یازدهمین کنگره ملی سراسری
فناوریهای نوین در حوزه توسعه پایدار ایران
11th National Congress of
the New Technologies in Sustainable Development of Iran
senaconf.ir

# پرکاربردترین الگوریتم های فرااکتشافی مبتنی بر جمعیت و کاربردهایشان

سمیرا درچالیان[1]، راضیه عسگرنژاد[2]

[1]دانشجوی کارشناسی مهندسی نرم افزار، دانشکده مهندسی کامپیوتر، دانشگاه دولتی فنی و حرفه ای دختران فدک، کاشان، ایران
samiradorchalian@gmail.com

[2] دکتری مهندسی نرم افزار، دانشکده مهندسی کامپیوتر، دانشگاه دولتی فنی و حرفه ای دختران فدک، کاشان، ایران
razyehan@gmail.com

**چکیده**

با توجه به الگوریتم های موجود و کارایی و خواص هایی که دارند میتوان با ترکیب این الگوریتم ها که شامل هوش جمعی، محاصره دسته جمعی، پروسه پیدا کردن کوتاه ترین مسیر،که قابل توسعه این سیستم ها با جمع آوری راه حل های تصادفی و جستجو برای بهینه سازی با به روز رسانی نسل های اینده امید داشت و سیستم های بهینه ی بهتری را توسعه و پیاده سازی کرد. در این مقاله انواع الگوریتم های فرااکتشافی موجود را بررسی و مقایسه کرده ایم تا بتوان راه حل های دیگری با ترکیب الگوریتم های موجود به طور یکنواخت و پیوسته و کارایی الگوریتم ها روش جدیدی را پیاده سازی کنیم.

**واژههای کلیدی**

الگوریتم فرااکتشافی، کروموزوم، ژن، جمعیت، عدد برازندگی ، عملگر جابه جابی یا ترتیب

## 1. مقدمه

الگوریتمهای فراابتکاری یا فرافراالکتشافی یا فراالکتشافی نوعی از الگوریتمهای تصادفی هستند که برای یافتن پاسخ بهینه به کار میروند.

روشها و الگوریتمهای بهینهسازی به دو دسته الگوریتمهای دقیق(exact) و الگوریتمهای تقریبی (approximate algorithms) تقسیمبندی میشوند. الگوریتمهای دقیق قادر به یافتن جواب بهینه به صورت دقیق هستند اما در مورد مسائل بهینهسازی سخت کارایی کافی ندارند و زمان اجرای آنها متناسب با ابعاد مسائل به صورت نمایی افزایش مییابد. الگوریتمهای تقریبی قادر به یافتن جوابهای خوب )نزدیک به بهینه( در زمان حل کوتاه برای مسائل بهینهسازی سخت هستند. الگوریتمهای تقریبی نیز به سه دسته الگوریتمهای ابتکاری (heuristic) و فراابتکاری (meta-heuristic) و فوق ابتکاری (hyper heuristic) بخشبندی میشوند. دو مشکل اصلی الگوریتمهای ابتکاری، گیر افتادن آنها در نقاط بهینه محلی، همگرایی زودرس به این نقاط است. الگوریتمهای فراابتکاری برای حل این مشکلات الگوریتمهای ابتکاری ارائه شدهاند. درواقع الگوریتمهای فراابتکاری، یکی از انواع الگوریتمهای بهینهسازی تقریبی هستند که دارای راهکارهای برونرفت از نقاط بهینه محلی هستند و قابلیت کاربرد در طیف گستردهای از مسائل را دارند ]1[.

در میان مجموعه ای از تکنیک های جستجو و بهینه سازی، توسعه یا الگوریتم های فراالکتشافی )Evolutionary Algorithms( در دهه گذشته بسیار مهم بوده است. الگوریتم های فراالکتشافی مجموعه ای از اکتشافات مدرن متشکل از موفقیت در بسیاری از برنامه های کاربردی با پیچیدگی های زیادی است. الگوریتم های فراالکتشافی با یک جمعیّت اولیّه کار خود را شروع میکنند و با تکثیر نسلهای بعدی از روی والدین، سعی در حل مسائل پیچیده به بهترین شکل دارند. این الگوریتمها که زیرشاخهای از علم هوش مصنوعی هستند را میتوان در علوم نظیر برق، مکانیک، صنایع، زیستشناسی، شیمی و رباتیک به کار گرفت. الگوریتمهای فراالکتشافی از روش ها و عملیات ابتدایی برای حل مسئله استفاده میکنند و در طی یک سری از تکرارها به راهحل مناسب برای مسئله میرسند. این الگوریتمها غالباً از یک جمعیت حاوی راهحلهای تصادفی شروع میکنند و در طی هر مرحله تکرار سعی در بهتر کردن مجموعه راهحلها دارند. در آغاز کار ،تعدادی از اعضای جامعه بهصورت تصادفی حدس زدهشده، سپس تابع هدف یا برازندگی برای هر یک از این اعضا محاسبه و نخستین نسل ایجاد خواهد شد. اگر هیچیک از معیارهای خاتمه بهینهسازی دیده نشوند، ایجاد نسل جدید آغاز خواهد شد. اعضا برحسب میزان شایستگیشان برای تولید فرزندها انتخاب میشوند. این افراد بهعنوان والدین محسوب میشوند و بازترکیب فرزندان را تولید مینمایند .

سپس تمامی فرزندها با یک مقدار معینی از احتمال، یعنی همان جهش، تغییر ژنتیکی مییابند. اکنون میزان شایستگی )برازندگی( فرزندان تعیین و در اجتماع جایگزین والدین شده و نسل جدید را ایجاد مینمایند. این چرخه آنقدر تکرار میشود تا یکی از معیارهای پایان بهینهسازی کسب شود ]2[ ]3[ ]4[ ]5[.

## 2. تعاریف و مراحل اولیه

فرآیند تکامل، از طریق انتخاب طبیعی موجودات در یک جمعیت، را میتوان به صورت یک جستجو در فضای مقادیر ممکن کروموزومها در نظر گرفت.

### 1.2. مراحل ایجاد یک الگوریتم فراالکتشافی

مرحله 1( تولید جمعیت اولیه: در این مرحله، یک جمعیت از کروموزومها یا همان پاسخهای مسئله تولید میشود. بدیهی است که قبل از تولید جمعیت اولیه بایستی نحوه نمایش کروموزومها، که یک پاسخ برای مسئله مورد بررسی است، تعیین شود. خروجی مرحله اول یک جمعیت از پاسخها میباشد.

مرحله 2( محاسبه برازش جمعیت ورودی: در این مرحله، برازش تکتک کروموزومهای جمعیت تولیدشده، با توجه به تابع برازش تعیین شده، محاسبه میگردد. خروجی این مرحله، یک جمعیت از پاسخهای ارزیابی شده میباشد.

مرحله 3( انتخاب برای تولید مثل: در این مرحله، آن دسته از اعضا جمعیت ورودی که برازش بالاتری نسبت به سایر اعضا دارند، برای تبیین قانون بقاء اصلح داروین، انتخاب میشوند. خروجی این مرحله، جمعیتی از والدین برازنده است.

مرحله 4( باز ترکیب والدین انتخاب شده: در این مرحله دو کروموزوم با استفاده از عملگر بازترکیب، جمعیتی از فرزندان تولید میشوند. اعمال عملگر بازترکیب با توجه به یک پارامتر احتمالی انجام میشود. خروجی این مرحله جمعیتی از فرزندان تولید شده میباشد.

مرحله 5( جهش فرزندان تولید شده در این مرحله فرزندان تولید شده قبل تحت عملگر جهش قرار میگیرند. البته عملگر جهش با یک احتمال بر روی دنباله ژنی فرزندان رخ میدهد. این احتمال با نام احتمال جهش شناخته میشود. خروجی این مرحله جمعیتی از فرزندان جهش یافته است.

مرحله 6( محاسبه برازش جمعیت فرزندان: در این مرحله، شایستگی فرزندان جهش یافته، با استفاده از تابع برازش، محاسبه میشود .

خروجی در این مرحله، جمعیت فرزندان ارزیابی شده است.

مرحله۷( انتخاب برای جایگزینی: در این مرحله، با توجه به جمعیت والدین)ورودی مرحله سوم( و جمعیت فرزندان ارزیابی شده)خروجی مرحله ششم( یک جمعیت جدید برای نسل بعد تولید میشود. بدیهی است که خروجی این مرحله جمعیتی است که در آن بخشی از والدین نسل قبل و تعدادی از فرزندان جدید تولید شده نسل جاری وجود دارند.

مرحله 8( بررسی شرط توقف: در این مرحله، در مورد ادامه فرآیند فرااکتشافی الگوریتم تصمیمگیری میشود. در صورت عدم ارضاء شرط توقف، فرآیند فرااکتشافی الگوریتم با رجوع به مرحله 3 ادامه مییابد. در غیر اینصورت الگوریتم متوقف شده و بهترین پاسخ در آخرین نسل به عنوان حاصل جستجوی فرااکتشافی در خروجی ارائه میشود.

### 2.2. مفاهیم مهم در الگوریتمهای فرااکتشافی

1- کروموزوم(Chromosome): اطلاعات ژنی یک موجود در کروموزوم ذخیره میشود. هر کروموزوم از DNA تشکیل میشود. در الگوریتم های فرااکتشافی منظور از کروموزوم، یک رشته، گراف، درخت یا دنبالهای از صفر و یکها میباشد که معادل یک پاسخ برای حل یک مسئله میباشند.

2- ژن(Gene): کروموزومها به چندین قسمت تقسیم میشوند که ژن نام دارند. ژنها خصوصیات گونهها یا افراد را تشکیل میدهند. در یک الگوریتم فرااکتشافی، هر پاسخ از چندین بخش تشکیل شده است که به هر کدام از آن بخشها یک مشخصه یا ویژگی گفته میشود .

مشخصه یا ویژگی در هر پاسخ معادل ژن در کروموزوم است.

3- آلل)Allele(: هر ژن مجموعهای از مقادیر مجاز را میتواند اختیار کند. به هر کدام از این مقادیر یک آلل گفته میشود. به عنوان مثال یک ژن مربوط به رنگ چشم است و آللهای ممکن برای آن عبارتند از: سیاه، خاکستری، قهوهای، آبی، سز و عسلی. درالگوریتم های فرااکتشافی مقادیر مجاز برای مشخصههای هر پاسخ معادل مفهوم آلل است.

4- ژنوتایپ)Genotype(: ترکیب کامل تمامی ژنها برای یک فرد مشخص، ژنوتایپ)ژنوتیپ( نامیده میشود. دنباله ژنی به هر پاسخ در یک الگوریتم فرااکتشافی ژنوتایپ آن پاسخ را نشان میدهد.

5- فنوتایپ(Phenotype): خصوصیات ظاهری یک شخص که از رمزگشایی یک ژنوتایپ حاصل میشود، فنوتایپ)فنوتیپ( نامیده میشود. به عبارت دیگر، تجلی معنایی ژنوتایپ مربوط به یک پاسخ در دنیای واقعی ، گویای فنوتایپ آن پاسخ است.

6- برازش(Reproduction): شایستگی یک موجود در یک جمعیت، برازش آن موجود نامیده میشود. برازش هر پاسخ در جمعیت، با توجه به شایستگی فنوتایپ آن پاسخ تعیین میگردد. در یک الگوریتم فرااکتشافی، با استفاده از یک یا چند تابع، برازش هر پاسخ محاسبه میشود. تابع برازش، تنها ارتباط یک الگوریتم فرااکتشافی با مسئله مورد تحلیل میباشد ]6[ ]۷[.

مولفههای تاثیرگذار بر روی الگوریتمهای فرااکتشافی شامل: 1( طریقه کد کردن راهحل مسئله به عنوان کروموزوم، 2( طراحی تابعی که با آن برازش کروموزومها)پاسخها( بررسی شود، 3( الگوریتم تولید جمعیت اولیه، 4( الگوریتم مربوط به عملگرهای انتخاب، بازترکیب و جهش، 5( نحوه تولید جمعیت در نسلهای بعدی و 6( تعیین شرط توقف می باشند ]8[ ]9[.

## 3. انواع الگوریتم های فرااکتشافی

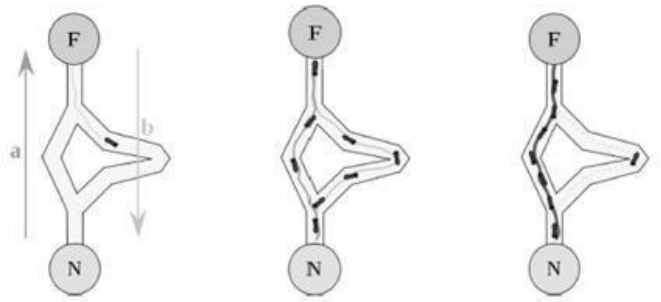
الگوریتم های فرااکتشافی که در این مقاله بررسی می شوند، شامل 1(الگوریتم های کلونی مورچگان، 2(الگوریتم ژنتیک، 3(الگوریتم پرندگان، 4(الگوریتم کلونی زنبور، 5(الگوریتم وال ها )نهنگ ها(، 6(الگوریتم گرگ خاکستری، ۷(الگوریتم کرم شب تاب، 8(الگوریتم توده ذرات، 9(الگوریتم جستجوی بلبل )فاخته(، 10(الگوریتم گرده افشانی گل ها، 11(الگوریتم ملخ ها، 12(اگوریتم

ازدحام گربه ها، 13(الگوریتم جهش قورباغه ها، 14(الگوریتم پرواز پرندگان، 15(الگوریتم کلاغ ها، 16(الگوریتم خفاش ها، 17(الگوریتم علف هرز مهاجم، 18(الگوریتم چرخه آب می باشند.

**3.1. الگوریتم کلونی مورچگان(Ant Colony Optimization)**

پروسه پیدا کردن کوتاهترین مسیر توسط مورچهها، ویژگیهای بسیار جالبی دارد، اول از همه قابلیت تعمیم زیاد و خود- سازمانده بودن آن است. در ضمن هیچ مکانیزم کنترل مرکزی ای وجود ندارد. ویژگی دوم قدرت زیاد آن است. سیستم شامل تعداد زیادی از عواملی است که به تنهایی بیاهمیت هستند بنابراین حتی تلفات یک عامل مهم، تأثیر زیادی روی کارایی سیستم ندارد. سومین ویژگی این است که، پروسه یک فرایند تطبیقی است. از آنجا که رفتار هیچکدام از مورچهها معین نیست و تعدادی از مورچهها همچنان مسیر طولانیتر را انتخاب میکنند، سیستم میتواند خود را با تغییرات محیط منطبق کند و ویژگی آخر اینکه این پروسه قابل توسعه است و میتواند به اندازه دلخواه بزرگ شود. همین ویژگیها الهام بخش طراحی الگوریتمهایی شدهاند که در مسائلی که نیازمند این ویژگیها هستند کاربرد دارند. اولین الگوریتمی که بر این اساس معرفی شد، الگوریتم ABC بود .

الگوریتم بهینه سازی کلونی مورچگان تحت عنوان الگوریتم های هوش ازدحامی )هوش گروهی( شناخته شده و به مدل سازی رفتار مورچه های واقعی می پردازد. مورچه ها حشراتی هستند که می توانند گروه ها )کلونی ها( را شکل دهند. چنین رویکرد جمعیت محوری این امکان را برای الگوریتم ACO ایجاد می کند تا به حل مسائل بهینه سازی پویا به طور کاملا کارآمد بپردازد. مورچه ها به عنوان مخلوقات خودسازمانده می باشند. چنین ویژگی به عنوان هسته مسئله می باشد زیرا این ویژگی دقیقا همان چیزی است که باعث می شود حشرات به سرعت با شرایط متغیر محیطی شان به منظور دستیابی به اهداف از طریق تعامل سطح پایین، وفق یابند. از آنجایی که مورچه ها اصلا چشم ندارند، تعاملات آن ها از طریق ماده شیمیایی فرومون Pheromone که از آن برای نشان گذاری مسیر استفاده می شود، انجام می گیرد. هرچه فرومن های بیشتری در مسیر قرار گیرد مابقی مورچه ها از این مسیر بیشتر استفاده می کنند؛ بنابراین ،چنین کمیتی نشان می دهد که این مسیر به عنوان یکی از بهینه ترین و کوتاه ترین راه می باشد. اکنون نگاهی به یک نمونه عینی در شکل زیر می اندازیم. هدف پیدا کردن بهترین راه از نقطه آغازی N )آشیانه( به نقطه مقصد F )منبع غذا( می باشد.




شکل1 . کلونی مورچه

ممکن است این حدس زده شود که احتمال برای مورچه ای که مسیر درست را می پیماید برابر با همان احتمالی می باشد که مسیر اشتباه را انتخاب کند. نکته در اینجا این است که مورچه ای که کوتاه ترین مسیر را می پیماید، اولین مورچه ای است که به نقطه مقصد رسیده و سپس به آشیانه ) نقطه مبدا حرکت( بر می گردد، بنابراین در این کوتاه ترین مسیر فرمون های بیشتری وجود دارد. از این رو فرمون دقیقا همان چیزی است که نشان می دهد که مورچه باید از چه مسیری برود و در پایان کوتاه ترین راه، بهترین مسیر می باشد .تبخیر شدن فرومون و احتمال-تصادف به مورچهها امکان پیدا کردن کوتاهترین مسیر را میدهد. این دو ویژگی باعث ایجاد انعطاف در حل هرگونه مسئله بهینهسازی میشوند[10] .

1- ارزش دهی مورچه ها )مقداردهی اولیه(: در این مورد، می توانیم از دو رویکرد متفاوت استفاده کنیم : قرار دادن تمام مورچه ها در یک نقطه یا در نقطه های مختلف، اینکه از چه روشی می بایست استفاده شود بستگی به موقعیت خاص آن دارد. در این مرحله، این مقدار دهی اولیه به فرمون ها نیز انجام می شود. ارزش فرمون بابد یک عدد صحیح کوچک باشد. این کار را باید به این منظور انجام داد تا از ماندن و حرکت نکردن مورچه ها در نقطه آغازی جلوگیری کرد.

2- یافتن راه حل: چون الگوریتم ACO تلاش می کند تا به تقلید رفتار مورچه های واقعی با استفاده از فرایند شبیه سازی بپردازد، از یک تابع احتمال برای توصیف مسیر استفاده می گردد.

در ادامه تعدادی از انواع الگوریتم کلونی مورچگان را معرفی میکنیم.

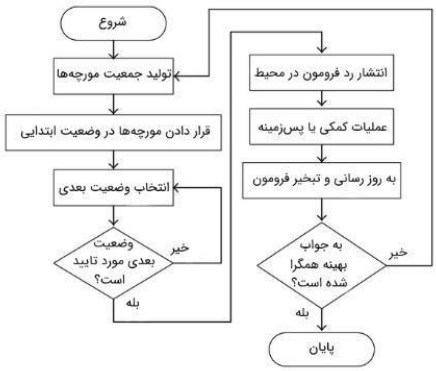
1- سیستم کلونی مورچه Ant System

2- سیستم مورچه نخبگان Elitist Ant System: در این روش بهترین راه حل کلی یافت شده تا کنون در هر تکرار فرمون آزاد میکند. همچنین این روش برای تمام مورچههای مصنوعی باید انجام شود.

3- سیستم مورچه ماکس- مین Min-Max Ant System: یک مقدار کمینه بیشینه برای فرمون تعیین کرده و فقط در هر مرحله بهترین جواب این مقدار را آزاد میکند و تمام گرههای مجاور ان به مقدار فرمون بیشینه مقدار دهی اولیه میشوند.

4- سیستم مورچه بر اساس رتبه Ranked Ant System: تمام راه حلهای بدست آماده بر اساس طول جواب رتبهبندی میشوند و بر اساس همین رتبهبندی مقدار فرمون آزاد سازی شده توسط آنها مشخص خواهد شد و راه حل با طول کمتر از راه حل دیگر با طول بیشتر مقدار فرمون بیشتری آزاد میکند.

5- سیستم مورچه متعامد مداوم Best-Worst Ant System: در این روش مکانیزم تولید فرمون به مورچه اجازه میدهد تا برای رسیدن به جواب بهتر و مشترک با بقیه مورچهها جستجو انجام دهد با استفاده از روش طراحی متعامد مورچه میتواند در دامنه تعریف شده خود به صورت مداوم برای بدست آوردن بهترین جواب جستجو کند که این عمل به هدف رسیدن به جواب بهینه و صحیح ما را نزدیک میکند. روش طراحی متعامد میتواند به دیگر روشهای جستجو دیگر گسترش پیدا کنند تا به مزیتهای این روشهای جستجو اضافه کند.




شکل 2. فلوچارت الگوریتم کلونی مورچه

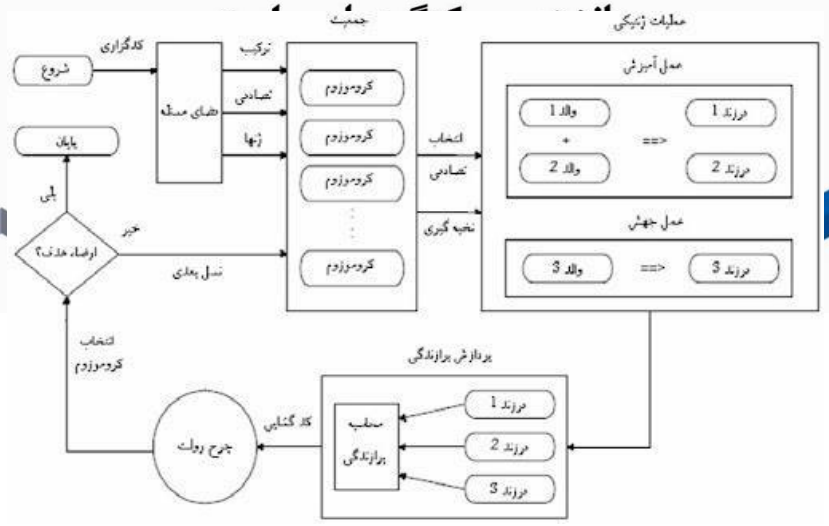
### 3.2. الگوریتم ژنتیک(Genetic algorithm)

جان هلند، دانشمند علوم کامپیوتر و روانشناس، مبدع شاخه ای از علوم کامپیوتربه نام «سیستمهای تطبیقی پیچیده» میباشد. او در کتاب خود یک سیستم تطبیقی یا وفقی را چنین شرح میدهد که سیستم مربوطه به طور یکنواخت و پیوسته خودش را تغییر میدهد تا از محیط اطراف خود بهتر استفاده کند. هُلند در خلال توسعه تئوری خود برای سیستمهای تطبیقی به شرح اپراتورهای ژنتیک برای تغییر حالت سیستم میپردازد. گر چه کتاب هُلند، اختصاصاً برای سیستمهای تطبیقی نگاشته شدهاست ولی یک مشخصه بسیار مهم دارد و آن ابداع و معرفی الگوریتم ژنتیک میباشد.

هر کروموزوم شامل بیت های اطلاعاتی و تکههای رقومی هستند. هر کدام از این تکهها موسوم به ژن میباشند. در یک بیان تناظری، نسبت مادی ژن به کروموزوم به فرد مثل نسبت معنوی حرف به کلمه به جمله میباشد. هر ژن میتواند مقادیری را اختیار کند و هر مقدار ممکن موسوم به یک اَلل میباشد. در صورتی که ژن کد باینری باشد، آنگاه دو اَلل بیشتر نداریم، یکی مقدار صفر) 0( و دیگری مقدار) 1( و موقعیت یک ژن در کروموزوم مربوطه معروف به مکان یا موقعیت میباشد ]11[.

الگوریتم ژنتیک روشی برای بهینه سازی با جستجوی وسیع است و کارکرد آن بر اصول انتخاب طبیعی حاکم بر ژنتیک طبیعی استوار است. در این الگوریتم اطلاعات تاریخی از چگونگی تکامل، به شکلی کارا استخراج شده و در روند جستجو استفاده می شود. الگوریتم ژنتیک روشی قدرتمند بوده و بر روی دسته وسیعی از مسائل بهخوبی عمل می کند. الگوریتم های ژنتیک که بر اساس ایده ی تکامل بیولوژیکی در طبیعت عمل می نمایند، بر روی جمعیتی از راه حل های بالقوه یا کروموزوم ها که هر یک می توانند بهعنوان پاسخی از مسئله تلقی شوند، با اعمال عملگرهای ژنتیک به جستجوی راه حل نهایی می پردازند. در الگوریتم های ژنتیکی، بسیاری از مکانیزم هایی که در زیست شناسی وجود دارد، نظیر انتخاب ژن برتر، ترکیب ژن ها، جهش ژن ها، مهاجرت افراد جمعیت، محلی بودن گونه ها و ... شبیه سازی می شوند. در این الگوریتم ها، جستجو بر روی مجموعه هایی از راه حل ها بهصورت موازی انجام می شود، درحالیکه در روش های سنتی جستجو بهصورت ترتیبی است ]12[ ]13[.

senaconf.ir

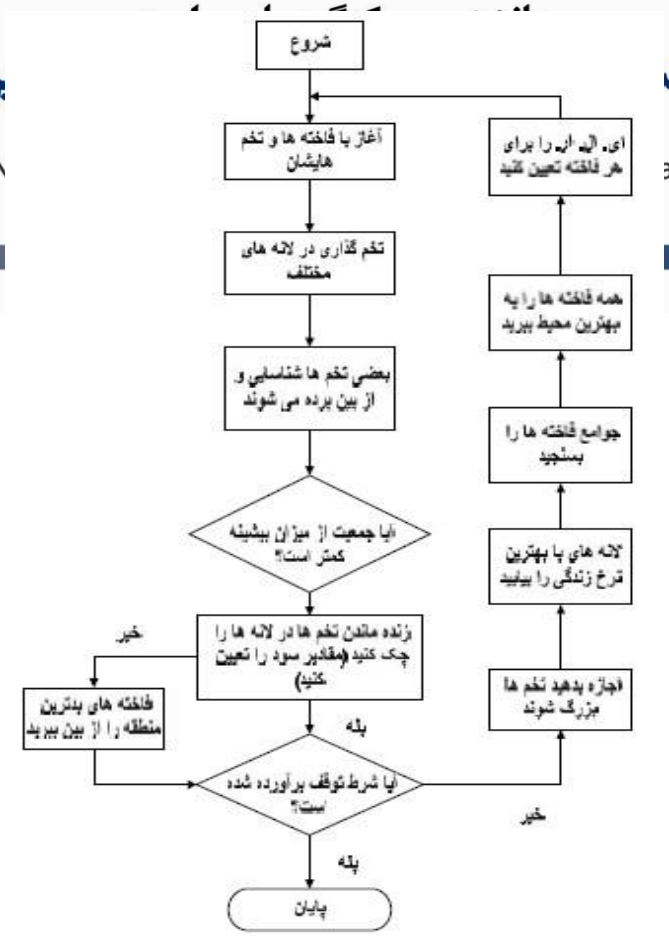



شکل 3. فلوچارت الگوریتم ژنتیک

روند کار الگوریتم ژنتیک: )در آغاز الگوریتم، تعدادی از افراد بهعنوان جمعیت اولیه و معمولاً بهصورت تصادفی ساختهشده و معیاری از کیفیت به نام تابع هدف یا برازندگی برای تک تک آن ها ارزیابی می شود. اگر شرط رسیدن به جواب برقرار نباشد )به جواب بهینه نرسیده باشیم(، نسل بعدی با انتخاب والدین بر اساس میزان برازندگی آن ها تولید می شود.در هر نسل، بهترین های آن نسل انتخاب می شوند و پس از زاد و ولد، مجموعه جدیدی از فرزندان را تولید می کنند. کروموزوم های موجود در جمعیت بر اساس مقدار برازندگی بهعنوان والد انتخاب می شوند. سپس تولیدمثل، بین جفت کروموزوم ها انجام می گیرد تا فرزندان ایجاد شوند و فرزندان با احتمالی ثابت دچار جهش می شوند. سپس میزان برازندگی فرزندان جدید محاسبهشده و جمعیت جدید، از جایگزینی فرزندان با والدین ایجاد می شود و جمعیت ایجادشده جدید بهعنوان نسل بعدی شناخته میشود و فرایند تکرار می شود.در این فرایند، افراد مناسب تر با احتمال بیشتری در نسل های بعد باقی خواهند ماند و این فرایند تا برقرار شدن شرط خاتمه تکرار می شود. الگوریتم زمانی پایان خواهد یافت که بهبودی بر روی جواب ها صورت نگیرد و یا اینکه تعداد مشخصی نسل تولید شود ]14[.

### 3.3. الگوریتم پرندگان) Particle swarm optimization(

الگوریتم PSO بهینه سازی ازدحام ذرات (PSO) یک روش بهینه سازی تصادفی مبتنی بر جمعیت است که توسط دکتر کندی )Kennedy( و دکتر ابراهارت )Eberhart( در سال 1995 طراحی شده است که الهام گرفته از رفتار اجتماعی ازدحام پرندگان و پرورش ماهی است. الگوریتم PSO شباهت زیادی با تکنیک های محاسباتی فرااکتشافی مانند الگوریتم ژنتیک (GA) دارد. این سیستم با جمع آوری راه حل های تصادفی و جستجو برای بهینه سازی با به روز رسانی نسل ها آغاز می شود. با این حال، بر خلاف GA الگوریتم PSO هیچ اپراتور فرااکتشافی مانند Cross Over و جهش ندارد. در PSO، راه حل های بالقوه، ذرات نامیده می شوند که از طریق فضای مشکل با دنبال کردن ذرات بهینه مطلوب پرواز می کنند. اطلاعات دقیق در بخش های زیر داده می شود. در مقایسه با الگوریتم ژنتیک، الگوریتم PSO آسان است و چند پارامتر برای تنظیم آن وجود دارد. این الگوریتم در بسیاری از زمینه ها به طور موفقیت آمیزی مورد استفاده قرار گرفته است: بهینه سازی عملکرد، آموزش شبکه های عصبی مصنوعی، کنترل سیستم فازی و سایر زمینه ها می تواند استفاده شود.




شکل 4 . فلوچارت الگوریتم پرندگان

الگوریتم PSO رفتارهای ازدحام پرنده را شبیه سازی می کند. تصور کنید سناریوی زیر: گروهی از پرندگان به طور تصادفی در یک منطقه در معرض غذا قرار می گیرند. در منطقه مورد جستجو تنها یک قطعه غذا وجود دارد. همه پرندگان نمی دانند کجا غذا است. اما آنها می دانند که چقدر مواد غذایی در هر تکرار است. بنابراین بهترین استراتژی برای یافتن غذا چیست؟ راه حل این است که دنبال پرنده ای که نزدیکتر به غذا است را دنبال کنید. طراحان PSO از این سناریو استقبال کردند و از آن برای حل مشکلات بهینه سازی استفاده کردند. در PSO، هر یک از راه حل یک "پرنده" در فضای جستجو است. ما آن را "ذره" می نامیم. تمام ذرات دارای مقادیر تناسب هستند که توسط تابع تناسب برای بهینه سازی ارزیابی می شوند و دارای سرعت هایی هستند که پرواز ذرات را هدایت می کنند. ذرات از طریق فضای مشکل با ذرات بهینه مطلوب جریان می یابند.

نحوه عملکرد PSO: الگوریتم PSO با گروهی از ذرات تصادفی )راه حل( آغاز می شود و سپس با به روز رسانی نسل ها جستجو می کند .در هر تکرار، هر ذره با دو ارزش "بهترین" به روزرسانی می شود. اولین مورد بهترین راه حل ) تابع تناسب ( است که تاکنون به دست آورده است. )ارزش تناسب نیز ذخیره می شود.( این ارزش pbest نامیده می شود. یکی دیگر از "بهترین ها" ارزشی است که تا کنون توسط هر ذره در جمعیت به دست آمده است. این بهترین ارزش جهانی بهترین است و به نام gbest است. هنگامی که یک ذره بخشی از جمعیت را به عنوان همسایگان توپولوژیکی خود می گیرد، بهترین ارزش بهترین محلی است و به نام lbest نامیده می شود.

### 3.4. الگوریتم کلونی زنبور)Artificial bee colony algorithm(

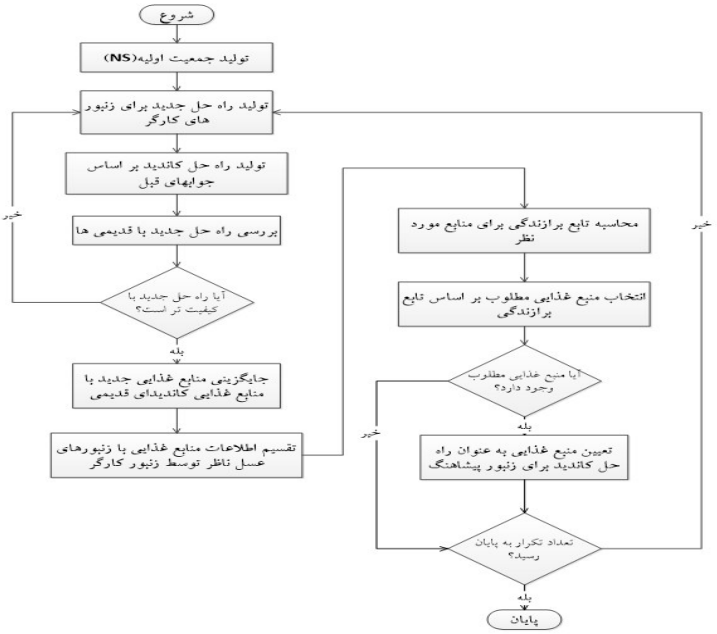
یک الگوریتم بهینه سازی بر اساس هوش جمعی و رفتار هوشمندانه جمعیت زنبور عسل است. الگوریتم کلونی زنبور عسل مصنوعی یا به اختصار ABC، یک راهکار بهینهسازی است که رفتار یک کلونی زنبور عسل را شبیهسازی میکند و برای اولین بار در سال 2005 توسط Dervis Karaboga، برای بهینهسازی ارائه شد.

در این مدل، کلونی زنبور عسل مصنوعی دارای سه نوع زنبورهای کارگر، ناظر و دیده بان است. زنبورهای کارگر روی گردآوری غذا و آوردن آن به کندو از یک منبع غذایی خاص کار میکنند. زنبورهای ناظر در میان کارگرها گشت میزنند تا تشخیص دهند یک منبع غذایی همچنان ارزش استفاده دارد یا خیر و در نهایت زنبورهای دیدهبان که بهدنبال کشف منابع غذایی جدید هستند. در الگوریتم

زنبور عسل، یک منبع غذایی به عنوان حالتی از جستجو برای منابع غذایی در ابتدا برابر با تعداد زنبورهای موجود در کندو است. کیفیت منابع غذایی توسط مقدار تابع هدف در آن موقعیت )مقدار تناسب( تعیین میشود. یک کلونی زنبور عسل میتواند در مسافت زیادی و نیز در جهتهای گوناگون پخش شود تا از منابع غذایی بهره برداری کند. گل ها با مقادی زیادی شهد و گرده که با تلاشی کم قابل جمع آوری است، به وسیله ی تعداد زیادی زنبور بازدید میشود؛ به طوری که قطعاتی از زمین که گرده یا شهد کمتری دارد، تعداد کمتری زنبور را جلب میکند.

جستجو غذا: پروسه جستجوی غذای یک کلونی به وسیله زنبورهای دیده بان آغاز میشود که برای جستجوی گل زار ها فرستاده میشوند. زنبورهای دیدهبان از گلزاری به گلزار دیگر حرکت میکنند. در طول فصل برداشت محصول )گلدهی(، کلونی با آماده نگه داشتن تعدادی از جمعیت کلونی به عنوان زنبور دیدهبان به جستجوی خود ادامه میدهند. هنگامی که جستجوی تمام گلزار ها پایان یافت، هر زنبور دیدهبان، بالای گلزاری که اندوخته کیفی مطمئنی از شهد و گرده دارد، رقص خاصی را اجرا میکند. این رقص که به نام رقص چرخشی شناخته میشود، اطلاعات مربوط به جهت تکه گلزار )نسبت به کندو(، فاصله تا گلزار و کیفیت گلزار را به زنبورهای دیگر انتقال میدهد. این اطلاعات زنبورهای اضافی و پیرو را به سوی گلزار میفرستد. بیشتر زنبورهای پیرو به سوی گلزارهایی میروند که امید بخشتر هستند و امید بیشتری برای یافتن نکتار و گرده در آنها وجود دارد. وقتی همه ً زنبورها به سمت ناحیهای مشابه بروند، دوباره به صورت تصادفی و به علت محدودهی رقصشان در پیرامون گلزار پراکنده میشوند تا به موجب این کار سرانجام نه یک گلزار، بلکه بهترین گلهای موجود درون آن تعیین موقعیت شوند.

الگوریتم زنبور عسل هر نقطه را در فضای پارامتری متشکل از پاسخهای ممکن به عنوان منبع غذا تحت بررسی قرار میدهد. زنبورهای دیدهبان به صورت تصادفی فضای پاسخ ها را ساده می کنند و به وسیله ی تابع شایستگی کیفیت موقعیت های بازدید شده را گزارش می دهند. جوابهای ساده شده رتبه بندی میشوند و دیگر زنبورها نیروهای تازهای هستند که فضای پاسخها را در پیرامون خود برای یافتن بالاترین رتبه محلها جستجو میکنند که گلزار نامیده میشود. الگوریتم به صورت گزینشی دیگر گلزارها را برای یافتن نقطه ی بیشینه ی تابع شایستگی جستجو میکند.




شکل 5. فلوچارت الگوریتم کلونی

زنبور **5.3. الگوریتم وال ها/نهنگ ها) The Whale Optimization Algorithm(**

الگوریتم وال هیکی از بزرگترین پستانداران دنیا وال یا نهنگ است از بین ۷ نهنگ دنیا معروف ترین آنها، نهنگ گوژپشت یا نهنگ کوهان دار است. یک وال گوژپشت بالغ تقریباً به اندازه یک اتوبوس مدرسه است. شکار مورد علاقه وال ها، کریل و گروه های ماهی کوچک است .

جالب ترین نکته در مورد وال های گوژپشت روش خاص شکار آنها می باشد. این رفتار کاوش گرانه به نام روش تغذیه حباب تور (Bubble-net)شناخته می شود ]15[.

وال های گوژپشت ترجیح می دهند تا دسته ای از شکار کوچک را در نزدیکی سطح آب شکار کنند. این نکته مشاهده شده است که این کاوش و شناسایی حباب غذا شاخصی در امتداد یک دایره یا مسیر مارپیچ به انجام می رسد. الگوریتم WOA یکی از الگوریتم های بهینه سازی الهام گرفته شده از طبیعت و مبتنی بر جمعیت است که در زمینه های مختلف می توان از آن استفاده کرد.

مراحل الگوریتم

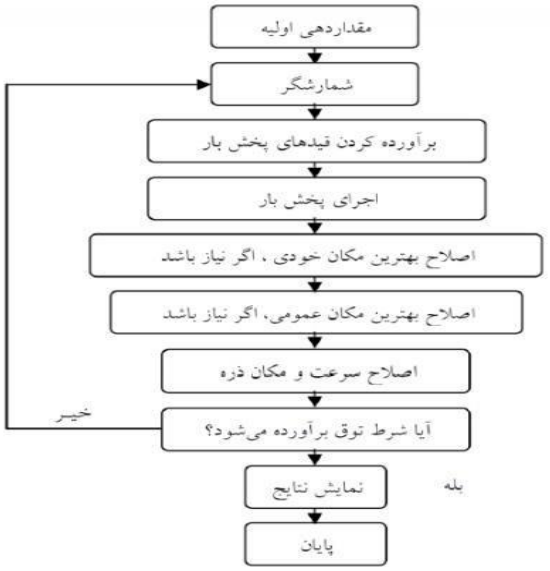
الگوریتم وال WOA در سه مرحله یا سه فاز انجام می شود که بصورت زیر است:

شکار محاصره ای

فاز بهره برداری : روش حمله به حباب

تور مرحله اکتشاف: جستجوی شکار

شکار محاصره ای: وال ها می توانند مکان شکار را شناسایی کرده و آنها را محاصره کنند. از آنجایی که مکان طراحی بهینه در فضای جستجو از راه مقایسه شناخته نمی شود، الگوریتم فرض می کند که بهترین راه حل کاندید حال حاضر، شکار هدف بوده و یا نزدیک به حالت مطلوب است. بعد از اینکه بهترین عامل جستجو شناسایی شد، عوامل دیگر جستجو سعی می کنند تا مکان خود را نسبت به بهترین عامل جستجو، به روزرسانی کنند.




شکل 6. فلوچارت الگوریتم وال ها )نهنگ ها(

### 3.6. الگوریتم گرگ خاکستری)Grey Wolf Optimizer(

الگوریتم گرگ خاکستری GWO یک الگوریتم متاهیورستیک (Meta Heuristic)است که از ساختار سلسله مراتبی hieratical و رفتار اجتماعی گرگ های خاکستری در هنگام شکار کردن الهام گرفته است. این الگوریتم مبتنی بر جمعیت بوده، فرآیند ساده ای دارد و به سادگی قابلیت تعمیم به مسائل با ابعاد بزرگ را دارد. گرگ های خاکستری به عنوان شکارچیان راس یا apex در نظر گرفته می شوند ،که در بالای هرم زنجیره غذایی هستند. گرگهای خاکستری ترجیح می دهند در یک گروه )دسته( زندگی کنند، هر گروه به طور متوسط 12-5 عضو دارد. همه اعضای این گروه دارای سلسله مراتب تسلط اجتماعی بسیار دقیق هستند و وظایف خاصی دارند. در هر گله از گرگ ها برای شکار کردن 4 درجه وجود دارد که مانند شکل زیر به صورت یک ساختار هرمی مدل می شود.

گرگ های رهبر گروه alpha نامیده می شوند که می توانند مذکر یا مونث باشند. این گرگ ها بر گله تسلط دارند.

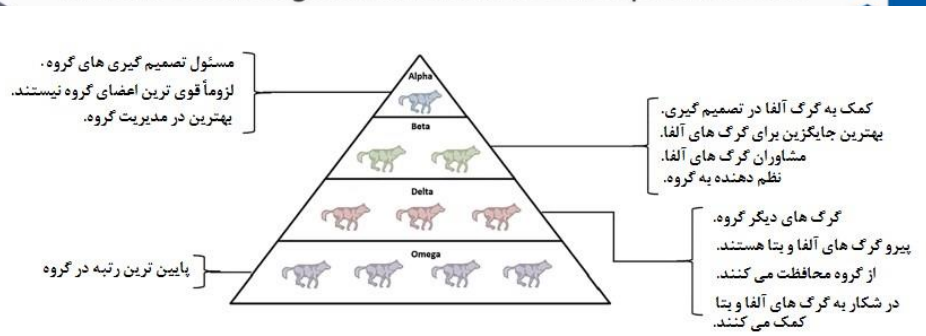
گرگ های beta: کمک به گرگ های alpha در فرایند تصمیم گیری بوده و همچنین مستعد انتخاب شدن به جای آن ها هستند.

گرگ های delta: پایین تر از گرگ های beta و شامل گرگ های پیر، شکارچی ها و گرگ های مراقبت کننده از نوزادان

گرگ های omega: پایین ترین مرتبه در هرم سلسله مراتب که کمترین حق را نسبت به بقیه اعضای گروه دارند. بعد از همه غذا می خورند و در فرایند تصمیم گیری مشارکتی ندارند.

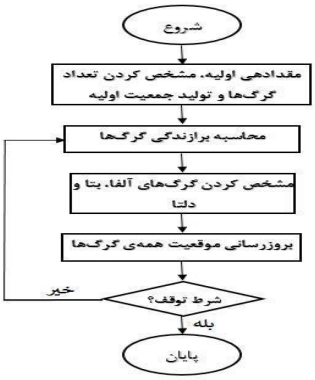
روش شکار در گرگ های خاکستری : در توضیح و آموزش الگوریتم گرگ خاکستری می توان گفت این الگوریتم شامل 3 مرحله اصلی است: مشاهده شکار، ردیابی و تعقیب آن (tracking and approaching) نزدیک شدن، احاطه کردن )حلقه زدن( به دور شکار و گمراه کردن آن تا زمانی که از حرکت باز بماند (Pursing and encircling) حمله به شکار (attacking) بهینه سازی با استفاده از گرگ های alpha, beta و delta انجام می شود. یک گرگ به عنوان alpha هدایت کننده اصلی الگوریتم فرض می شود

و یک گرگ beta و delta نیز مشارکت دارند و بقیه گرگ ها به عنوان omega محسوب می شوند. گرگ های خاکستری توانایی تخمین موقعیت شکار را دارند برای مدل سازی این فرایند فرض را بر این گذاشتیم که هیچ ایده ای در مورد موقعیت شکار نداریم .فرض می شود گرگ های alpha, beta, delta دانش اول بهتری در خصوص موقعیت شکار )نقطه بهینه جواب( دارند.وقتی شکار توسط گرگ ها احاطه شده است حرکت بعدی سایر گرگ ها alpha باید می شود ]16[.




شکل ۷ . گرگ خاکستری

ترتیب الگوریتم: برازندگی کلیه جواب ها محاسبه شده و سه جواب برتر به عنوان alpha, beta, deltaتا پایان الگوریتم انتخاب می شوند. در هر تکرار سه جواب برتر )گرگ های (alpha, beta, deltaقابلیت تخمین موقعیت شکار را داشته و این کار را در هر iteration با استفاده از رابطه زیر انجام می دهند: در هر تکرار بعد از تعیین موقعیت گرگ های alpha, beta, delta، آپدیت موقعیت بقیه جواب ها با تبعیت از آن ها انجام می شود. در هر تکرار بردار A و به تبع آن (AوC( آپدیت می شوند.در پایان تکرارها موقعیت گرگ alpha به عنوان نقطه بهینه معرفی می شود.



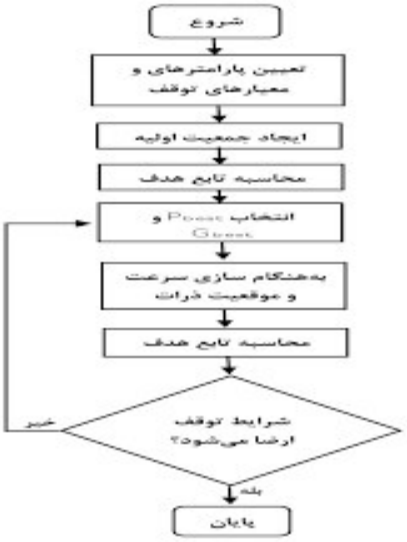

شکل 8. فلوچارت الگوریتم گرگ خاکستری

### 3.7. الگوریتم کرم شب تاب (Firefly Algorithm)

الگوریتم بهینه سازی کرم شبتاب، از رفتار کرم های شبتاب طبیعی که در مجموعه ها بزرگ در کنار هم زندگی می کنند الهام گرفته شده است و یکی از الگوریتم های بسیار کارآمد در حل مسائل بهینه سازی ترکیبی است.الگوریتم های دیگری نیز بر اساس الگوریتم کرم شب تاب ساخته شده اند که همگی سیستم های چند عاملی Multi Agent هستند و عامل ها کرم های شبتاب مصنوعی هستند که مشابه با کرم های شبتاب واقعی رفتار می کنند. الگوریتم کرم شبتاب، یک مثال بارز از هوش جمعی هستند که در آن عامل هایی که قابلیت چندان بالایی ندارند، در کنار هم و با همکاری یکدیگر می توانند نتایج بسیار خوبی به دست بیاورند. از این دسته الگوریتم ها می توان به الگوریتم های کلونی مورچگان، زنبور عسل، پرندگان و ... اشاره کرد.(کرم شب تاب از خانواده حشرات و زیرمجموعه سوسکها است. کرمهای شب تاب نوری از خود تولید میکنند که فاقد طیفهای فرابنفش میباشد. نور کرم شبتاب کاملاً شبیه سایر نورهاست. به استثنای آن که این نور حرارتی ندارد. این قبیل نور را به نام )لومی نسانس( میشناسند. در کرم شبتاب این نور توسط مادهای به نام )لوسی فرین( تولید میشود که این ماده با اکسیژن ترکیب شده و تولید نور مینماید این نور دارای طول موج 510 تا 670 نانومتر متغییر است و میتواند به رنگهای زرد، سبز یا قرمز کمرنگ دیده شود.) دانشمندان قبلاً بر این عقیده بودند که این حشرات با نورشان موجب جلب توجه جنس مخالف برای جفت گیری و یا صید برای شکار میشوند ]17[ ]18[.

کرمهای شب تاب از نوردهی به عنوان یک سیستم دفاعی برای مقابله با شکارچیان استفاده میکنند. تاکنون بیش از 2000 گونه از این نوع حشرات در نواحی معتدل و گرمسیری شناسایی شده است. بسیاری از آن ها در مناطق مردابی و جنگلی، زندگی میکنند. در بین بیشتر گونهها هر دو جنس نر و ماده توانایی پرواز دارند. اما در بین بعضی گونهها، جنس ماده قادر به پرواز نمیباشد ]19[ ]20[.

این الگوریتم برای حل و بررسی محدوده وسیعی از مسائل بهینه سازی ترکیبی به کار رفته است. از این میان می توان به حل مسأله کلاسیک فروشنده دوره گرد، همچون مسأله راهیابی در شبکه های مخابرات راه دور، تابه مسأله (مسأله فروشنده دوره گرد (Traveling Salesman Problem)و یا به اختصار TSP، یکی از مسائل مشهور بهینه سازی ترکیبی است. در این مسأله، یک فروشنده دوره گرد می خواهد به چند شهر سفر کند و کالای خود را به فروش برساند و می‌بایست تمام شهرها عبور کند، از هر شهر فقط یک بار عبور کند و با طی کوتاه ترین مسیر، سفر خود را به پایان برساند ]21[ ]22[.(




شکل 9. فلوچارت الگوریتم شب

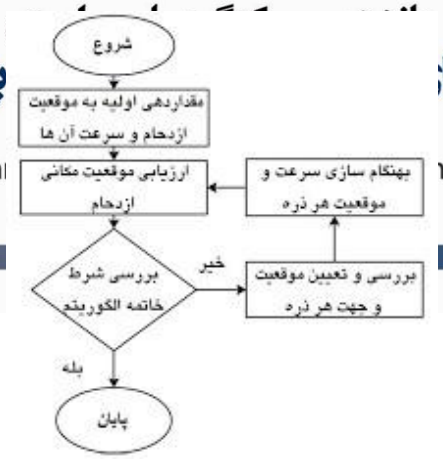
تاب این الگوریتم با فرضیه زیر فرمول بندی شده است:

همه کرم شب تاب ها تمایل جنسی دارند، به طوری که یک کرم شب تاب به تمام کرم شب تاب های دیگر را جذب می کند.جذابیت متناسب است به روشنایی خود، و برای هر دو کرم شب تاب یکی کمتر روشن خواهد شد جذب )و در نتیجه به حرکت می افتد ( یکی روشن تر، با این حال، روشنایی می تواند به عنوان فاصله آنها افزایش و یا کاهش یابد . اگر کرم شب تابی روشن تر از کرم شب تاب داده شده وجود داشته باشد آن را به طور تصادفی حرکت خواهد داد.روشنایی باید با تابع هدف در ارتباط باشد.

تشریح الگوریتم: الگوریتم FA با مدلسازی رفتار مجموعه ای از کرم های شب تاب و تخصیص مقداری مرتبط با برازندگی مکان هر کرم شب تاب به عنوان مدلی برای میزان رنگدانه های شب تاب و به روز کردن مکان کرم ها در تکرار های متوالی الگوریتم به جستجوی جواب بهینه مسئله می پردازد. در واقع دو مرحله اصلی الگوریتم در هر تکرار فاز به روز کردن رنگدانه و فاز حرکت هستند. کرم های شب تاب به سمت کرم های شب تاب دیگر با رنگدانه بیشتر که در همسایگی آنها باشند حرکت می کنند. به این ترتیب طی تکرار های متوالی مجموعه به سمت جواب بهتر متمایل می گردد.

### 3.8. الگوریتم توده)ازدحام( ذرات) Particle swarm optimization(

در سال 1995 ابرهارت و کندی برای اولین بار )Particle swarm optimization( یا به اختصار PSO به عنوان یک روش جستجوی غیر قطعی برای بهینهسازی تابعی مطرح گشت این الگوریتم از حرکت دسته جمعی پرندگانی که به دنبال غذا میباشند، الهام گرفته شدهاست. گروهی از پرندگان در فضایی به صورت تصادفی دنبال غذا میگردند. تنها یک تکه غذا در فضای مورد جستجو وجود دارد. هر راه حل که به آن یک ذره گفته میشود، PSO در الگوریتم معادل یک پرنده در الگوریتم حرکت جمعی پرندگان میباشد. هر ذره یک مقدار شایستگی دارد که توسط یک تابع شایستگی محاسبه میشود. هر چه ذره در فضای جستجو به هدف )غذا در مدل حرکت پرندگان( نزدیکتر باشد، شایستگی بیشتری دارد. همچنین هر ذره دارای یک سرعت است که هدایت حرکت ذره را بر عهده دارد. هر ذره با دنبال کردن ذرات بهینه در حالت فعلی، به حرکت خود در فضای مسئله ادامه میدهد.یک الگوریتم جستجوی جمعی است که از روی رفتار اجتماعی دستههای پرندگان مدل شدهاست. در ابتدا این الگوریتم به منظور کشف الگوهای حاکم بر پرواز همزمان پرندگان و تغییر ناگهانی مسیر آنها و تغییر شکل بهینه ً دسته به کار گرفته شد. درPSO ، ذرات در فضای جستجو جاری میشوند. تغییر مکان ذرات در فضای جستجو تحت تأثیر تجربه و دانش خودشان و همسایگانشان است؛ بنابراین موقعیت دیگر توده ذرات روی چگونگی جستجوی یک ذره اثر میگذارد. نتیجه ً مدلسازی این رفتار اجتماعی فرایند جستجویی است که ذرات به سمت نواحی موفق میل میکنند ]23[.



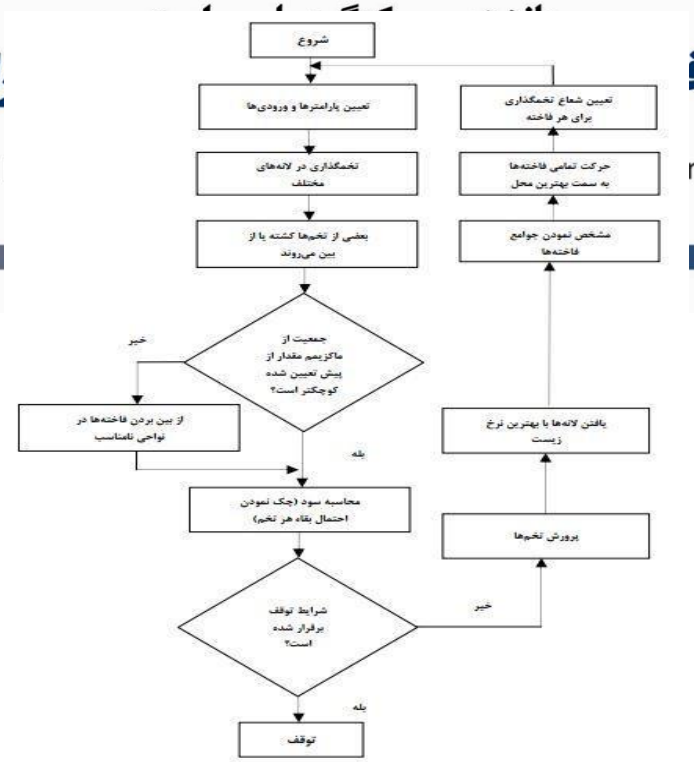

شکل 10. فلوچارت الگوریتم ازدحام ذرات

ذرات از یکدیگر میآموزند و بر مبنای دانش بدست آمده به سمت بهترین همسایگان خود میروند اساس کار PSO بر این اصل استوار است که در هر لحظه هر ذره مکان خود را در فضای جستجو با توجه به بهترین مکانی که تاکنون در آن قرار گرفتهاست و بهترین مکانی که در کل همسایگیاش وجود دارد ،تنظیم میکند. یک روش سراسری بهینهسازی است که با استفاده از آن میتوان با مسائلی که جواب آنها یک نقطه یا سطح در فضای n بعدی میباشد، برخورد نمود. در این چنین فضایی، فرضیاتی مطرح میشود و یک سرعت ابتدایی به آنها اختصاص داده میشود، همچنین کانالهای ارتباطی بین ذرات در نظر گرفته میشود. سپس این ذرات در فضای پاسخ حرکت میکنند، و نتایج حاصله بر مبنای یک «ملاک شایستگی» پس از هر بازه ٔ زمانی محاسبه میشود. با گذشت زمان، ذرات به سمت ذراتی که دارای ملاک شایستگی بالاتری هستند و در گروه ارتباطی یکسانی قرار دارند، شتاب میگیرند. علیرغم اینکه هر روش در محدودهای از مسائل به خوبی کار میکند، این روش در حل مسائل بهینهسازی پیوسته موفقیت بسیاری از خود نشان دادهاست ]23[.

### 9.3. الگوریتم بلبل ها/فاخته) Cuckoo search(

جستجوی کوکو یا جستجوی بلبل، الگوریتم بهینهسازی است که زین–شی یانگ و سوآش دب در سال 2009 طراحی کردند. این الگوریتم برگرفته از ملزوم کردن تخم انگلی بعضی گونههای بلبل به قرار دادن تخمهایش در آشیانه پرندگان میزبان دیگر )از گونههای دیگر( است. بعضی پرندگان میزبان میتوانند با فاختههای سربار و مزاحم جنگ و دعوا کنند. برای مثال اگر پرنده میزبان تخمهایی را پیدا کند که متعلق به آنها نیست، او این تخمهای بیگانه را دور میاندازد یا آشیانه اش را به راحتی ترک میکند و جای دیگر آشیانه جدیدی میسازد.

هر تخم در آشیانه نمایانگر یک راه حل است و تخم بلبل یک راه حل جدید را نشان میدهد. هدف آن است تا از راه حلهای بهتر و جدید بالقوه )بلبلها( استفاده شود تا راه حل نه خیلی ایدهآلی در اشیانهها جایگزین شود . براساس سد قانونی تأیید شدهاست ،هر بلبل یکی یکی بر روی یک تخم میخوابد و تخم خود را در آشیانه تصادفی انتخاب شده میاندازد .بهترین اشیانهها با بهترین کیفیت تخمها به تولید بعدی واگذار میشود ]24[ ]25[.



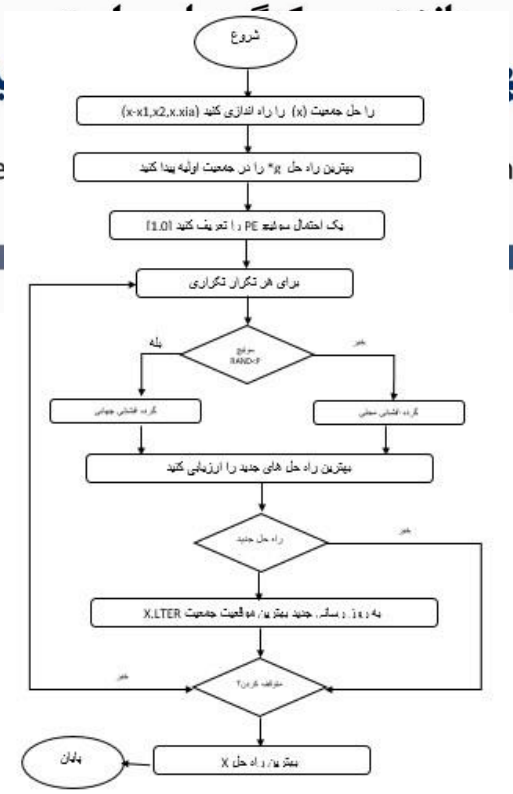

شکل 11. فلوچارت الگوریتم بلبل ها/فاخته

اگر چه جستجوی بلبل یک الگوریتم بر پایه انبوه هوش است، اما هنوز میتواند با الگوریتمهای دیگر بر پایه انبوه همچون psoترکیب شود. برای مثال به نظر میرسد الگوریتم ترکیبی cs-pso نقص pso را اصلاح میکند. موارد استفاده: موارد استفاده جستجوی بلبل در مسایل بهینهسازی مهندسی در بازده پیشبینی شده آن نشان داده شدهاست. الگوریتم جستجوی بلبل گسسته پیشبینی شده به تازگی طراحی میشود تا مسئله زمانبندی پرستاری را حل کند. روش محاسباتی مؤثر بر پایه جستجوی بلبل که برای انتشار اطلاعات در شبکههای حسگیری سیم فرض شدهاست. جستجوی بلبل بر پایه مقدار معین انجام شد تا مسائل کی تاپ سک (knapsack) را حل کند که تأثیر آن را نشان دهد. از جستجوی بلبل همچنین میتوان استفاده کرد تا مسیرهای آزمایشی مستقل را برای آزمایش نرمافزار ساختاری و برای ایجاد اطلاعات آزمایشی به وجود آورد ]26[.

مسئله کولهپشتی که با نامهای Knapsackیا Rucksackمطرح میشود مسئلهای در بهینهسازی ترکیبیاتی است. فرض کنید مجموعهای از اشیا که هر کدام داری وزن و ارزش خاصی هستند در اختیار دارید. به هر شی تعدادی را تخصیص دهید بهطوریکه وزن اشیا انتخاب شده کوچکتر یا مساوی حدی از پیش تعیین شده، و ارزش آنها بیشینه شود. علت نامگذاری این مسئله، جهانگردی است که کولهپشتیای با اندازه ٔ محدود دارد و باید آن را با مفیدترین صورت ممکن از اشیا پر کند.معمولاً در تخصیص منابع با محدودیتهای مالی، با این مسئله روبرو هستیم. همچنین مسائلی از این قبیل در ترکیبیات، نظریه پیچیدگی محاسباتی، رمزنگاری و ریاضیات کاربردی به چشم میخورد ]2۷[ ]28[.

### 10.3. الگوریتم گرده افشانی گل ها)Flower Pollination(

اولین بار توسط فردی به نام Xin-She Yang و در سال 2012 منتشر شد. برای اولین بار ازالگوریتم جدید برای اعتبار سنجی استفاده می کردند. عملکرد آن را با الگوریتم های ژنتیکی و بهینه سازی ذرات مقایسه می کنند. نتایج شبیه سازی ما نشان می دهد که الگوریتم گرده افشانی گلها نسبت به GA و PSO کارایی بیشتری دارد. همچنین از الگوریتم گرده افشانی گلها برای حل یک معیار طراحی غیر خطی استفاده کرد که نشان می دهد نرخ همگرایی تقریبا نمایی است ]29[.




شکل 12. فلوچارت الگوریتم گرده افشانی گل ها

هدف اصلی یک گل در نهایت تولید مثل توسط گرده افشانی است. گرده گل به طور معمول با انتقال گرده ارتباط دارد و چنین انتقال اغلب با گرده افشان هایی مانند حشرات، پرندگان، خفاش ها و دیگر حیوانات مرتبط است. در حقیقت، برخی گلها و حشرات به یک مشارکت بسیار حرفه ای گرده افشانی را انجام می دهند.

به عنوان مثال، برخی از گل ها فقط می توانند به یک گونه خاص حشرات برای گرده افشانی موفقیت آمیز جذب شوند.

گرده افشانی می تواند دو شکل عمده: abiotic و .biotic

حدود 90 درصد از گیاهان گلدار متعلق به گرده افشانی زیستی هستند. یعنی گرده توسط یک گرده افشانی مانند حشرات و حیوانات منتقل می شود. حدود 10 درصد گرده افشانی طول می کشد که فرم گرده افشانی را به وجود می آورد که به گرده افشان نیاز ندارد.باد و انتشار در آب گرده افشانی از گیاهان گلدار و چمن به عنوان مثال خوب است. تخمین زده می شود که حداقل 000.200 انواع حشرات گرده افشان مانند حشرات، خفاش ها و پرندگان وجود دارد. زنبور عسل نمونه خوبی از حشرات گرده افشان است. گرده افشانی می تواند از گرده گل یک گیاه متفاوت اتفاق بیفتد.

### 3.11. الگوریتم ملخ ها(Grasshopper Optimization)

در این الگوریتم به روز رسانی موقعیت هر ملخ به فاصله هر ملخ از تمام جمعیت ملخ ها در نسل جاری و موقعیت بهترین ملخ وابسته است. از ویژگی های این الگوریتم می توان به سادگی و دارا بودن فقط یک پارامتر تنظیم، اشاره کرد. نتایج حاصل از توابع محک گوناگون حاکی از عملکرد مطلوب الگوریتم در مقایسه با الگوریتم های مشابه است.

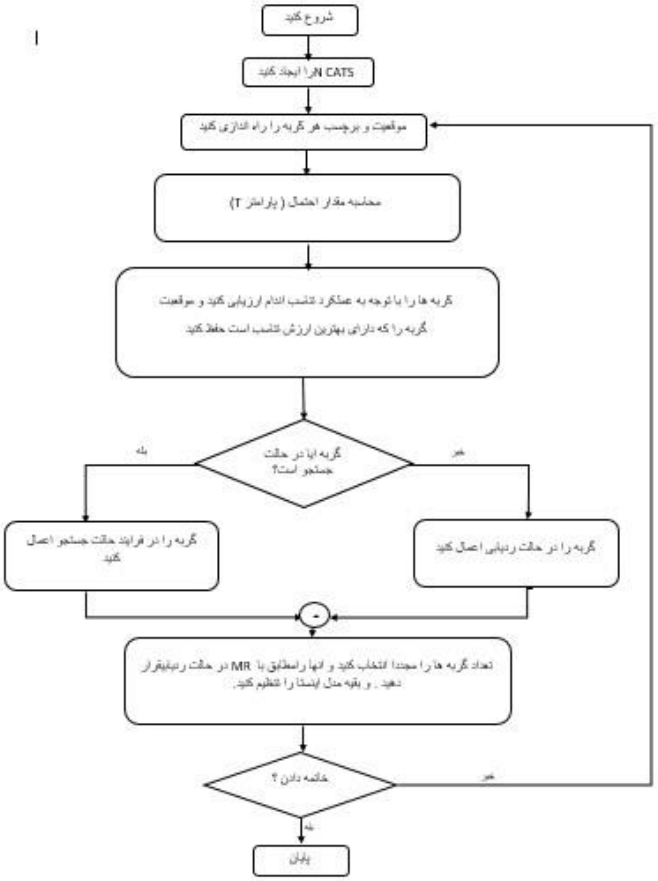
الگوریتم پیشنهادی به لحاظ ریاضیاتی مدل سازی شده و یک مدل ریاضیاتی به منظور شبیه سازی نیروهای جاذبه و دافعه بین ملخ ها ارائه شده است. نیروهای دافعه به ملخ ها این امکان را می دهند تا فضای جستجو را اکتشاف کنند در حالی که نیروهای جاذبه آنها را تشویق می کنند تا مناطق امیدوارکننده را بهره برداری کنند. به منظور تعادل بین اکتشاف و بهره برداری،ملخ با ضریبی تجهیز شده که ناحیه آسایش ملخ ها را به طور تطبیقی کاهش می دهد. نهایتاً، بهترین راه حل حاصل تاکنون توسط گروه به عنوان هدفی در نظر گرفته می شود که باید تعقیب شده و توسط ملخ ها بهبود یابد. به منظور ارزیابی عملکرد الگوریتم پیشنهادی، یک سری تست ها انجا م شده است. در ابتدا ،یک مجموعه از توابع تست دو بعدی با استفاده از ملخ حل شده اند تا عملکرد آنها به لحاظ کیفی مشاهده شود ]30[.

فرآیند جستجو را به دو بخش تقسیم می کنند: اکتشاف و بهره برداری

-در اکتشاف، عامل های جستجو به حرکت ناگهانی تشویق می گردند ،در حالی که در طول بهره برداری تمایل دارند تا به صورت محلی حرکت کنند.

فرآیند یافتن بهترین مقادیر برای متغیرهای مکانیزم سازی یک تابع هدف، بهینه سازی نامیده می شود.

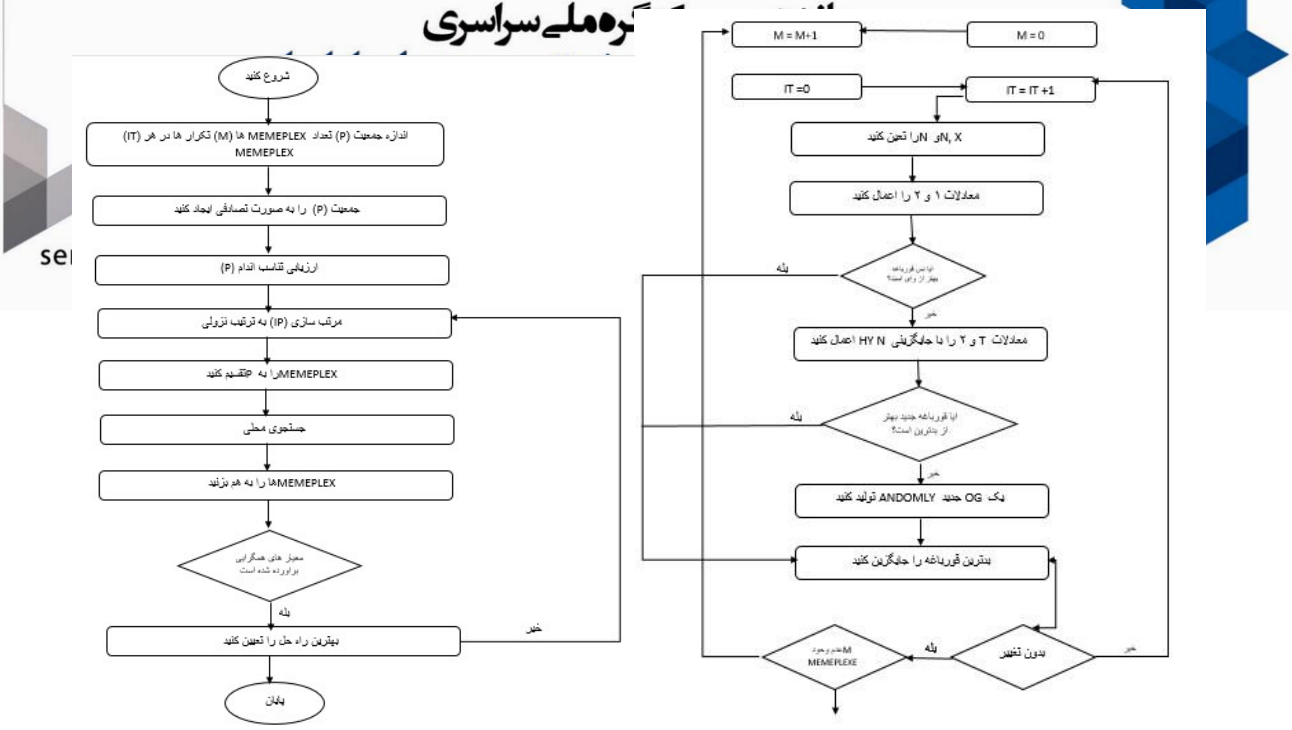
**12.3. الگوریتم ازدحام گربه ها(Cat Swarm Optimization)** CSO با مشاهده رفتارهای گربه ها تولید می شود و از دو مدل فرعی تشکیل شده است ، یعنی حالت ردیابی و حالت جستجو ، که بر اساس رفتار های گربه ها مدل سازی می شوند. نتایج تجربی با استفاده از شش تابع محک نشان می دهد که عملکرد CSO بسیار بهتر از بهینه سازی توده ذرات (PSO)است ]31[.




13. فلوچارت الگوریتم ازدحام گربه ها

**3.13. الگوریتم جهش قورباغه ها(Shuffled Frog Leaping)**

به اختصار SFLA الگوریتم جهش قورباغه،الگوریتم جهش قورباغه یا SFLA، در مقابل الگوریتم های ژنتیک که در آن صفات و قابلیت ها، توسط والدین برای فرزاندان به ارث گذاشته می شود، در الگوریتم های ممتیک، هر فردی )با توجه به نظریه فرااکتشافی لامارک( صفات و ویژگی های مفید را، با جستجو در اطراف خود )به صورت جستجوی محلی( به دست می آورد .یعنی، علاوه بر تکامل در جمعیت، تکامل به صورت فردی نیز به پیش می رود ]22[.




شکل 14. فلوچارت الگوریتم جهش قورباغه

### 14.3. الگوریتم پنگوئن ها(Penguin)

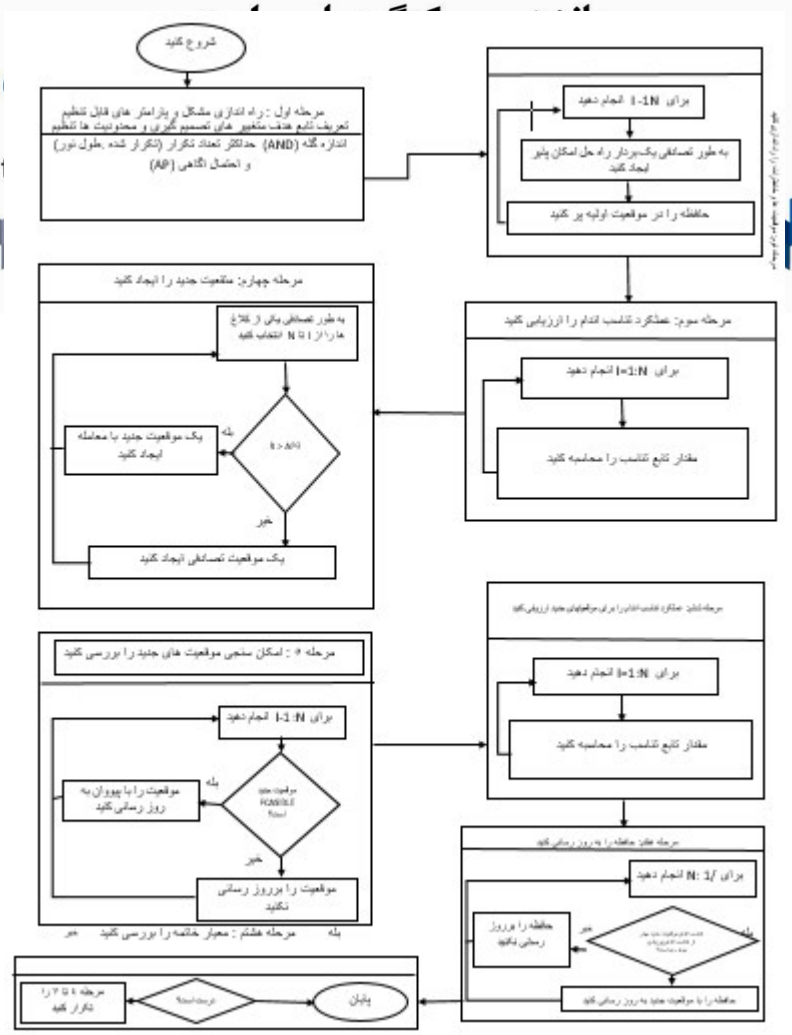
یکی از مهمترین به روز رسانی های الگوریتم گوگل که موجب به وجود آمدن سر و صدای زیادی شد، مربوط به الگوریتم پنگوئن بود.الگوریتم پنگوئن گوگل به منظور کشف وب سایت های اسپم و غیرمفید تمرکز دارد. لینک سایت و خرید بک لینک انبوه، وب سایت های بدون محتوای با کیفیت با پیج رنک بالا و لینک بلدینگ از سایت های بی کیفیت چند رویکرد این الگوریتم است. الگوریتم گوگل پنگوئن با شناسایی و جریمه وب سایت هایی که از روش های سئو کلاه سیاه استفاده می کنند، سعی می کند وب سایت های باکیفیت را در صدر موتورجستجوی گوگل قرار دهد. دیگر سئو به راحتی سال های قبل نیست و یک مشاور سئو می تواند با دیدی دقیق تر و عمیق تر راهگشا در فرایند سئو باشد ]32[ ]33[.

### 3.15. الگوریتم کلاغ ها(Crow Search Algorithm)

کلاغ ها جزو باهوش ترین پرندگان محسوب می شوند. آنها بزرگترین مغز را نسبت به اندازه بدن خود دارند. بر اساس شاخص نسبت مغز به بدن، مغز آنها کمی پایین تر از مغز انسان است. شواهد زیادی مبنی بر باهوشی و زیرکی کلاغ ها وجود دارد. آنها خودآگاهی را در آزمایشات موسوم به آزمایش آینه نشان داده اند و از توانایی ابزار سازی برخورد دارند. کلاغ ها می توانند چهره ها را بخاطر بسپارند و وقتی شخص غیر دوستانه نزدیک می شود به یکدیگر هشدار می دهند. علاوه بر این، آنها می توانند به روش های پیچیده ای ارتباط برقرار کنند و مخفی گاه غذای خود را تا چندین ماه بعد به یاد آورند ]34[ ]35[.

کلاغ ها با مشاهده مکان هایی که سایر پرندگان غذای خود را پنهان می کنند پس از رفتن صاحب آنها، غذاهای پرندگان را می دزدند. اگر کلاغی مرتکب دزدی شده باشد، اقدامات احتیاطی اضافی مانند جابجایی مکان های مخفی کاری را برای جلوگیری از قربانی شدن در آینده انجام می دهد. در واقع ، آنها از تجربه خود در مورد دزد بودن برای پیش بینی رفتار یک دزدگیر استفاده می کنند ، و می توانند ایمن ترین مسیر را برای محافظت از مخازن خود در برابر سرقت تعیین کنند. در این مقاله، بر اساس رفتارهای هوشمند فوق الذکر ،یک الگوریتم فرا ابتکاری مبتنی بر جمعیت CSA ایجاد شده است. اصول الگوریتم جستجوی کلاغ به شرح زیر ذکر شده است:

- کلاغ ها به صورت گروهی زندگی می کنند.
- کلاغ ها موقعیت مخفیگاه های خود را حفظ می کنند.
- کلاغ ها برای سرقت به دنبال هم می آیند.
- کلاغ ها از ذخیره سازی خود در برابر احتمال سرقت محافظت می کنند.




شکل 15. فلوچارت الگوریتم کلاغ ها

### 16.3. الگوریتم خفاش ها

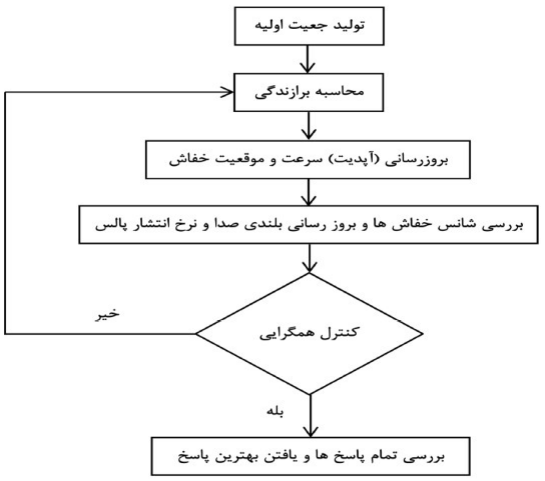
هوش جمعی یکی از قویترین تکنیک های بهینه سازی می باشد که بر مبنای رفتارهای گروهی است، که در سال 2010 توسط آقای یانگ ارائه شد. این الگوریتم بر مبنای استفاده از خاصیت انعکاس صدا توسط خفاش ها است. خفاش ها مسیر و محل دقیق طعمه خود را بوسیله فرستادن امواج صوتی و دریافت بازتاب آن، پیدا میکنند. زمانی که امواج صوتی به سمت فرستنده امواج )خفاش( باز می گردد، این پرنده می تواند یک تصویر صوتی از موانع روبروی محیط اطراف خود رسم کند و محیط اطراف را حتی در تاریکی مطلق به خوبی ببیند. با استفاده از این سیستم، خفاش ها می توانند اجسام متحرک مثل حشرات و اجسام بی حرکت مثل درختان را تشخیص دهند ]36[ ]37[
]38[ ]39[ ]40[.

الگوریتم خفاش مبتنی بر ویژگی پژواکیابی ریزخفاش ها است. در حالت کلی دو نوع خفاش وجود دارد که نوع اول بزرگ خفاش ها هستند و نوع دوم ریزخفاش ها نام دارند. ریزخفاش ها از این ویژگی برای پرواز در شب و شکار استفاده می کنند. پژواکیابی ریزخفاش ها در عمل یک سامانه ادراکی است که در آن موجهای فرا صوتی برای بدست- آوردن پژواک، تولید میشوند. مغز و دستگاه عصبی خفاش با مقایسه موجهای فرستاده شده و موج های بازتاب شده می تواند تصویری از فضای پیرامون و جزئیاتش را برای خود بسازد. این توانایی به ریزخفاش ها اجازه می دهد تا در تاریکی مطلق شکارشان را شناسایی کنند ]40[ ]41[.

ریزخفاش ها برای این هدف دو روش دارند:

- پژواکیابی با چرخه هایی با بازه زمانی کوتاه : این دسته از خفاش ها می توانند صدای فرستاده شده از سوی خود را با کمک زمان- سنجی، از صدای بازتاب شده، شناسایی کنند.

پژواک یابی با چرخه هایی با بازه های زمانی مختلف خفاش ها را قادر می کنند و پالس ها و پژواک ها را با تغییر بسامد از هم جدا می کنند. آنها می توانند بسامد به سرعت پرد از پالس هر شکار تولید شده را تغییر دهند به این ترتیب پژواک دریافتی همچنان در بازه شنوایی مناسب قرار می گیرد ]42[ ]43[.

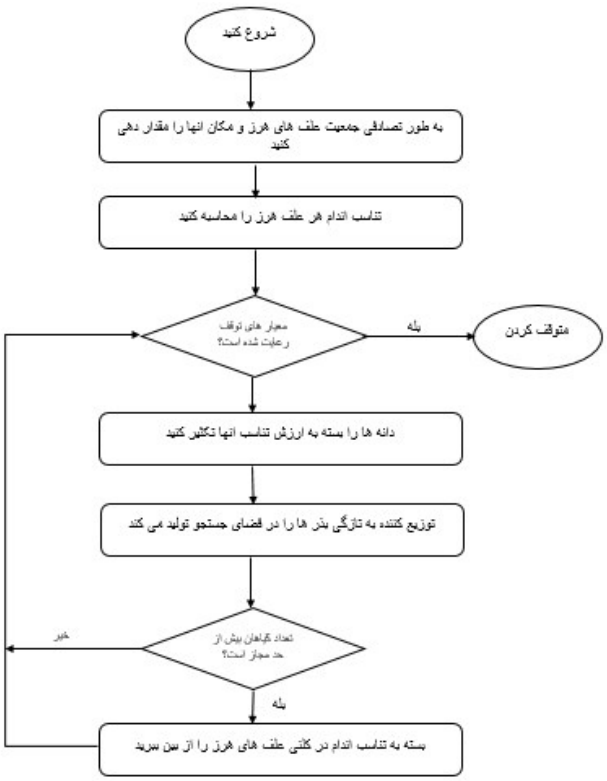



شکل 16. فلوچارت الگوریتم خفاش ها

### 3.17. الگوریتم علف هرز مهاجم (Invasive Weed Optimization)

الگوریتم بهینه سازی علف هرز مهاجم یا Invasive Weed Optimizationبه اختصار (IWO)، با الهام از روند تکثیر، بقا و تطبیق پذیری علف های هرز، ایجاد و ابداع شده است. ابداع کنندگان این الگوریتم، محمدرضا محرابیان و مرحوم پرفسور لوکس هستند.

پس از پیاده سازی IWO برای حل یک مساله نمونه نسبتا ساده، از این الگوریتم برای حل مساله بسته بندی یا Bin Packing Problem استفاده شده است، که یکی از مسائل مهم و کاربردی محسوب می شود و بسیاری از مسائل کاربردی در حوزه های مختلف مهندسی )از جمله: مهندسی صنایع، مدیریت، مخابرات و...(، پس از بررسی و ساده سازی، به صورت یک مساله از نوع بسته بندی قابل طرح هستند ]44[ ]45[ ]46[ ]47[.




شکل ۱۷. فلوچارت الگوریتم علف هرز مهاجم

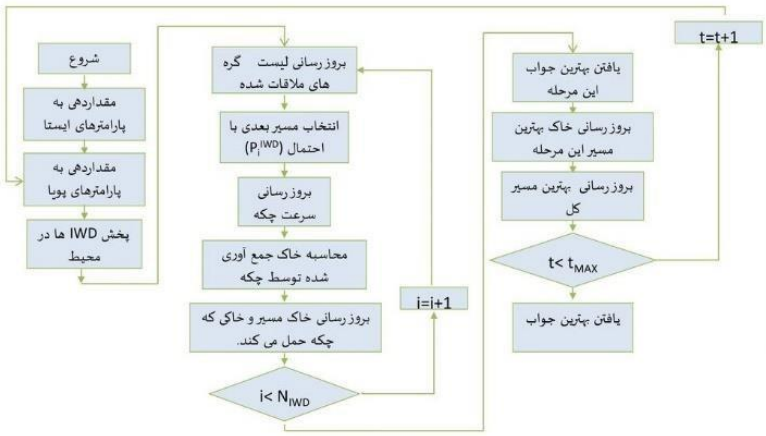
**3.18. الگوریتم چرخه آب**

روش های بهینه سازی فراابتکاری دریچه ای نو به سوی حل مسائل مهندسی و کامپیوتر و علوم پایه گشوده است. در محاسبات سنگین تحلیلی در عرصه خودروسازی و هوافضا، الگوریتم های بهینه سازی هوشمند، نقش بسزایی در بهینه کردن پارامترهای ساخت و طراحی ایفا می کنند.

آشنایی با روش های بهینه سازی، ابزاری مهم و حیاتی در اختیار طراحان، مهندسان و سازندگان قرار می دهد تا با استفاده از قابلیت جستجو موثر و توانمندی بالای این روش ها به دنبال جواب بهینه مورد نظر باشند. از طرفی طیف وسیعی از این الگوریتم ها امکان مقایسه عملکرد و سنجش این الگوریتم ها را فراهم می کند ]44[ ]48[.

چگونگی حرکت نهرها به سمت رودخانه ها و رودخانه ها به سمت دریا به شیوه ای ساده اما کاربردی در این روش بهینه سازی مدل شده است. هدف از این فرادرس، آشنایی با مفاهیم پایه ای و بنیادی الگوریتم چرخه آب به همراه آموزش برنامه نویسی این الگوریتم می باشد .

استفاده از این روش در کنار روش های دیگر می تواند برای طراحان و مهندسان گزینه مناسبی جهت بهبود طراحی و همچنین معیاری جهت مقایسه عملکرد توسط الگوریتم های دیگر باشد. در این فرادرس سعی شده است علاوه بر مطالب مربوط به مبحث الگوریتم چرخه آب، مباحث کلیدی و تکمیلی نیز عنوان شوند. اهمیت دانستن و آشنایی با الگوریتم هایی که در منابع استفاده می شوند میتواند باعث درک عمیق تری از روش های بهینه سازی شود و یا حتی می تواند منجر به ایجاد الگوریتم های ترکیبی و یا بهبود یافت شده شود ]45[.




شکل 18. فلوچارت الگوریتم چرخه آب

## 5. نتیجه گیری

با توجه به الگوریتم های فرااکتشافی نام برده شده و دیگر الگوریتم فرااکتشافی موجود که از طبیعت الهام گرفته شده اند ،میتوان با به کار گیری هوش جمعی و کوتاه ترین مسیر ها روشی جدید برای حل محاسبات پیچیده و سنگین یافت. الگوریتمهای فراابتکاری برای حل این مشکلات الگوریتمهای ابتکاری ارائه شدهاند. درواقع الگوریتمهای فراابتکاری، یکی از انواع الگوریتمهای بهینهسازی تقریبی هستند که دارای راهکارهای برونرفت از نقاط بهینه محلی هستند و قابلیت کاربرد در طیف گستردهای از مسائل را دارند. الگوریتم های فرااکتشافی مجموعه ای از اکتشافات مدرن متشکل از موفقیت در بسیاری از برنامه های کاربردی با پیچیدگی های زیادی است. الگوریتم های فرااکتشافی با یک جمعیّت اولیّه کار خود را شروع میکنند و با تکثیر نسلهای بعدی از روی والدین، سعی در حل مسائل پیچیده به بهترین شکل دارند. این الگوریتمها که زیرشاخهای از علم هوش مصنوعی هستند را میتوان در علوم نظیر برق، مکانیک، صنایع ،زیستشناسی، شیمی و رباتیک به کار گرفت. الگوریتمهای فرااکتشافی از روش ها و عملیات ابتدایی برای حل مسئله استفاده میکنند و در طی یک سری از تکرارها به راهحل مناسب برای مسئله میرسند.

## منابع

]1[ G. Rudolph and H. P. Schwefel, "Evolutionäre Algorithmen—ein robustes Optimierkonzept," *Physikalische Blätter*, vol. 50, pp. 236-238, .4991

]2[ T. Bäck, D. Fogel, and Z. Michalewicz, "Introduction to evolutionary algorithms," *Evolutionary computation*, vol. 1, pp. 59-63, 2000.

یازدهمین کنگره ملی سراسری
فناوریهای نوین در حوزه توسعه پایدار ایران
11th National Congress of
the New Technologies in Sustainable Development of Iran
sena.com.ir

]3[ X.-S. Yang, *Engineering optimization: an introduction with metaheuristic applications*: John Wiley & Sons, 2010.
]4[ J. R. Koza and J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection* vol. 1: MIT press, 1992.
]5[ R. Asgarnezhad, S. A. Monadjemi, and M. S. Aghaei, "A new hierarchy framework for feature engineering through multi - objective evolutionary algorithm in text classification," *Concurrency and Computation: Practice and Experience* ,p. e6594, 2021.
]6[ L. M. Schmitt, "Theory of genetic algorithms II: models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling," *Theoretical Computer Science*, vol. 310, pp. 181-231, .4002
]7[ D. Whitley, "A genetic algorithm tutorial," *Statistics and computing*, vol. 4, pp. 65-85, .4991
]8[ L. M. Schmitt, "Theory of genetic algorithms," *Theoretical Computer Science*, vol. 259, pp. 1-61, 2001.
]9[ M. D. Vose, *The simple genetic algorithm: foundations and theory*: MIT press, 1999.
]10[ Y. Zhang, X. Chen, D. Lv, and Y. Zhang, "Optimization of urban heat effect mitigation based on multitype ant colony algorithm," *Applied Soft Computing*, vol. 112, p. 107758, 2.021
]11[ R. Poli, W. B. Langdon, N. F. McPhee, and J. R. Koza, "A Field Guide to Genetic Programming. lulu. com," *With contributions by JR Koza*, 20.80
]12[ M. Melanie, "An introduction to genetic algorithms," *Cambridge, Massachusetts London, England, Fifth printing*, vol. 3, pp. 62-75, 1999.
]13[ H. Etemadi, A. A. A. Rostamy, and H. F. Dehkordi, "A genetic programming model for bankruptcy prediction: Empirical evidence from Iran," *Expert Systems with Applications*, vol. 36, pp. 3199-3207, 2009.
]14[ D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," 1988.
]15[ S. Chakraborty, A. K. Saha, S. Sharma, S. Mirjalili, and R. Chakraborty, "A novel enhanced whale optimization algorithm for global optimization," *Computers & Industrial Engineering*, vol. 153, p. 107086, 2021.
]16[ R. Asgarnezhad, S. A. Monadjemi, and M. Soltanaghaei, "An application of MOGW optimization for feature selection in text classification," *The Journal of Supercomputing*, vol. 77, pp. 5806-5839, 2021.
]17[ O. N. Almasi and M. Rouhani, "A new fuzzy membership assignment and model selection approach based on dynamic class centers for fuzzy SVM family using the firefly algorithm," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 24, pp. 1797-1814, 201.6
]18[ V. Kumar and D. Kumar, "A systematic review on firefly algorithm: past, present, and future," *Archives of Computational Methods in Engineering*, vol. 28, pp. 3269-3291, 2021.
]19[ M. A. Lones, "Metaheuristics in nature-inspired algorithms," in *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 2014, pp. 1419-1422.
]20[ X.-S. Yang, "Mathematical analysis of nature-inspired algorithms," in *Nature-inspired algorithms and applied optimization* ,ed: Springer, 2018, pp. 1-25.
]21[ A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm," *Computers & Structures*, vol. 169, pp. 1-12, 2016.
]22[ Y. Huang, X.-N. Shen, and X. You, "A discrete shuffled frog-leaping algorithm based on heuristic information for traveling salesman problem," *Applied Soft Computing*, vol. 102, p. 107085, 2021.
]23[ F. Wang, H. Zhang, and A. Zhou, "A particle swarm optimization algorithm for mixed-variable optimization problems," *Swarm and Evolutionary Computation*, vol. 60, p. 100808, 2021.
]24[ S. Gao, Y. Gao, Y. Zhang, and T. Li, "Adaptive cuckoo algorithm with multiple search strategies," *Applied Soft Computing*, p. 107181, 2021.
]25[ J. García and C. Maureira, "A KNN quantum cuckoo search algorithm applied to the multidimensional knapsack problem," *Applied Soft Computing*, vol. 102, p. 107077, 2021.
]26[ D. Weyland, "A critical analysis of the harmony search algorithm—How not to solve sudoku," *Operations Research Perspectives*, vol. 2, pp. 97-105, 2015.
]27[ A. W. Mohamed, A. A. Hadi, and A. K. Mohamed, "Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm," *International Journal of Machine Learning and Cybernetics*, vol. 11, pp. 1501-1529, 2020.

]28[ A. Afzal, A. M. Samee, R. Kiltel, M. Mujtaba, A. M. Mahalkari, and K. A. Razak, "Battery thermal management: an optimization study of parallelized conjugate numerical analysis using Cuckoo search and Artificial bee colony algorithm," *International Journal of Heat and Mass Transfer*, vol. 166, p. 120798, 2021.

]29[ P. E. Mergos and X.-S. Yang, "Flower pollination algorithm parameters tuning," *Soft computing*, vol. 25, pp. 14429-14447, 2021.

]30[ Y. Meraihi, A. B. Gabis, S. Mirjalili, and A. Ramdane-Cherif, "Grasshopper optimization algorithm: theory, variants, and applications," *IEEE Access*, vol. 9, pp. 50001-50024, .1202

]31[ R. Ihsan, S. Almufti, B. Ormani, R. Asaad, and R. Marqas, "A Survey on Cat Swarm Optimization Algorithm," *Asian Journal of Research in Computer Science*, pp. 22-32, 2021.

]32[ P. D. Justesen, "Multi-objective optimization using evolutionary algorithms," *University of Aarhus, Department of Computer Science, Denmark*, vol. 33, 2009.

]33[ G .Dhiman, D. Oliva, A. Kaur, K. K. Singh, S. Vimal, A. Sharma, *et al.*, "BEPO: a novel binary emperor penguin optimizer for automatic feature selection," *Knowledge-Based Systems*, vol. 211, p. 106560, 2021.

]34[ M. Gendreau and J.-Y. Potvin, *Handbook of metaheuristics* vol. 2: Springer, 2010.

]35[ Y. Meraihi, A. B. Gabis, A. Ramdane-Cherif, and D. Acheli, "A comprehensive survey of Crow Search Algorithm and its applications," *Artificial Intelligence Review*, vol. 54, pp. 2669-2716, 2021.

]36[ 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.

]37[ X.-S. Yang, "Bat algorithm for multi-objective optimisation," *International Journal of Bio-Inspired Computation*, vol. 3, pp. 267-274, 2011.

]38[ G. Komarasamy and A. Wahi, "An optimized K-means clustering technique using bat algorithm," *European Journal of Scientific Research*, vol. 84, pp. 263-273, 2012.

]39[ Y. Wang, P. Wang, J. Zhang, Z. Cui, X. Cai, W. Zhang, *et al.*, "A novel bat algorithm with multiple strategies coupling for numerical optimization," *Mathematics*, vol. 7, p. 135, 2019.

]40[ G. Wang and L. Guo, "A novel hybrid bat algorithm with harmony search for global numerical optimization ",*Journal of Applied Mathematics*, vol. 2013, 2013.

]41[ K. Khan, A. Nikov, and A. Sahai, "A fuzzy bat clustering method for ergonomic screening of office workplaces," in *Third international conference on software, services and semantic technologies S3T 2* ,2011 ,011pp. 59-66.

]42[ X.-S. Yang and X. He, "Bat algorithm: literature review and applications," *International Journal of Bio-inspired computation*, vol. 5, pp. 141-149, 2013.

]43[ X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature inspired cooperative strategies for optimization (NICSO 2010)*, ed: Springer, 2010, pp. 65-74.

]44[ X.-S. Yang, "Flower pollination algorithm for global optimization," in *International conference on unconventional computing and natural computation*, 2012, pp..249-240

]45[ A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological informatics*, vol. 1, pp. 355-366, 2006.

]46[ J. Dekker, "The Evolutionary Ecology of Weeds and Invasive Plants," *Evolution ecology*, vol. 12, p. 08, 2009.

]47[ W. Wang, G. Tian, G. Yuan, and D. T. Pham, "Energy-time tradeoffs for remanufacturing system scheduling using an invasive weed optimization algorithm," *Journal of Intelligent Manufacturing*, pp. 1-19, 2021.

]48[ Y.-g. Zhang, J. Tang, R.-p. Liao, M.-f. Zhang, Y. Zhang, X.-m. Wang, *et al.*, "Application of an enhanced BP neural network model with water cycle algorithm on landslide prediction," *Stochastic Environmental Research and Risk Assessment*, vol. 35, pp. 1273-1291, .1202