نسبت به الگوریتم GSA در تمامی توابع به جز F20 بهبود قابل توجهی را داشته است.

بهینه‌سازی توابع ریاضی کمپوزیت (F24-F29) یک کار بسیار چالش برانگیز است زیرا تنها با ایجاد تعادل مناسب بین اکتشاف و بهره برداری می‌توان از دام بهینه محلی رها شد. نتایج ارزیابی روش

جدول ۶. نتایج ارزیابی روش پیشنهادی در مقایسه با روش‌های موجود

Function		IWOGSA	IWO	GSA	EO	WOA	SSA	PSO	GA	FEP	SHADE	LSHADES PACMA	
unimodal	F ₁	Avg	1.07E-15	4.59802	2.53E-16	3.32E-40	1.41E-30	1.58E-07	0.00013 6	0.55492	0.00057	1.42E-09	0.2237
		Std	1.52E-15	15.7815 7	9.67E-17	6.78E-40	4.91E-30	1.71E-07	0.00020 2	1.2301	0.00013	3.09E-09	0.148
	F ₂	Avg	1.78E-06	0.04889 7	0.05565 5	7.12E-23	1.06E-21	2.66293	0.04214 4	0.00566	0.0081	0.0087	21.1133
		Std	8.89E-06	0.01218 8	0.19407 4	6.36E-23	2.39E-21	1.66802	0.04542 1	0.01443	0.00077	0.0213	9.5781
	F ₃	Avg	20.7435 8	3898.38	896.534 7	8.06E-09	5.39E-07	1709.94	70.1256 2	846.344	0.016	15.4352	88.7746
		Std	6.17281 6	1932.49 8	318.955 9	1.6E-08	2.93E-06	11242.3	22.1192 4	161.499	0.014	9.9489	47.23
	F ₄	Avg	17.3166 4	29.0830 3	7.35487	5.39E-10	0.07258 1	11.6741	1.08648 1	4.55538	0.3	0.9796	2.117
		Std	5.39977 7	4.01135 7	1.74145 2	1.38E-09	0.39747	4.1792	0.31703 9	0.59153	0.5	0.7995	0.4928
	F ₅	Avg	45.7164 7	129.534 4	67.5430 9	25.3233 1	27.8655 8	296.125	96.7183 2	268.248	5.06	24.4743	28.8255
		Std	29.9807 4	272.434 2	62.2253 4	0.16957 8	0.76362 6	508.863	60.1155 9	337.693	5.87	11.208	0.8242
	F ₆	Avg	1.16363 6	86.5666 7	2.5E-16	8.29E-06	3.11626 6	1.8E-07	0.00010 2	0.5625	0	5.31E-10	0.2489
		Std	2.01626 4	209.896 3	1.74E-16	5.02E-06	0.53242 9	3E-07	8.28E-05	1.71977	0	6.35E-10	0.1131
	F ₇	Avg	0.01143 8	0.03159 9	0.08944 1	0.00117 1	0.00142 5	0.1757	0.12285 4	0.04293	0.1415	0.0235	0.0047
		Std	0.00589 4	0.00810 7	0.04339	0.00065 4	0.00114 9	0.0629	0.04495 7	0.00594	0.3522	0.0088	0.0019
multimodal	F ₈	Avg	-7325.3	-6809.7	-2821.07	-9016.34	-508.76	-7455.8	-4841.29	-10546.1	-12554.5	-11713.1	-3154.4
		Std	656.517 8	676.798 9	493.037 5	595.111 3	695.796 8	772.811	1152.81 4	353.158	52.6	230.49	317.921
	F ₉	Avg	18.8047 2	56.1984 6	25.9684 1	0	0	58.3708	46.7042 3	30.8229	0.046	8.5332	67.542
		Std	4.24021 4	13.5811 8	7.47006 8	0	0	20.016	11.6293 8	7.57295	0.012	2.1959	10.016
	F ₁₀	Avg	2.15E-08	6.31848 3	0.06208 7	8.34E-14	7.4043	2.6796	0.27601 5	1.63551	0.018	0.3957	0.0393
		Std	2.01E-08	9.08154 4	0.23628	2.53E-14	9.89757 2	0.8275	0.50901	0.46224	0.0021	0.5868	0.0151
	F ₁₁	Avg	0.04291 4	463.730 6	27.7015 4	0	0.00028 9	0.016	0.00921 5	0.56112	0.016	0.0048	0.8948
		Std	0.01754 9	61.3062 7	5.04034 3	0	0.00158 6	0.0112	0.00772 4	0.26942	0.022	0.0077	0.1078
	F ₁₂	Avg	0.01036 7	6.39216 8	1.79961 7	7.97E-07	0.33967 6	6.9915	0.00691 7	0.03088	9.2E-06	0.0346	0.00081 8
		Std	0.03163 2	1.99841 8	0.95114	7.69E-07	0.21486 4	4.4175	0.02630 1	0.04092	3.6E-06	0.0875	0.001
F ₁₃	Avg	0.00073 2	0.00075 7	8.89908 4	0.02929 5	1.88901 5	15.8757	0.00667 5	0.36222	0.00016	0.00073 2	0.0102	
	Std	0.00278 8	0.00278 9	7.12624 1	0.03527 1	0.26608 8	16.1462	0.00890 7	0.30975	0.00007 3	0.0028	0.0103	

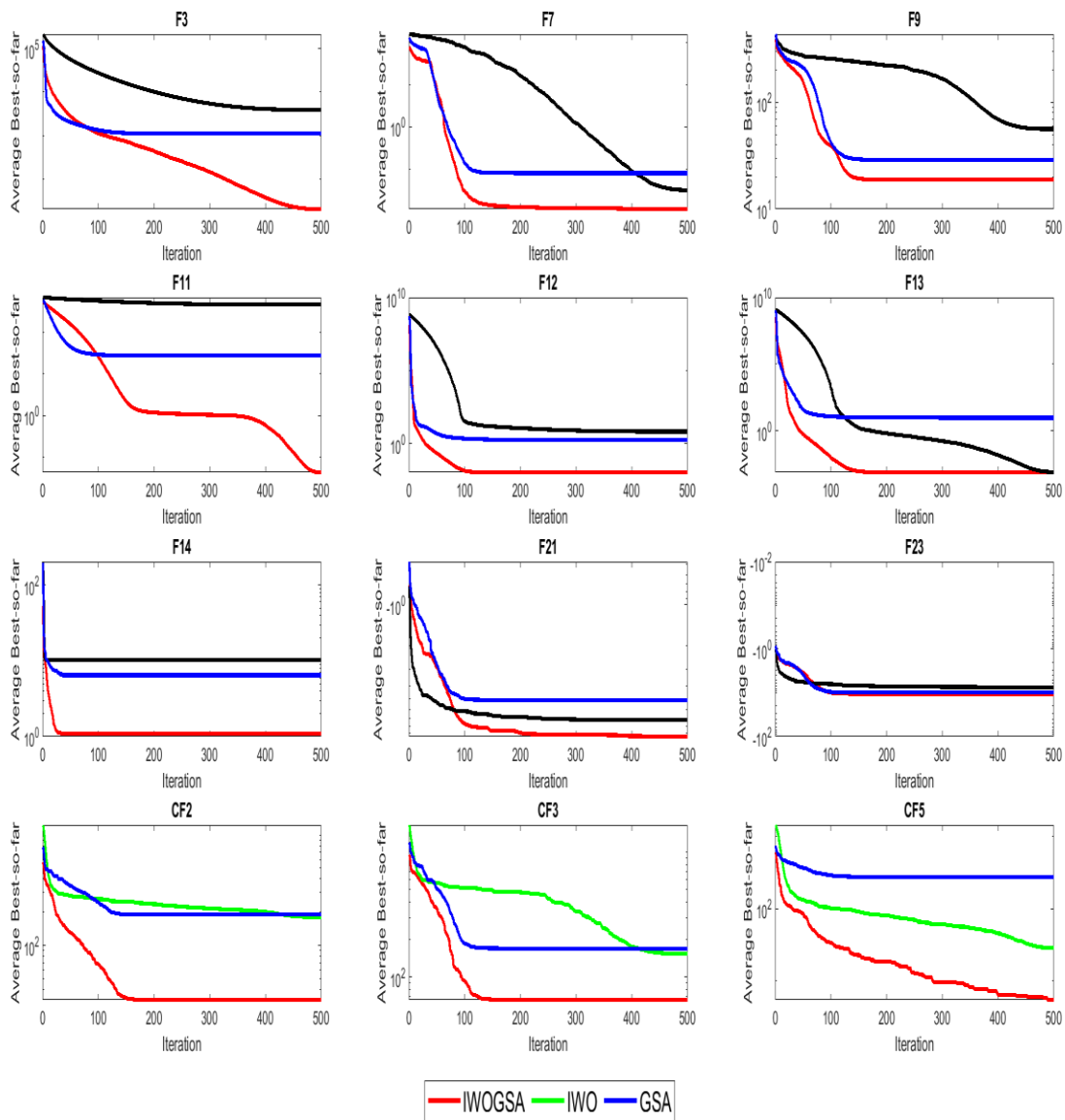
صفحه ۴۸ الی ۷۷

(fixed-dimension) Multimodal	F۱۴	Avg	1.09727 5	10.3778 9	5.85983 8	0.99800 4	2.11197 3	1.1965	3.62716 8	0.998004	1.22	0.99800 4	1.9416
		Std	0.39953	7.02353 9	3.83129 9	1.54E-16	2.49859 4	0.5467	2.56082 8	4.23E-12	0.56	5.83E-17	2.9633
	F۱۵	Avg	0.00030 7	0.00280 5	0.00367 3	0.00239 8	0.00057 2	0.00088 6	0.00057 7	0.005206	0.0005	0.00237 4	0.0003
		Std	1.83E-19	0.00595 9	0.00164 7	0.00609 7	0.00032 4	0.00025 7	0.00022 2	0.007028	0.00032	0.0061	1.93E-19
	F۱۶	Avg	-1.03163	-1.03163	-1.03163	-1.03162	-1.03163	-1.03163	-1.03163	-1.03162	-1.03	-1.03162	-1.03162
		Std	5.98E-16	7.68E-09	4.88E-16	6.04E-16	4.2E-07	6.13E-14	6.25E-16	1.34E-06	4.9E-07	6.51E-16	1E-15
	F۱۷	Avg	0.39788 7	0.39788 7	0.39788 7	0.39788 7	0.39791 4	0.39788 7	0.39788 7	0.39789	0.398	0.39788 7	0.39788 7
		Std	0	3.31E-09	0	0	0.00002 7	3.41E-14	0	1.08E-05	1.5E-07	3.24E-16	0
	F۱۸	Avg	3	3	3	3	3	3	3	3.000002	3.02	3	3
		Std	1.67E-15	4.83E-07	4.17E-15	1.56E-15	4.22E-15	2.2E-13	1.33E-15	4.06E-06	0.11	1.87E-15	1.25E-15
	F۱۹	Avg	-3.86278	-3.86278	-3.86278	-3.86278	-3.85616	-3.86278	-3.86278	-3.86278	-3.86	-3.86278	-3.86278
		Std	2.64E-15	1.54E-07	2.29E-15	2.59E-15	0.00270 6	1.47E-10	2.58E-15	1.63E-07	0.00001 4	2.69E-15	2.7E-15
	F20	Avg	-3.29821	-3.20309	-3.31778	-3.2687	-2.98105	-3.2304	-3.26634	-3.27443	-3.27	-3.27047	-3.28234
		Std	0.04837 6	1.31E-05	0.02308 1	0.05701	0.37665 3	0.0616	0.06051 6	0.05924	0.059	0.0599	0.057
	F21	Avg	-9.4722	-7.14234	-5.95512	-8.55481	-7.04918	-9.6334	-6.8651	-5.72536	-5.52	-9.2343	-9.4735
		Std	1.74414 8	3.37809 2	3.73707 9	2.76377	3.62955 1	1.8104	3.01964 4	3.32622	1.59	2.4153	1.7626
	F22	Avg	-10.4029	-9.36501	-9.68447	-9.3353	-8.18178	-9.0295	-8.45653	-6.94349	-5.53	-10.1479	-10.2258
		Std	4.66E-16	2.40842 5	2.01408 8	2.43834	3.82920 2	2.3911	3.08709 4	3.56118	2.12	1.3969	0.9704
F23	Avg	-10.5364	-7.38001	-10.5364	-9.63655	-9.34238	-9.0333	-9.95291	-7.0208	-6.57	-10.2809	-10.5364	
	Std	1.75E-15	3.73925 3	2.6E-15	2.38811	2.41473 7	2.9645	1.78278 6	3.85233	3.14	1.3995	1.77E-15	
Composite Function	F24 (CF1)	Avg	2.21E-17	80.0014 4	6.63E-17	66.666	0.56884 6	43.333	100	86.671	100	63.333	3.3333
		Std	3.73E-18	130.383 5	2.78E-17	95.893	0.50594 6	67.891	81.65	97.324	188.56	80.872	18.254
	F25 (CF2)	Avg	32.2428 9	180.479 4	200.620 2	89.837	75.3087 4	31.133	155.91	142.72	161.99	40.508	0
		Std	48.2279 8	44.3634 3	67.7208 7	56.366	43.0785 5	52.149	13.176	119.58	151	61.462	0
	F26 (CF3)	Avg	65.7547 9	153.692 5	180	161.73	55.6514 7	235.11	172.03	214.67	214.06	139.48	104.29
		Std	63.8172 6	81.2329 4	91.8936 6	33.227	21.8794 4	80.839	32.769	73.47	74.181	33.366	14.266
	F27 (CF4)	Avg	239.688 2	615.432 9	170	356.44	53.8377 8	232.44	314.3	447.01	616.4	316.62	278.63
		Std	32.3164 3	190.341 1	82.3272 6	115.66	21.621	43.643	20.066	112.34	671.92	96.752	7.067
	F28 (CF5)	Avg	5.45715 5	28.5066 6	200	52.309	77.8064	27.538	83.45	91.831	358.3	39.515	2.02E-17
		Std	6.53060 5	47.1644 7	47.1404 5	95.565	52.0234 6	41.598	101.11	73.898	168.26	51.233	7.69E-17
F29 (CF6)	Avg	490.526 3	900.001 9	142.090 6	768.48	57.8844 5	628.69	861.42	811.21	900.26	684.51	540.23	
	Std	31.4813 3	0.00034 4	88.8714 1	192.94	34.4460 1	184.48	125.81	173.11	0.0832	201.22	122.75	
Friedman mean rank		3.82758 6	8.27586 2	6.36206 9	4.32758 6	5.48275 9	6.70689 7	6.60344 8	7.810345	6.86206 9	4.62069 3	5.12069 4	
Rank		1	11	6	2	5	8	7	10	9	3	4	

۳-۴- تجزیه و تحلیل همگرایی

نشان داده شده روش ترکیبی IWOGSA نسبت به دو روش IWO و GSA در توابع تک هدفه و چندهدفه، F3، F7، F9، F11، F12، F13، F14، F21 و توابع کامپوزیت FC2، FC3 و FC5 به همگرایی سریع‌تر و بهتری رسیده است و این نشان می‌دهد که با ترکیب الگوریتم GSA با IWO، قابلیت همگرایی IWO و GSA اصلی بهبود یافته است.

نمودار همگرایی روش پیشنهادی در مقایسه با روش‌های علف‌های هرز (IWO) و الگوریتم جستجوی گرانشی در برخی از توابع بنچمارک در شکل ۳ آورده شده است. با توجه به نمودارهای نمونه‌ی



شکل ۳. مقایسه همگرایی الگوریتم‌های بهینه‌سازی IWOGSA، IWO و GSA در توابع مختلف

$$\begin{aligned} g_1(x) &= -x_1 + 0.0193x_3 \leq 0 \\ g_2(x) &= -x_2 + 0.0954x_3 \leq 0 \\ g_3(x) &= -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0 \\ g_4(x) &= x_4 - 240 \leq 0 \end{aligned} \quad (14)$$

۵- الگوریتم IWOGSA برای حل مسائل

مهندسی

روش پیشنهادی برای حل سه مساله مهندسی شناخته شده‌ی طراحی مجرای فشار، طراحی پرتو جوش و مینیمم‌سازی وزن یک مورد استفاده قرار گرفته و میزان کارایی آن در حل چنین مسائلی مورد ارزیابی قرار گرفته شده است. یک روش ساده برای کنترل محدودیت به نام جریمه ایستا، در حل این مسائل اعمال می‌شود تا اگر هرگونه محدودیت در هر سطح از مرزهای تعریف شده خود نقض شود، عملکرد هدف را با مقدار زیادی جریمه می‌کنیم. ضریب جریمه باید به اندازه کافی بزرگ باشد تا بتواند عملکرد هدف را در محدودیتهای برابری / نابرابری مجازات کند.

روابط مربوط به هر یک از این مسائل و نتایج حاصل از روش IWOGSA در حل آن‌ها در ادامه آورده شده است. قابل ذکر است که تمام ارزیابی‌ها به تعداد دفعات ۳۰ بار با در نظر گرفتن ۲۰ جمعیت اولیه و ۵۰۰ تکرار انجام شده و مقدار میانگین نتایج و بهترین نتیجه، بدترین نتیجه و مقدار انحراف معیار به دست آمده در مقایسه با روش‌های موجود در جداول مربوطه آورده شده است.

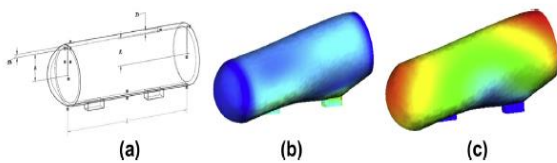
۵-۱- طراحی یک مجرای فشار (Pressure Vessel)

هدف از این مسئله مینیمم‌سازی هزینه کلی مواد، شکل دهی و جوشکاری یک مجرای استوانه‌ای نشان داده شده در شکل ۴ می‌باشد. در اینجا چهار متغیر طراحی وجود دارد: x_1 (T_s)، ضخامت پوسته، x_2 (T_h)، ضخامت دماغه، x_3 (R) شعاع داخلی و x_4 (L)، طول بخش استوانه ای (مجرا). و T_s و T_h مضرب‌های صحیحی از 0.0625 اینچ که ضخامت‌های در دسترس صفحات استیل پیچیده شده می‌باشند، خواهند بود. R و L پیوسته هستند. مسئله می‌تواند با رابطه ۱۴ تعریف شود:

$$f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_3$$

به منظور

از محدوده‌های زیر برای متغیرها استفاده می‌شود:
 $1 \leq x_1 \leq 99, 1 \leq x_2 \leq 99, 10.0 \leq x_3 \leq 200.0, 10.0 \leq x_4 \leq 200.0$



شکل ۴. مجرای فشار: (a) شماتیک، (b) نقشه حرارتی تنش (c) نقشه حرارتی جابجایی [15]

الگوریتم‌های تکاملی زیادی از جمله GSA، MSCA، PRO، SMO، WOA، GWO، PSO، GA، برای حل این مساله به کار رفته‌اند. الگوریتم GSA با مینیمم هزینه ۸۵۳۸،۸۳۶، الگوریتم MSCA با مینیمم هزینه ۵۹۳۵،۷۱۶، الگوریتم PRO با مینیمم هزینه ۶۰۵۹،۷۴۱، الگوریتم WOA با مینیمم هزینه ۶۰۵۰،۷۱۳، الگوریتم IWOGSA با مینیمم هزینه ۵۸۸۷،۸۸۷۵۰۵۲۵۹۶۸ در مقابل الگوریتم IWOGSA با مینیمم هزینه تنها نسبت به آن روش‌ها بلکه توانسته است نسبت به همه روش‌هایی دیگری که در جدول آورده شده است پیشی بگیرد. مقادیر میانگین، بیشترین، کمترین و انحراف معیار نتایج ارزیابی به همراه سایر روش‌هایی که این مقادیر برای آن‌ها قابل دسترس بود در جدول آورده شده است. مطابق این جدول نیز روش IWOGSA نسبت به همه روش‌های مورد مقایسه به نتایج بهتری رسیده است.

جدول ۷. مقایسه نتایج بهینه‌سازی IWOGSA با روش‌های بهینه‌سازی دیگر در طراحی مخزن تحت فشار

Algorithm	Optimal Value				Optimal Cost
	T_s	T_h	R	L	
IWOGSA	0.7785	0.3848	40.3375	1.99749	5885.92
A	15	21	98	8	6
IWO	0.7789	0.3850	40.3565	1.99517	5887.88
	40	82	98	4	7
GSA	1.125	0.625	55.9886	84.4542	8538.83
[35]			6		6
MSCA	0.7792	0.3996	40.3254	199.921	5935.71
[43]	56		5	3	6

$$\begin{aligned}
 g_1(x) &= \tau(X) - \tau_{\max} \leq 0 \\
 g_2(x) &= \sigma(X) - \sigma_{\max} \leq 0 \\
 g_3(x) &= x_1 - x_4 \leq 0 \\
 g_4(x) &= 0.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0 \\
 g_5(x) &= 0.125 - x_1 \leq 0 \\
 g_6(x) &= \delta(x) - \delta_{\max} \leq 0 \\
 g_7(x) &= P - P_c(x)
 \end{aligned} \quad (15)$$

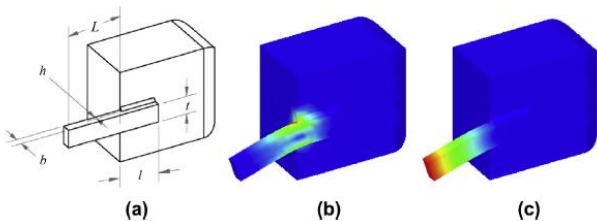
که

$$\begin{aligned}
 \tau(x) &= \sqrt{(\tau')^2 + 2\tau'\tau'' \frac{x_2}{2R} + (\tau'')^2} \\
 \tau' &= \frac{P}{\sqrt{2x_1x_2}}, \quad \tau'' = \frac{MR}{J}, \quad M = P \left(L + \frac{x_2}{2} \right) \\
 R &= \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2} \right)^2} \\
 J &= 2 \left\{ \sqrt{2x_1x_2} \left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2} \right)^2 \right] \right\}
 \end{aligned}$$

$$\begin{aligned}
 \sigma(X) &= \frac{6PL}{x_4x_3^2}, \quad \delta(X) = \frac{4PL^3}{Ex_3^3x_4} \\
 P_c(X) &= \frac{4.013E\sqrt{x_3^2x_4^6}}{L^2} \left(1 - \frac{x_3}{2L} \sqrt{\frac{E}{4G}} \right) \\
 P &= 6000lb, \quad L = 14in, \quad E = 30 \times 10^6 psi \\
 G &= 12 \times 10^6 psi, \quad \tau_{\max} = 13600 psi \\
 \sigma_{\max} &= 30000 psi, \quad \delta_{\max} = 0.25in
 \end{aligned}$$

محدوده‌های زیر برای متغیرها استفاده می‌شود.

$$0.1 \leq x_1 \leq 2.0, \quad 0.1 \leq x_2 \leq 10.0, \quad 0.1 \leq x_3 \leq 10.0, \quad 0.1 \leq x_4 \leq 2.0$$



شکل ۵. ساختار طرح تیر جوش (a) شماتیک (b) نقشه حرارتی

تنش (c) نقشه حرارتی جابجایی [15]

Algorithm	Min	mean	Max	Std
EO [13]	0.8125	0.4375	42.0984	176.636
			4	59
			59	43
PRO [44]	0.7445	0.4424	38.4899	200
			8	3
SMO [45]	0.8242	0.4072	42.3658	173.797
			5	3
			3	9
WOA [35]	0.8125	0.4375	42.0982	176.638
			7	9
			9	1
GWO [15]	0.8125	0.4345	42.0891	176.758
			8	7
			7	4
PSO [46]	0.8125	0.4375	42.0912	176.746
			7	5
			5	8
GA [39]	0.8125	0.4345	40.3239	200
				6288.74
				5

جدول ۸. مقایسه نتایج آماری ICO با روش‌های بهینه‌سازی دیگر در طراحی مخزن تحت فشار

Algorithm	Min	mean	Max	Std
IWOGSA	5885.926	7918.9905	52628.27	8451.458
IWO	5887.888	22294.318	90989.54	25337.115
GSA [35]	8538.836	8932.95	65535	683.5475
EO [13]	6059.714	6668.114	7544.493	566.24
WOA [35]	6059.741	6068.05	65535	65.6519
PSO [46]	6061.078	6531.1	65535	154.3716
CPSO [46]	6061.078	6147.133	6363.804	86.4545
HGA(1) [47]	6065.821	6632.376	8248.003	515
HGA(2) [47]	6832.584	7187.314	8012.615	276
T-Cell [48]	6390.554	6737.065	7694.066	357
DTS-GA [49]	6059.946	6177.253	6469.322	130.92
HAIS-GA [50]	6061.123	6743.085	7368.06	457.99
BFOA [51]	6060.46	6074.625	65535	156
ES [52]	6059.746	6850	7332.87	426

۵-۲- طراحی پرتو جوش (Welded Beam)

هدف مینیمم سازی هزینه یک پرتو جوش به منظور قیدهایی بر روی تنش کششی، تنش خمشی در پرتو، بار مقابل بر روی میله، خمش نهایی پرتو و قیدهای جانبی می‌باشد که در شکل ۵ نشان داده شده است. مسئله با رابطه ۱۵ نمایش داده می‌شود:

مینیمم سازی

$$f(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$$

با هدف

WOA [35]	1.73049 9	1.732	65535	0.0226
PSO [46]	65535	1.7422	65535	1.01275
SBM [55]	2.4426	2.5215	2.6315	65535
BFOA [51]	2.3868	2.404	65535	0.016
SCA [56]	2.3854	3.2551	6.3996	0.959
EA [57]	2.3816	65535	2.38297	0.00034
T-Cell [48]	2.3811	2.4398	2.7104	0.09314
FSA [58]	2.3811	2.4041	2.4889	65535
IPSO [59]	2.381	2.3819	65535	0.00523
HSA-GA [60]	2.25	2.26	2.28	0.0078
CDE [61]	1.7335	1.76815 8	1.82410 5	0.02219 4
CPSO [46]	1.728	1.74883 1	1.78214 3	0.01292 6

الگوریتم‌های تکاملی مختلفی برای حل این مساله به کار گرفته شده‌اند. مینیمم مقدار به دست آمده به همراه مقادیر محاسبه شده برای متغیرها در مقایسه با روش‌های موجود در جدول آورده شده است. همچنین مقادیر میانگین، بیشترین، کمترین و انحراف معیار نتایج ۳۰ بار اجرا انجام شده در مقایسه با روش‌های موجود در جدول آورده شده است. همانگونه که در جداول مشخص است روش IWOGSA عملکرد مناسبی را برای حل این مساله نداشته است و در مقابل الگوریتم MSCA با مقدار بهینه ۱,۶۹۷۹ از بقیه روش‌ها در حل این مساله پیشی گرفته است.

جدول ۹. مقایسه نتایج بهینه‌سازی ICO با روش‌های بهینه‌سازی دیگر در مساله طراحی تیر جوش

Algorithm m	Optimal Value				Optimal Cost
	H	L	t	B	
IWOGSA	0.205732 96	7.09241 4	9.03662 39	0.20572 96	2.21815 1
IWO	0.206507	7.07893 4	9.01893 4	0.20663	2.22336 5
GSA [35]	0.182129	3.85697 9	10	0.20237 6	1.87995 2
MSCA [43]	0.20545	3.2524	9.0576	0.20568	1.6979
WOA [35]	0.205396	3.48429 3	9.03742 6	0.20627 6	1.73049 9
EO [13]	0.2057	3.4705	9.03664	0.2057	1.7249
GWO [15]	0.205676	3.47837 7	9.03681	0.20577 8	1.72624
GA [39]	65535	65535	65535	65535	1.8245
HS [53]	0.2442	6.2231	8.2915	0.2433	2.3807
Random [54]	0.4575	4.7313	5.0853	0.66	4.1185
Simple [54]	0.2792	5.6256	7.7512	0.2796	2.5307
David [54]	0.2434	6.2552	8.2915	0.2444	2.3841
Approx [54]	0.2444	6.2189	8.2915	0.2444	2.3815

۵-۳- مینیمم سازی وزن یک

این مسئله شامل مینیمم سازی وزن یک^۱ به منظور قیدهایی بر روی مینیمم انحراف، تنش کشش، فرکانس غلبان و محدودیت بر روی ضخامت خارجی و بر روی متغیرهای طراحی می‌باشد که در شکل ۶ نشان داده شده است. متغیرهای طراحی، متوسط قطر هسته D، قطر سیم d و تعداد هسته‌های فعال N می‌باشد. مسئله می‌تواند با رابطه ۱۶ تشریح گردد:

مینیمم سازی

$$f(x) = (N+2)Dd^2$$

به منظور

$$g_1(X) = 1 - \frac{D^3 N}{71785d^4} \leq 0$$

$$g_2(X) = \frac{4D^2 - dD}{12566(Dd^3 - d^4)} + \frac{1}{5108d^2} - 1 \leq 0 \quad (16)$$

$$g_3(X) = 1 - \frac{140.45d}{D^2 N}$$

$$g_4(X) = \frac{D+d}{1.5} - 1 \leq 0$$

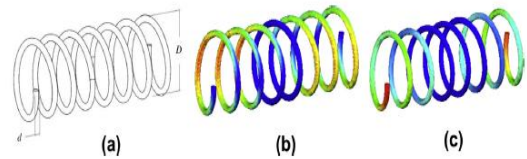
و از محدوده‌های زیر برای متغیرها استفاده می‌شود:

جدول ۱۰. مقایسه نتایج آماری IWOGSA با روش‌های

بهینه‌سازی دیگر در مساله طراحی تیر جوش

Algorithm m	Min	mean	Max	Std
IWOGS A	2.21815 1	2.25857 9	2.54442 5	0.07263 0
GSA [35]	1.87995 2	3.5761	65535	0.2874
EO [13]	1.72485 3	1.72648 2	1.73672 5	0.00325 7

$$0.05 \leq x_1 \leq 2.0, 0.25 \leq x_2 \leq 1.3, 2.0 \leq x_3 \leq 15.0$$



شکل ۶. مینیم سازی وزن یک: (a) شماتیک، (b) نقشه حرارتی تنش (c) نقشه حرارتی جابجایی [15]

این مسئله مدل سازی ریاضی را می‌توان با تکنیک‌های ریاضی مانند تصحیح محدودیت در یک تابع هزینه و توابع جریمه [62] و یا با الگوریتم‌های فرا اکتشافی حل کرد. Chen و همکاران در مقاله [43] از روش MSCA که بهبود یافته روش SCA می‌باشد برای حل این مساله استفاده کرده و به وزن بهینه 0.126670 رسیدند. Coello و همکاران [39] از GA برای رفع این مشکل استفاده کرده و وزن بهینه 0.127048 بدست آورده‌اند.

الگوریتم‌های WOA, EO, GWO, PSO, GA, RO, HIS و DE برای حل این کار استفاده شدند. همانطور که در جدول دیده می‌شود، IWOGSA با داشتن وزن مطلوب 0.012666 ، در کنار الگوریتم EO و GWO بهترین الگوریتم در بین الگوریتم‌های متااکتشافی استاندارد و تکنیک‌های ریاضی مورد استفاده در این مقاله است و این امر نشان می‌دهد که IWOGSA می‌تواند به عنوان ابزاری مفید برای این مساله باشد. مقدار مینیم، ماکزیمم، میانگین و انحراف معیار محاسبه شده برای این الگوریتم در مقایسه با روش‌هایی که مقادیر مذکور برای آن‌ها قابل دسترس بود برای مقایسه در جدول آورده شده است. قابل ذکر است که میانگین و انحراف معیار به دست آمده با روش IWOGSA از میانگین و انحراف معیار به دست آمده با روش EO بهتر است.

جدول ۱۱. مقایسه نتایج بهینه‌سازی IWOGSA با روش‌های

بهینه‌سازی دیگر در مساله مینیم سازی وزن یک

Algorithm	Optimal Value			Optimal Cost
	D	N	D	
IWOGSA	0.0515	0.3529	11.5149	0.012666
IWO	0.0523	0.3708	10.5403	0.012711
MSCA [43]	0.050276	0.32368	13.52541	0.012702
WOA [35]	0.051207	0.345215	12.00403	0.012676
EO [13]	0.05162	0.355054	11.38797	0.012666
GWO [15]	0.05169	0.356737	11.28885	0.012666
GSA [35]	0.051668	0.356199	11.3207	0.012667
PSO [46]	0.051728	0.357644	11.24454	0.012674
GA [39]	0.05148	0.351661	11.6322	0.012704
RO [22]	0.05137	0.349096	11.76279	0.012679
DE [61]	0.051609	0.354714	11.41083	0.012670

جدول ۱۲. مقایسه نتایج آماری IWOGSA با روش‌های

بهینه‌سازی دیگر در مساله مینیم سازی وزن یک

Algorithm	Min	Mean	Max	Std
IWOGSA	0.012666	0.012752	0.013031	0.000084
IWO	0.012711	0.012861	0.013807	0.000263
GSA [35]	0.012702	0.0136	65535	0.002630
WOA [35]	0.012676	0.0127	65535	0.0003
PSO [46]	0.012675	139	65535	0.0033
SI [63]	0.01306	0.015526	0.018992	65535
GA(1) [39]	0.012704	0.012769	0.012822	3.93E-05
T-Cell [48]	0.012665	0.013309	0.012732	0.000094
GA(2) [64]	0.012681	0.012742	0.012973	0.000095
SCA [56]	0.012669	0.012922	0.016717	0.000592
EO [13]	0.012666	0.01302	0.013997	0.000391
CDE [61]	0.01267	0.012703	0.01279	2.07E-05

۶- الگوریتم IWOGSA برای مساله گسسته

افزایش نفوذ

بهینه‌سازی نفوذ یک مساله بهینه‌سازی است که هدف از آن انتخاب گره‌های اولیه با بیشترین میزان نفوذ است. این مساله توسط الگوریتم‌های تکاملی مختلفی [65-68] توسط محققان مورد بررسی قرار گرفته شده است. الگوریتم تکاملی IWOGSA برای حل مسائل بهینه‌سازی پیوسته در قسمت‌های قبل مطرح گردید در این بخش مدل گسسته IWOGSA برای به کارگیری این الگوریتم برای مساله ی افزایش نفوذ به نام DIWOGSA بیان می‌گردد.

۶-۱- الگوریتم پیشنهادی DIWOGSA

روش پیشنهادی از یک استراتژی مبتنی بر توپولوژی شبکه برای بهینه‌سازی نفوذ در شبکه‌های پیچیده استفاده می‌کند. در روش پیشنهادی یک عملگر هوشمند جدید مبتنی بر درجه گره‌ها برای مقاردهی راه حل‌های اولیه به کار گرفته شده است. همچنین از یک عملگر جستجوی محلی برای افزایش سرعت همگرایی این مساله استفاده شده است. در ادامه مراحل روش پیشنهادی برای مساله افزایش نفوذ با جزئیات کامل بیان می‌گردد.

۶-۱-۱- نمایش مساله

هر یک از راه حل‌ها از جمعیت از یک بردار k عضوی تشکیل می‌یابد $X_i = (X_{i1}, X_{i2}, \dots, X_{ik})$ که هر امان از این ارایه X_{ij} بیانگر گرهی از گراف G است. برای مثال اگر $k=5$ و بردار $X_i =$

۴-۱-۶ تکثیر

در DIWOGSA نیز همانند حالت پیوسته تمامی راه حل ها در جمعیت فعلی به عنوان والد در نظر گرفته شده و تکثیر می شوند. در روش گسسته تعداد تکثیر هر والد همانند روش پیوسته بر مبنای شایستگی هر والد محاسبه می شود ولی نحوه تکثیر متفاوت است به طوری که در روش گسسته مقادیر σ_{t1} ، σ_{t2} و σ_{t1} ، σ_{t2} همانند روش پیوسته محاسبه می شود سپس مقدار فرزندان با رابطه ۱۸ ایجاد خواهند شد.

$$X_{ij} = \begin{cases} X_{ij} & \text{if } rand_i \times \sigma_{t1} \geq 1 \\ Replace(X_{i,j}, Nodes) & \text{else} \end{cases} \quad \text{if } r \leq \sigma_{t2} \quad (18)$$

$$= \begin{cases} X_{ij} & \text{if } V_{ij} \geq 1 \\ Replace(X_{i,j}, Nodes) & \text{else} \end{cases}$$

که در آن r یک عدد تصادفی در بازه ۰ تا ۱، و با تابع $Replace(X_{i,j}, Nodes)$ یک گره به صورت تصادفی از میان گره های شبکه که در راه حل نباشد انتخاب شده و با $X_{i,j}$ جایگزین می شود.

۵-۱-۶ جستجوی محلی

بعد از انتخاب بهترین گره ها و به کارگیری جستجوی محلی برای افزایش شایستگی بهترین گره صورت می گیرد. در این مرحله تک تک گره های موجود در راه حل مورد بررسی قرار گرفته و با اولین همسایه ای که منجر به افزایش نفوذ در راه حل گردد جایگزین خواهد شد. شبه کد جستجوی محلی در الگوریتم ۲ آورده شده است.

Algorithm 2. local search

1: Input: Particle Xa.
2: Xb ← Xa;
3: for each element $x_{bi} \in X_b$ do
4: Flag ← FALSE;
5: Neighbors ← $N_{x_{bi}}^1$
6: while (1)
7: $x_{bi} \leftarrow Replace(x_{bi}, Neighbors);$
8: if EDV(Xb) > EDV(Xa) then
9: Xa ← Xb;
10: else
11: Flag TRUE;

(A, E, H, I, C) باشد. این به این معنی است که گره A, E, H, I, C و C به عنوان گره های اولیه^۹ از گراف G انتخاب شده و نفوذ در شبکه با این گره ها صورت می گیرد.

۲-۱-۶ تابع هدف

محاسبه میزان نفوذ و کشف مجموعه ای از دانه ها با بهترین نفوذ در یک شبکه دو مساله در زمینه بهینه سازی نفوذ است که به ترتیب به عنوان مسائل P-hard و NP-hard شناخته شده اند. به منظور محاسبه میزان گسترش نفوذ، شبیه سازی سنتی مونت کارلو نیاز به حداقل ده ها هزار بار اجرا دارد تا یک برآورد دقیق بدست آورد، که بسیار وقت گیر است. برای مقابله با این مشکل، جیانگ و همکاران یک تابع EDV سریع بیان شده در رابطه ۱۷ را برای جایگزینی شبیه سازی های انتشار در مدل IC ارائه داده است [69].

$$EDV = \sigma_0(S) + \sigma_1(S) +$$

$$= k$$

$$+ \sum_{i \in N_S^{(1)}/S} (1$$

$$- \prod_{(i,j) \in E, j \in S} (1 - p_{ij})) \quad (17)$$

که در آن $N_S^{(1)}$ نشان دهنده همسایگان مجموعه S است. پارامتر P_{ij}^* احتمال فعال ثابت گره i است که برابر با p در مدل IC است. در روش پیشنهادی از این مدل به عنوان تابع هدف برای تخمین میزان شایستگی هر راه حل استفاده شده است.

۳-۱-۶ مقداردهی اولیه

در روش پیشنهادی جمعیت اولیه از N راه حل تشکیل می شود. برای ایجاد هر راه حل k گره با بیشترین درجه (به صورت مرتب شده بر مبنای درجه شان) به عنوان گره های اولیه پایه IN در نظر گرفته می شود. سپس برای ایجاد تنوع در میان اعضا گره هایی انتخاب و به صورت تصادفی با گره های دیگری از شبکه جایگزین می گردند.

^۹ Seed set

$$A_k^* = \operatorname{argmax}_{A \in \{S \subseteq V; |S|=K\}} \sigma(A) \quad (19)$$

در این بخش، ما الگوریتم پیشنهادی خود را بر روی شبکه‌های دنیای واقعی (بیان شده در جدول) پیاده‌سازی کرده و نتایج به دست آمده با روش‌های موجود مقایسه خواهد شد. ارزیابی‌ها بر روی پلتفرم نرم افزاری MATLAB R2019b تحت سیستم عامل ویندوز ۱۰ با یک پردازنده Intel (R) Core (TM) i7 CPU @ 3.15 انجام شده است.

ما برای ارزیابی روش پیشنهادی خود و مقایسه آن با روش‌های موجود و اخیر شش شبکه مختلف واقعی با سایزهای متفاوتی را در نظر گرفته‌ایم. مشخصات مربوط به شبکه‌های مورد ارزیابی در جدول آورده شده است. که در آن $|V|$ تعداد گره‌ها، $|E|$ تعداد لینک‌ها، $\langle k \rangle$ میانگین درجه نودها، C میانگین ضریب خوشه‌بندی و AC^{10} ضریب تجزیه است. شبکه Netscience [71]، Email [72]، شبکه Ca-GrQc [73] و شبکه Ca-HepTh [74] شبکه‌های ارتباطاتی بین محققان در زمینه‌های مختلف است.

```

12:     end if
13:     if Flag is TRUE
14:         Break;
15:     end if
16: end while
14:  $Xb \leftarrow Xa$ ;
15: end for
16: Output:  $Xb$ .

```

۶-۲- ارزیابی الگوریتم DIWOGSA در حل مساله

بیشینه‌سازی نفوذ

مساله بیشینه‌سازی نفوذ با در نظر گرفتن گراف $G(V,E)$ می‌توان به صورت ریاضی با رابطه ۱۹ تعریف کرد [70]. اگر n و u به ترتیب تعداد المان‌های V و E باشد. K عدد صحیح مثبت $K < n$ است. هدف از مساله بیشینه‌سازی نفوذ، پیدا کردن مجموعه‌ای از K گره با عنوان S ، به عنوان گره‌های اولیه به صورتی که $\sigma(A_k^*) \geq \sigma(S)$ باشد.

^{۱۰} Assortativity Coefficient

گره با بیشترین Betweenness Centrality به عنوان گره‌های اولیه انتخاب می‌شوند.

- PageRank: برای بیشینه‌سازی نفوذ با روش PageRank [78]، K گره با بالاترین PageRank به عنوان گره‌های اولیه در نظر گرفته خواهد شد.

در تمامی ارزیابی‌های انجام گرفته برای محاسبه میزان نفوذ روش انتشار مدل آبخاری مستقل (IC^{۱۵}) با ۱۰۰۰۰ مونت کارلو در نظر گرفته شده است. مدل آبخاری مستقل یکی از مدل‌های انتشار اطلاعات احتمالاتی است که توسط کمپ و همکاران در مقاله [79] برای مساله بیشینه‌سازی نفوذ استفاده شده است. در این مدل، یک گره می‌تواند در حالت فعال یا در حالت غیر فعال باشد. در ابتدا (به عنوان مثال در $t = 0$) همه گره‌ها به جز دانه‌ها غیرفعال هستند. هر گره فعال (مثلاً U_i) در هر دور t فرصتی برای فعال کردن همسایه فعالاً غیرفعال خود ($U_j \in N^{out}(U_i)$) و غیرفعال گره (U_i) با احتمال وزن لبه خود دارد. اگر U_i موفق شود، U_j به یک گره فعال در دور $t + 1$ تبدیل می‌شود. یک گره می‌تواند حالت خود را از غیر فعال به فعال تغییر دهد اما از حالت فعال به غیر فعال تبدیل نمی‌شود. این روند آبخاری تا زمانی که گره فعال دیگری در دور وجود نداشته باشد ادامه خواهد یافت.

۶-۲-۱- مقایسه میزان نفوذ

برای محاسبه میزان نفوذ روش انتشار مدل آبخاری با ۱۰۰۰۰ مونت کارلو در نظر گرفته شده است. در الگوریتم DPSO مطابق مقاله [76] مقادیر پارامترهای C_1 و C_2 برابر با ۲، ω برابر با ۰.۸، n و g_{max} برابر با ۱۰۰ در نظر گرفته شده است. الگوریتم CELF با ۱۰۰۰۰ مونت کارلو مورد ارزیابی قرار گرفته است. در ارزیابی الگوریتم DIWOGSA مقدار پارامتر جمعیت اولیه و $Imax$ به ترتیب برابر ۲۰ و ۲۵۰ است. S_{max} برای شبکه‌های Email، Netscience و NetGrQc برابر ۵ و برای سایر شبکه‌ها با سایز بیشتر برابر با ۲۰ در

جدول ۱۳. مشخصات مربوط به شبکه‌های واقعی

Datasets	V	E	$\langle k \rangle$	C	AC
Netscience ^{۱۱}	379	914	4.823	0.798	- 0.082
Email ^{۱۲}	1133	5451	9.622	0.254	0.078
Ca-GRQC ^{۱۳}	5242	14496	5.53	0.5296	0.659
Ca-HepTH ^{۱۴}	9877	25998	5.264	0.600	0.268

روش‌هایی که برای مقایسه با روش پیشنهادی در نظر گرفته شده‌اند شامل الگوریتم CELF، DPSO، Single Discount، Degree، Betweenness Centrality و PageRank است. در ادامه توضیح مختصری از هر کدام بیان شده است.

- CELF: الگوریتم "Cost-Efficient Lazy Forward" یک الگوریتم حریصانه بهبود یافته با استراتژی بهینه‌سازی "Lazy forward" است [75]. این الگوریتم ۷۰۰ برابر سریعتر از الگوریتم greedy است.
- DPSO: الگوریتم DPSO یک الگوریتم بیشینه‌سازی نفوذ مبتنی بر بهینه‌سازی ازدحام ذرات گسسته است [76]. در این روش از جستجوی محلی برای افزایش نفوذ استفاده شده است.
- Single Discount: در این روش در هر تکرار از الگوریتم گره‌ای با بیشترین درجه انتخاب می‌شود سپس درجه همسایگان آن گره یک واحد کاهش پیدا می‌کند [77].
- Degree: در این روش K گره با بیشترین درجه به عنوان گره‌های اولیه انتخاب می‌شوند.
- Betweenness Centrality: معیار Betweenness به جای توجه به خود گره‌ها در شبکه‌ها، ساختار شبکه را برای تشخیص گره‌های با نفوذ در نظر می‌گیرد. در این روش K

^{۱۱} <http://networkrepository.com/ca-netscience.php>

^{۱۲} <https://www.cc.gatech.edu/dimacs10/archive/clustering.shtml>

^{۱۳} <https://snap.stanford.edu/data/ca-GrQc.html>

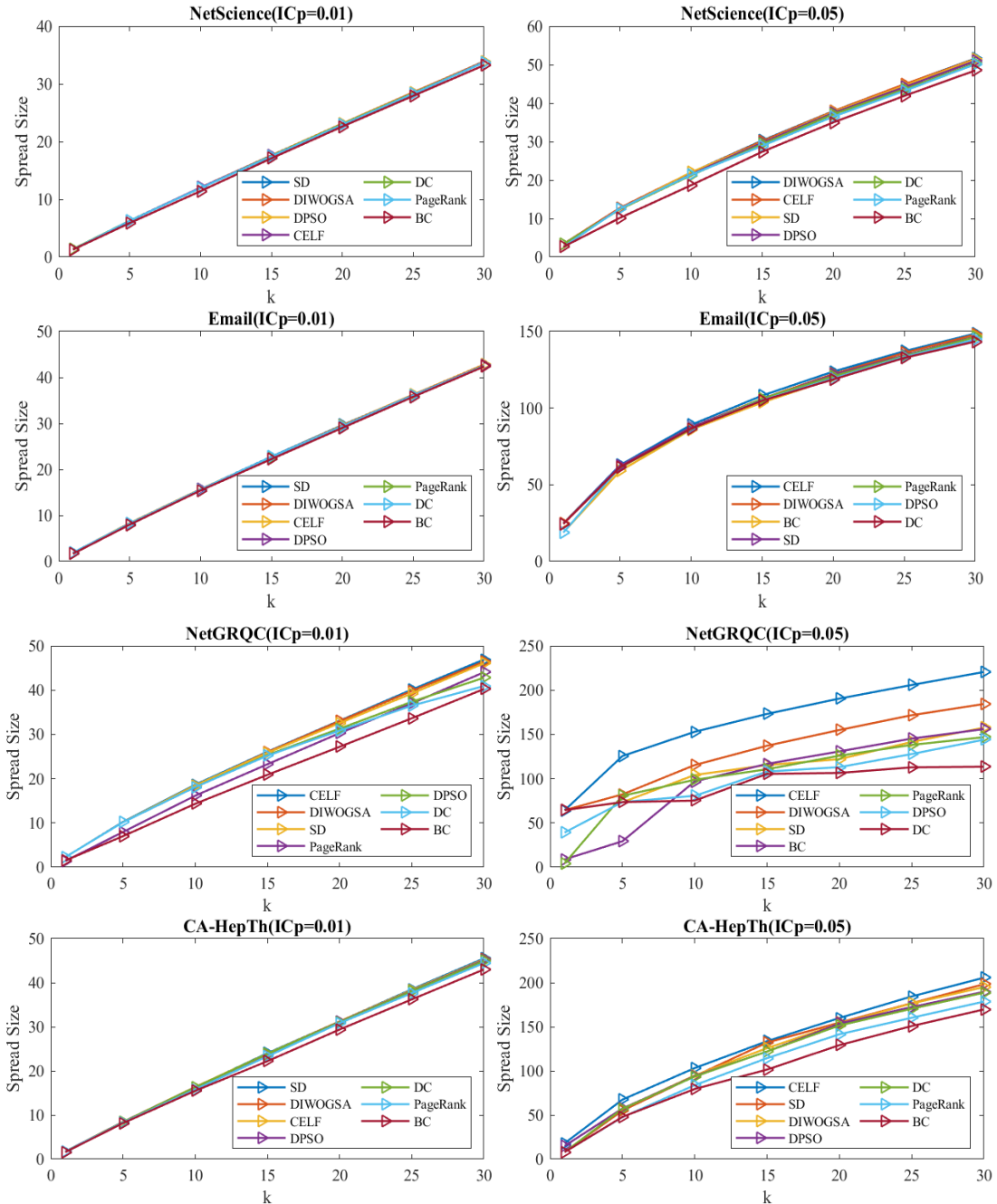
^{۱۴} <https://snap.stanford.edu/data/ca-HepTh.html>

^{۱۵} Independent Cascade Model

روش‌ها با مقدار احتمال انتشار ۰,۰۱ و ۰,۰۵ در شکل نشان داده شده است.

مطابق نتایج به دست آمده در شکل الگوریتم greedy-based CELF در بیشتر شبکه‌ها از همه‌ی روش‌های مورد مقایسه نتایج بهتری را داشته است. الگوریتم DIWOGSA در بیشتر شبکه‌ها به عنوان دومین بهترین بوده است. در مقایسه الگوریتم DIWOGSA با الگوریتم DPSO، الگوریتم DIWOGSA در تمامی شبکه‌ها نسبت به الگوریتم DPSO میزان نفوذ بالاتری را دارد.

نظر گرفته شده است. پارامتر $\sigma_{2initial} \cdot \sigma_{2final}$ به ترتیب برابر با تعداد گره‌های موجود در راه‌حل و ۱ است. پارامترهای $\sigma_{2initial} \cdot \sigma_{2final}$ به ترتیب برابر با ۰,۵ و ۰,۱ بوده و سایر پارامترها همانند روش پیوسته از جدول در نظر گرفته شده است. روش‌های فرا اکتشافی DPSO و DIWOGSA به تعداد ۳۰ بار به صورت مستقل اجرا شده و میانگین نتایج در نمودارها آورده شده است. ارزیابی‌ها با در نظر گرفتن تعداد گره‌های اولیه مختلف $k=\{1, 5, 10, 15, \dots, 30\}$ صورت گرفته و نتایج ارزیابی‌ها در مقایسه با سایر

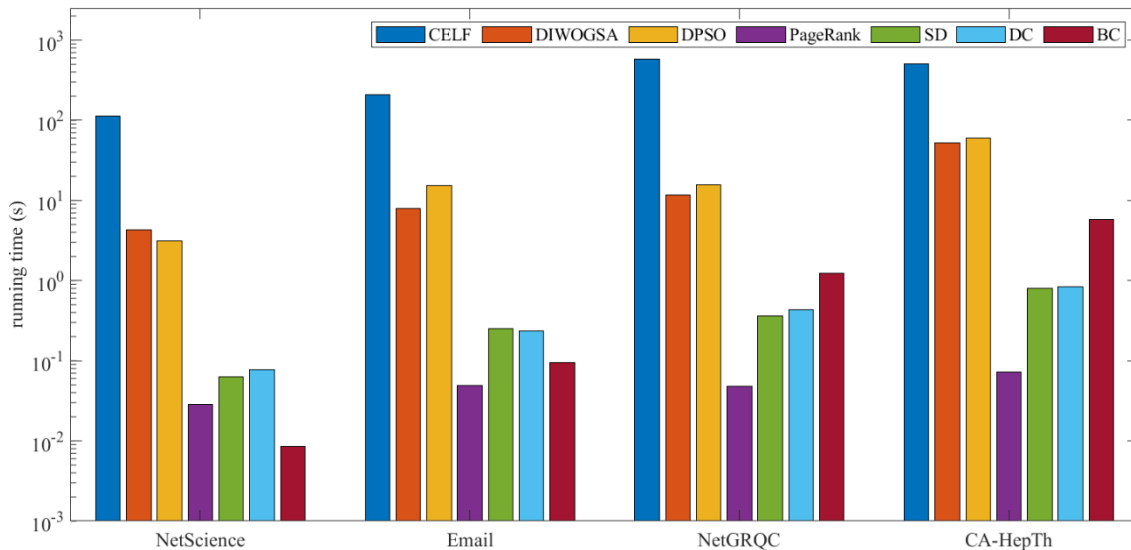


شکل ۷. میزان نفوذ روش پیشنهادی در مقایسه با سایر روش‌ها با در نظر گرفتن روش مدل آبخاری با احتمال انتشار ۰,۰۵

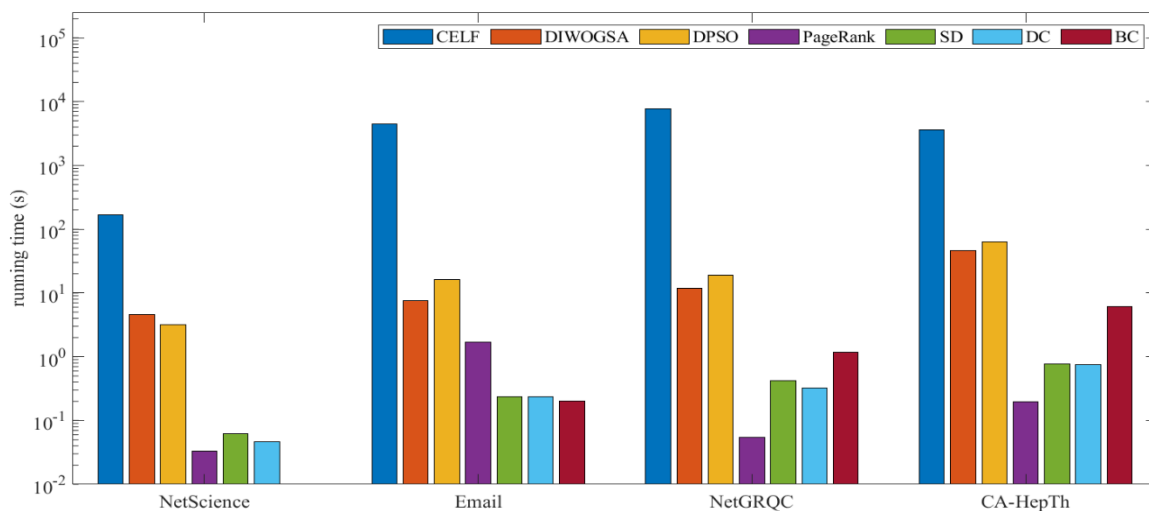
۱-۲-۶- مقایسه زمان اجرا

شده به طور میانگین، روش PageRank کمترین زمان اجرا و الگوریتم CELF بیشترین زمان اجرا را دارد. همچنین روش پیشنهادی نسبت به الگوریتم CELF در تمامی شبکه‌ها و نسبت به DPSO در تمامی شبکه‌ها به غیر از NetScience زمان اجرای کمتری را داشته است.

مدت زمان اجرا روش پیشنهادی در مقایسه با سایر روش‌ها با در نظر گرفتن روش مدل آبخاری با مقدار احتمال ۰,۰۱ و ۰,۰۵ به ترتیب در شکل ۷ و شکل ۸ نشان داده شده است. مطابق نمودار رسم



شکل ۷. مدت زمان اجرا روش پیشنهادی در مقایسه با سایر روش‌ها با در نظر گرفتن روش مدل آبخاری با مقدار احتمال ۰,۰۱



شکل ۸. مدت زمان اجرا روش پیشنهادی در مقایسه با سایر روش‌ها با در نظر گرفتن روش مدل آبخاری با مقدار احتمال ۰,۰۵

شده در رتبه‌بندی فریدمن برابر با $3.26e-06$ است. چون این مقدار کمتر از ۰,۰۵ می‌باشد می‌توان نتیجه گرفت که بین نتایج الگوریتم‌های مختلف تفاوت معنی داری وجود دارد. مطابق نتایج به دست آمده الگوریتم پیشنهادی DIWOGSA در کنار greedy-based CELF رتبه اول را در مقایسه با سایر روش‌های به دست آورده‌اند.

۲-۲-۶- تحلیل آماری

رتبه‌بندی فریدمن روش‌های مختلف تشخیص نفوذ، با در نظر گرفتن میزان نفوذ روش‌های مختلف با در نظر گرفتن $k=30$ با مقدار احتمال انتشار اطلاعات ۰,۰۱ و ۰,۰۵ در پایگاه داده‌های مختلف صورت گرفته است. میانگین رتبه‌های هر روش به همراه رتبه نهایی آن روش در جدول نشان داده شده است. مقدار P-value محاسبه

جدول ۱۴. رتبه‌بندی الگوریتم‌های مختلف با آزمون رتبه‌بندی

Algorithm	Average Ranking	Final Rank
CELF	6	1
DIWOGSA	6	1
DPSO	2.625	3
Page Rank	3.5	4
Single Discount	5.75	2
Degree Centrality	2.25	5
Betweenness Centrality	1.875	6

۷- نتیجه‌گیری

الگوریتم IWOOGSA یک روش بهینه‌سازی ترکیبی از الگوریتم علف‌های هرز و الگوریتم جستجوی گرانشی برای حل مسائل بهینه‌سازی می‌باشد. برای بررسی کارایی روش پیشنهادی این روش را برای حل ۲۹ تابع مختلف بنچمارک در چهار دسته توابع تک هدفه، توابع چندهدفه، توابع چندهدفه با ابعاد ثابت و توابع کامپوزیت به کار گرفته‌ایم. همچنین برای بررسی کارایی روش پیشنهادی ما این روش را با الگوریتم‌های جدید و رایج در این زمینه به نام‌های الگوریتم بهینه‌سازی ازدحام ذرات (PSO)، الگوریتم علف‌های هرز (IWO)، بهینه‌سازی تعادل (EO)، الگوریتم ازدحام سالپ (SSA) و الگوریتم وال (WOA)، الگوریتم جستجوی گرانشی (GSA)، روش برنامه‌نویسی تکاملی سریع (FEP) و الگوریتم ژنتیک (GA) و دو روش SHADE و LSHADE-SPACMA مقایسه کرده‌ایم. مطابق نتایج به دست آمده روش پیشنهادی در مقایسه با روش‌های موجود مطرح شده در این مقاله با روش رتبه‌بندی فریدمن به عنوان بهترین روش معرفی شده است. همچنین این روش برای حل سه مساله مهندسی شناخته شده طراحی مجرای فشار، طراحی پرتو جوش و مینیمم‌سازی وزن یک مورد استفاده قرار گرفته و میزان کارایی آن در حل چنین مسائلی مورد ارزیابی قرار گرفته شده است. در مقایسه با روش‌های موجود از میان سه مساله مطرح در زمینه مهندسی روش پیشنهادی در دو مورد از آن‌ها به بهترین نتیجه دست یافته است.

همچنین در این مقاله مدل گسسته از الگوریتم IWOOGSA به نام DIWOGSA برای حل مساله پیشینه‌سازی نفوذ پیشنهاد شده است. روش پیشنهادی با روش‌های موجود و رایج در این زمینه مقایسه شده است. مطابق رتبه‌بندی فریدمن الگوریتم پیشنهادی DIWOGSA بعد از الگوریتم حریمانه CELF رتبه دوم را کسب کرده است. از نظر زمان اجرا الگوریتم پیشنهادی بسیار سریعتر از

الگوریتم CELF است. همچنین روش پیشنهادی نسبت به DPSO نیز در بیشتر دیتاست‌ها زمان اجرای کمتری را دارد. به عنوان کارهای آتی، می‌توان روش پیشنهادی IWOOGSA را برای حل مسائل باینری و چندهدفه طراحی کرد.

مراجع

- [1] Y. Chen and D. Pi, "An innovative flower pollination algorithm for continuous optimization problem," *Applied Mathematical Modelling*, vol. 83, pp. 237-265, 2020/07/01/ 2020.
- [2] Z. Cheng, H. Song, J. Wang, H. Zhang, T. Chang, and M. Zhang, "Hybrid firefly algorithm with grouping attraction for constrained optimization problem," *Knowledge-Based Systems*, vol. 220, p. 106937, 2021/05/23/ 2021.
- [3] J. Jiang, Y. Liu, and Z. Zhao, "TriTSA: Triple Tree-Seed Algorithm for dimensional continuous optimization and constrained engineering problems," *Engineering Applications of Artificial Intelligence*, vol. 104, p. 104303, 2021/09/01/ 2021.
- [4] A. Mortazavi, "Bayesian Interactive Search Algorithm: A New Probabilistic Swarm Intelligence Tested on Mathematical and Structural Optimization Problems," *Advances in Engineering Software*, vol. 155, p. 102994, 2021/05/01/ 2021.
- [5] Z.-k. Feng, W.-j. Niu, and S. Liu, "Cooperation search algorithm: A novel metaheuristic evolutionary intelligence algorithm for numerical optimization and engineering optimization problems," *Applied Soft Computing*, vol. 98, p. 106734, 2021/01/01/ 2021.
- [6] M. Abdel-Basset, D. El-Shahat, M. Jameel, and M. Abouhawwash, "Young's double-slit experiment optimizer : A novel metaheuristic optimization algorithm for global and constraint optimization problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 403, p. 115652, 2023/01/01/ 2023.
- [7] M. Dehghani, Z. Montazeri, E. Trojovská, and P. Trojovský, "Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving

- problems," *Applied Soft Computing*, vol. 93, p. 106391, 2020/08/01/ 2020.
- [18] Q. Liu, L. Wu, W. Xiao, F. Wang, and L. Zhang, "A novel hybrid bat algorithm for solving continuous optimization problems," *Applied Soft Computing*, vol. 73, pp. 67-82, 2018/12/01/ 2018.
- [19] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition," in *2007 IEEE Congress on Evolutionary Computation*, 2007, pp. 4661-4667.
- [20] E. Bogar and S. Beyhan, "Adolescent Identity Search Algorithm (AISA): A novel metaheuristic approach for solving optimization problems," *Applied Soft Computing*, vol. 95, p. 106503, 2020/10/01/ 2020.
- [21] M. A. Eita and M. M. Fahmy, "Group counseling optimization," *Applied Soft Computing*, vol. 22, pp. 585-604, 2014/09/01/ 2014.
- [22] A. Kaveh and V. R. Mahdavi, "Colliding bodies optimization: A novel metaheuristic method," *Computers & Structures*, vol. 139, pp. 18-27, 2014/07/15/ 2014.
- [23] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, vol. 1, no. 4, pp. 355-366, 2006/12/01/ 2006.
- [24] A. Ouyang, X. Peng, Q. Wang, and Y. Wang, "Discrete invasive weed optimization algorithm for traveling salesman problems," in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, 2017, pp. 523-528.
- [25] S. Roy, M. I. Sk, S. Das, S. Ghosh, and A. Vasilakos, "A simulated weed colony system with subregional differential evolution for multimodal optimization," *Engineering Optimization - ENG OPTIMIZ*, vol. 45, pp. 1-23, 11/30 2012.
- [26] D. Kundu, K. Suresh, S. Ghosh, S. Das, B. K. Panigrahi, and S. Das, "Multi-objective optimization with artificial weed colonies," optimization problems," *Knowledge-Based Systems*, vol. 259, p. 110011, 2023/01/10/ 2023.
- [8] C. Li, K. Liang, Y. Chen, and M. Pan, "An exploitation-boosted sine cosine algorithm for global optimization," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105620, 2023/01/01/ 2023.
- [9] J. R. Koza, *Genetic programming*. MIT Press, 55 Hayward St., Cambridge, MA, United States, 1992.
- [10] K. Price, R. Storn, and J. A. Lampinen, *Differential Evolution, A Practical Approach to Global Optimization (Natural Computing Series)*. Springer-Verlag Berlin Heidelberg, 2005.
- [11] I. Rechenberg, "Evolution Strategy: Nature's Way of Optimization," in *Optimization: Methods and Applications, Possibilities and Limitations*, Berlin, Heidelberg, 1989, pp. 106-126: Springer Berlin Heidelberg.
- [12] I. M. Ali, D. Essam, and K. Kasmarik, "A novel differential evolution mapping technique for generic combinatorial optimization problems," *Applied Soft Computing*, vol. 80, pp. 297-309, 2019/07/01/ 2019.
- [13] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," *Knowledge-Based Systems*, vol. 191, p. 105190, 2020/03/05/ 2020.
- [14] W. Al-Sorori and A. M. Mohsen, "New Caledonian crow learning algorithm: A new metaheuristic algorithm for solving continuous optimization problems," *Applied Soft Computing*, vol. 92, p. 106325, 2020/07/01/ 2020.
- [15] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46-61, 2014/03/01/ 2014.
- [16] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28-39, 2006.
- [17] X. Chu *et al.*, "An artificial bee colony algorithm with adaptive heterogeneous competition for global optimization

- [36] J. J. Liang, P. N. Suganthan, and K. Deb, "Novel composition test functions for numerical global optimization," in *Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005. SIS 2005.*, 2005, pp. 68-75.
- [37] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, 1995, vol. 4, pp. 1942-1948 vol.4.
- [38] Y. Xin, L. Yong, and L. Guangming, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82-102, 1999.
- [39] C. A. Coello Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Computers in Industry*, vol. 41, no. 2, pp. 113-127, 2000/03/01/ 2000.
- [40] R. Tanabe and A. Fukunaga, "Success-history based parameter adaptation for Differential Evolution," in *2013 IEEE Congress on Evolutionary Computation*, 2013, pp. 71-78.
- [41] A. W. Mohamed, A. A. Hadi, A. M. Fattouh, and K. M. Jambi, "LSHADE with semi-parameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems," in *2017 IEEE Congress on Evolutionary Computation (CEC)*, 2017, pp. 145-152.
- [42] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163-191, 2017/12/01/ 2017.
- [43] H. Chen, M. Wang, and X. Zhao, "A multi-strategy enhanced sine cosine algorithm for global optimization and constrained practical engineering problems," *Applied Mathematics and Computation*, vol. 369, p. 124872, 2020/03/15/ 2020.
- [44] S. H. Samareh Moosavi and V. K. Bardsiri, "Poor and rich optimization algorithm: A new human-based and multi populations algorithm," *Engineering Applications of Information Sciences*, vol. 181, no. 12, pp. 2441-2454, 2011/06/15/ 2011.
- [27] P. Pahlavani, M. R. Delavar, and A. U. Frank, "Using a modified invasive weed optimization algorithm for a personalized urban multi-criteria path optimization problem," *International Journal of Applied Earth Observation and Geoinformation*, vol. 18, pp. 313-328, 2012/08/01/ 2012.
- [28] GhoshArnob, DasSwagatam, ChowdhuryAritra, and GiriRitwik, "An ecologically inspired direct search method for solving optimal control problems with Bézier parameterization," *Engineering Applications of Artificial Intelligence*, 2011.
- [29] A. Ojha and R. Yenugula, "Hybridizing Particle Swarm Optimization with Invasive Weed Optimization for Solving Nonlinear Constrained Optimization Problems," *Advances in Intelligent Systems and Computing*, vol. 336, pp. 595-606, 12/01 2015.
- [30] M. R. Panda, S. Dutta, and S. Pradhan, "Hybridizing Invasive Weed Optimization with Firefly Algorithm for Multi-Robot Motion Planning," *Arabian Journal for Science and Engineering*, vol. 43, no. 8, pp. 4029-4039, 2018/08/01 2018.
- [31] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232-2248, 2009/06/13/ 2009.
- [32] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "BGSA: binary gravitational search algorithm," *Natural Computing*, vol. 9, no. 3, pp. 727-745, 2010/09/01 2010.
- [33] M. Dehbashian, Advanced Elitism gravitational Search Algorithm (AEGSA). 2010.
- [34] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *Trans. Evol. Comp*, vol. 1, no. 1, pp. 67-82, 1997.
- [35] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51-67, 2016/05/01/ 2016.

- [53] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 36, pp. 3902-3933, 2005/09/23/ 2005.
- [54] K. M. Ragsdell and D. T. Phillips, "Optimal Design of a Class of Welded Structures Using Geometric Programming," *Journal of Engineering for Industry*, vol. 98, no. 3, pp. 1021-1025, 1976.
- [55] S. Akhtar, K. Tai, and T. Ray, "A socio-behavioural simulation model for engineering design optimization," *Engineering Optimization*, vol. 34, no. 4, pp. 341-354, 2002/01/01 2002.
- [56] T. Ray and K. M. Liew, "Society and civilization: An optimization algorithm based on the simulation of social behavior," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 4, pp. 386-396, 2003.
- [57] J. Zhang, C. Liang, Y. Huang, J. Wu, and S. Yang, "An effective multiagent evolutionary algorithm integrating a novel roulette inversion operator for engineering optimization," *Applied Mathematics and Computation*, vol. 211, no. 2, pp. 392-416, 2009/05/15/ 2009.
- [58] A.-R. Hedar and M. Fukushima, "Derivative-Free Filter Simulated Annealing Method for Constrained Continuous Global Optimization," *Journal of Global Optimization*, vol. 35, no. 4, pp. 521-549, 2006/08/01 2006.
- [59] S. He, E. Prempan, and Q. H. Wu, "An improved particle swarm optimizer for mechanical design optimization problems," *Engineering Optimization*, vol. 36, no. 5, pp. 585-605, 2004/10/01 2004.
- [60] S.-F. Hwang and R.-S. He, "A hybrid real-parameter genetic algorithm for function optimization," *Advanced Engineering Informatics*, vol. 20, no. 1, pp. 7-21, 2006/01/01/ 2006.
- [61] F.-z. Huang, L. Wang, and Q. He, "An effective co-evolutionary differential evolution for constrained optimization," *Artificial Intelligence*, vol. 86, pp. 165-181, 2019/11/01/ 2019.
- [45] S. Balochian and H. Baloochian, "Social mimic optimization algorithm and engineering applications," *Expert Systems with Applications*, vol. 134, pp. 178-191, 2019/11/15/ 2019.
- [46] Q. He and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 1, pp. 89-99, 2007/02/01/ 2007.
- [47] H. S. Bernardino, H. J. C. Barbosa, A. C. C. Lemonge, and L. G. Fonseca, "A new hybrid AIS-GA for constrained optimization problems in mechanical engineering," in *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, 2008, pp. 1455-1462.
- [48] V. S. Aragón, S. C. Esquivel, and C. A. C. Coello, "A modified version of a T-Cell Algorithm for constrained optimization problems," *International Journal for Numerical Methods in Engineering*, vol. 84, no. 3, pp. 351-378, 2010/10/15 2010.
- [49] C. Coello, "Use of dominance-based tournament selection to handle constraints in genetic algorithms," *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 11, 01/01 2001.
- [50] C. A. C. Coello and N. C. Cortés, "Hybridizing a genetic algorithm with an artificial immune system for global optimization," *Engineering Optimization*, vol. 36, no. 5, pp. 607-634, 2004/10/01 2004.
- [51] E. M. Montes and B. H. Oca "Bacterial foraging for engineering design problems: preliminary results," *4th Mex. Congr. Evol. Comput. COMCEV'2008, Mexico,,* pp. 33-38, 2008.
- [52] E. Mezura-Montes and C. A. C. Coello, "An empirical study about the usefulness of evolution strategies to solve constrained optimization problems," *International Journal of General Systems*, vol. 37, no. 4, pp. 443-473, 2008/08/01 2008.

- Artificial Intelligence, San Francisco, California, 2011.
- [70] W. Chen, L. V. S. Lakshmanan, and C. Castillo, "Information and Influence Propagation in Social Networks," *Synthesis Lectures on Data Management*, vol. 5, no. 4, pp. 1-177, 2013/10/27 2013.
- [71] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, p. 036104, 09/11/ 2006.
- [72] R. Guimerà, L. Danon, A. Díaz-Guilera, F. Giralt, and A. Arenas, "Self-similar community structure in a network of human interactions," *Physical Review E*, vol. 68, no. 6, p. 065103, 12/17/ 2003.
- [73] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Discov. Data*, vol. 1, no. 1, pp. 2–es, 2007.
- [74] F. Lu, W. Zhang, L. Shao, X. Jiang, P. Xu, and H. Jin, "Scalable influence maximization under independent cascade model," *Journal of Network and Computer Applications*, vol. 86, pp. 15-23, 2017/05/15/ 2017.
- [75] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," presented at the Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, San Jose, California, USA, 2007. Available: <https://doi.org/10.1145/1281192.1281239>
- [76] M. Gong, J. Yan, B. Shen, L. Ma, and Q. Cai, "Influence maximization in social networks based on discrete particle swarm optimization," *Information Sciences*, vol. 367-368, pp. 600-614, 2016/11/01/ 2016.
- [77] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009. Available: <https://doi.org/10.1145/1557019.1557047>
- Applied Mathematics and Computation*, vol. 186, no. 1, pp. 340-356, 2007/03/01/ 2007.
- [62] A. D. Belegundu and J. S. Arora, "A study of mathematical programming methods for structural optimization. Part I: Theory," *International Journal for Numerical Methods in Engineering*, vol. 21, no. 9, pp. 1583-1599, 1985/09/01 1985.
- [63] T. Ray and P. Saini, "ENGINEERING DESIGN OPTIMIZATION USING A SWARM WITH AN INTELLIGENT INFORMATION SHARING AMONG INDIVIDUALS," *Engineering Optimization*, vol. 33, no. 6, pp. 735-748, 2001/08/01 2001.
- [64] C. A. Coello Coello and E. Mezura Montes, "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection," *Advanced Engineering Informatics*, vol. 16, no. 3, pp. 193-203, 2002/07/01/ 2002.
- [65] K. Zhang, H. Du, and M. W. Feldman, "Maximizing influence in a social network: Improved results using a genetic algorithm," *Physica A: Statistical Mechanics and its Applications*, vol. 478, pp. 20-30, 2017/07/15/ 2017.
- [66] M. T. Nguyen and K. Kim, "Genetic convolutional neural network for intrusion detection systems," *Future Generation Computer Systems*, vol. 113, pp. 418-427, 2020/12/01/ 2020.
- [67] J. Jabari Lotf, M. Abdollahi Azgomi, and M. R. Ebrahimi Dishabi, "An improved influence maximization method for social networks based on genetic algorithm," *Physica A: Statistical Mechanics and its Applications*, vol. 586, p. 126480, 2022/01/15/ 2022.
- [68] G. Iacca, K. Konotopska, D. Bucur, and A. Tonda, "An evolutionary framework for maximizing influence propagation in social networks," *Software Impacts*, vol. 9, p. 100107, 2021/08/01/ 2021.
- [69] Q. Jiang, G. Song, G. Cong, Y. Wang, W. Si, and K. Xie, "Simulated annealing based influence maximization in social networks," presented at the Proceedings of the Twenty-Fifth AAAI Conference on

- through a social network," presented at the Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, Washington, D.C., 2003. Available: <https://doi.org/10.1145/956750.956769>
- [78] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Computer Networks and ISDN Systems*, vol. 30, no. 1, pp. 107-117, 1998/04/01/ 1998.
- [79] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence