

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشکده: عمران

گروه: عمران - سازه

مقایسه عملکرد روش های نوین بهینه سازی سازه ها
(روش جزیره ای الگوریتم ژنتیک، روش ACO، روش CSS)

دانشجو:

صادق عابدی

استاد راهنما:

دکتر وحید رضا کلاتجاری

استاد مشاور:

مهندس محمد حسین طالب پور

پایان نامه ارشد جهت اخذ درجه کارشناسی ارشد

شهریور ۱۳۹۱

دانشگاه صنعتی شاهرود

دانشکده: عمران

گروه: عمران - سازه

پایان نامه کارشناسی ارشد آقای صادق عابدی

تحت عنوان:

مقایسه عملکرد روش های نوین بهینه سازی سازه ها

(روش جزیره ای الگوریتم ژنتیک، روش ACO، روش CSS)

در تاریخ توسط کمیته تخصصی زیر جهت اخذ مدرک کارشناسی ارشد
(رساله دکتری) مورد ارزیابی و با درجه مورد پذیرش قرار گرفت.

امضاء	اساتید مشاور	امضاء	اساتید راهنما
	نام و نام خانوادگی : مهندس محمد حسین طالب پور		نام و نام خانوادگی : دکتر وحید رضا کلات جاری

امضاء	نماینده تحصیلات تکمیلی	امضاء	اساتید داور
	نام و نام خانوادگی : سید علی حسینی		نام و نام خانوادگی : دکتر سید فضل ا... ساغروانی
			نام و نام خانوادگی : دکتر پوریا منصوریان
			نام و نام خانوادگی :
			نام و نام خانوادگی :

تقدیم بہ:

تو ای پدر از تو هر چه می گویم باز هم کم می آورم
خوشیدی شدی و از روشنائی ات جان گرفتم و در ناامیدی ماندم را کشیدی
و لبریزم کردی از شوق اکنون حاصل دستان خسته ات رمز موفقیتم شد
به خودم تبریک می گویم که تو را دارم و دنیا با همه بزرگیش مثل تو را ندارد.....
و تو ای مادر، ای شوق زیبائی نفس کشیدن، ای روح مهربان، هستی ام
تو رنگ شادی هایم شدی و لحظه ها را با تمام وجود از من دور کردی و
عمری حسنی را به جان خریدی تا اکنون توانستی طعم خوش پیروزی را به من بچشانی

تشکر و قدردانی

سپاس لطف و عنایت الهی را که توفیق اتمام این پژوهش را به من ارزانی داشت.

باسپاس فراوان از استاد خوبم آقای دکتر وحید رضا کلات جاری که با تواضع و صبر در به ثمر رسیدن این پژوهش گام به گام مرا یاری رساندند.

باسپاس از جناب آقای مهندس طالب پور بابت همکاری صمیمانه ایشان.

دانشجو تأیید می نماید که مطالب مندرج در این پایان نامه (رساله) نتیجه تحقیقات خودش می باشد و در صورت استفاده از نتایج دیگران مرجع آن را ذکر نموده است.

کلیه حقوق مادی مترتب از نتایج مطالعات ، آزمایشات و نوآوری ناشی از تحقیق موضوع این پایان نامه (رساله) متعلق به دانشگاه صنعتی شاهرود می باشد .

شهریور ۱۳۹۱

چکیده:

اکثر الگوریتم های بهینه یابی که بر مبنای روش های برنامه نویسی عددی خطی و غیر خطی می باشند نیازمند اطلاعات مشتق تابع هدف هستند و از این اطلاعات استفاده می کنند تا محدوده جواب ها نزدیک به یک نقطه شروع اولیه را مورد جستجو قرار دهند. بدین دلیل در بکارگیری این الگوریتم ها در مسائل بهینه یابی دنیای طبیعی اطراف ما محدودیت وجود دارد. اگر بیش از یک مینیمم محلی در مسئله بهینه یابی وجود داشته باشد نتیجه نهایی به مقدار نقطه شروع اولیه وابسته خواهد بود و حل بهینه حاصل الزاماً بهینه مطلق نخواهد بود. به همین دلیل در چند دهه اخیر استفاده از الگوریتم های فرا ابتکاری جهت بهینه سازی توجه بسیاری از محققین را به خود جلب نموده است. این روش ها که بر پایه الهام از فرآیندهای طبیعی بنا نهاده شده اند، به علت عدم نیاز به انجام محاسبات سنگین ریاضی و نیز قابلیت نیل به نقطه بهینه سراسری و یا نزدیک به آن، در حل بسیاری از مسائل مختلف مورد استفاده قرار می گیرند. به دلیل اهمیت بالای مسائل بهینه یابی در علوم مهندسی و استفاده از الگوریتم های فرا ابتکاری در حل این مسائل، در این پایان نامه با بهره گیری از روش های نوین الگوریتم ژنتیک، اجتماع مورچگان، جستجوی هارمونی و جستجوی ذرات باردار شیوه جدیدی جهت بهینه یابی سطح مقطع سازه های خرپایی، بصورت همزمان، ارائه شده است تا بر اساس آن، فضای طراحی مساله به صورت مناسبی مورد کاوش قرار گیرد و جواب های نزدیک به جواب بهینه حاصل گردد. همچنین با طرح مثال های متعدد نتایج حاصل از سایر الگوریتم ها و این روش ابتکاری که روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری نامیده می شود مورد نقد و بررسی قرار گرفته است و کارآمد بودن این روش ابتکاری را نشان می دهد.

کلمات کلیدی:

بهینه سازی، سازه های خرپایی، الگوریتم ژنتیک، الگوریتم اجتماع مورچگان، روش جستجوی هارمونی، روش جستجوی ذرات باردار، روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری

مقالات مستخرج از پایان نامه

۱- "مقایسه و بررسی روند بهینه سازی سازه های خرپایی با استفاده از روش جستجوی چند منظوره و روش جستجوی ذرات باردار". اولین کنفرانس ملی عمران و توسعه (اسفند ۱۳۹۰)

۲- "بهینه یابی سازه های خرپایی با استفاده از روش جبری نیروها و برنامه ریزی غیر خطی". دومین کنفرانس ملی مدیریت بحران، زلزله و آسیب پذیری اماکن و شریانهای حیاتی (خرداد ۱۳۹۰)

۳- "مقایسه روند بهینه یابی سازه های خرپایی با استفاده از الگوریتم ترکیبی اجتماع مورچگان و جستجوی هارمونی و روش جستجوی چند منظوره". دومین کنفرانس ملی مدیریت بحران، زلزله و آسیب پذیری اماکن و شریانهای حیاتی (خرداد ۱۳۹۰)

فصل اول: مقدمه

- ۱-۱ پیشگفتار..... ۲
- ۲-۱ روش‌های بهینه‌سازی..... ۳
- ۱-۲-۱ دسته بندی مسائل بهینه سازی سازه..... ۵
- ۳-۱ اهداف تحقیق..... ۶
- ۴-۱ ساختار پایان نامه..... ۷

فصل دوم: الگوریتم های فراابتکاری

- ۱-۲ مقدمه..... ۹
- ۲-۲ الگوریتم های فرا ابتکاری..... ۹
- ۳-۲ مفاهیم الگوریتم ژنتیک ۱۴
- ۱-۳-۲ تاریخچه الگوریتم ژنتیک..... ۱۴
- ۲-۳-۲ بهینه یابی با الگوریتم ژنتیک..... ۱۶
- ۱-۲-۳-۲ کدگذاری کروموزوم ها..... ۱۸
- ۱-۱-۲-۳-۲ کد گذاری باینری..... ۱۸
- ۲-۲-۳-۲ شایستگی هر کروموزوم..... ۲۱
- ۱-۲-۲-۳-۲ تابع پناالتی..... ۲۱
- ۲-۲-۲-۳-۲ تابع شایستگی..... ۲۲
- ۳-۲-۳-۲ انتخاب..... ۲۴
- ۴-۲-۳-۲ مفهوم برگزیده(شایسته سالاری)..... ۲۵
- ۵-۲-۳-۲ عملگر های الگوریتم ژنتیک..... ۲۵
- ۱-۵-۲-۳-۲ عملگر ادغام..... ۲۶
- ۱-۱-۵-۲-۳-۲ نرخ ادغام..... ۲۷

۲۷.....	۲-۳-۲-۵ جهش
۲۸.....	۲-۳-۲-۵-۱ عمل جهش در کد گذاری دودویی
۲۸.....	۲-۳-۲-۵-۲ نرخ جهش
۳۰.....	۲-۳-۲-۶ دستورخروج از الگوریتم
۳۱.....	۲-۴ الگوریتم اجتماع مورچگان
۳۱.....	۲-۴-۱ مقدمه ای بر الگوریتم اجتماع مورچگان
۳۲.....	۲-۴-۲ متا هیوریستیک بهینه سازی کلونی مورچگان
۳۲.....	۲-۴-۱ رفتار در جستجوی غذای مورچگان
۳۳.....	۲-۴-۲ آزمایشات پل
۳۶.....	۲-۴-۳ از مورچه های طبیعی تا روش فراکاوشی کلونی مورچگان
۳۷.....	۲-۴-۴ شباهت ها و تفاوت های بین مورچه های واقعی و مصنوعی
۳۸.....	۲-۴-۵ استیگمرجی و فرومون مصنوعی
۴۰.....	۲-۴-۶ شکل ساده ای از بهینه سازی کلونی مورچگان
۴۴.....	۲-۴-۷ سیستم مورچه
۴۷.....	۲-۴-۸ سیستم کلونی مورچه ها
۵۱.....	۲-۴-۹ سیستم مورچگان رتبه بندی شده
۵۲.....	۲-۵ الگوریتم جستجوی هارمونی
۵۸.....	۲-۵-۱ مراحل بهینه یابی با جستجوی هارمونی
۵۸.....	۲-۵-۱-۱ مقداردهی اولیه به مسئله بهینه یابی و پارامتر های الگوریتم
۵۹.....	۲-۵-۱-۲ مقدار دهی اولیه به حافظه هارمونی
۶۰.....	۲-۵-۱-۳ تولید یک هارمونی جدید
۶۰.....	۲-۵-۱-۴ به روزرسانی حافظه هارمونی
۶۰.....	۲-۵-۱-۵ معیار توقف
۶۱.....	۲-۵-۲ بررسی پارامتر های الگوریتم جستجوی هارمونی
۶۴.....	۲-۶ الگوریتم جستجوی ذرات باردار

۶۴.....	۲-۶-۱ مرور بر قوانین الکتریکی و مکانیکی
۶۴.....	۲-۶-۱-۱ قوانین الکتریکی
۶۷.....	۲-۶-۱-۲ قوانین مکانیک برداری
۶۹.....	۲-۶-۲ روش جستجوی بار
۷۷.....	۲-۶-۳ مختصری از مراحل الگوریتم جستجوی ذرات باردار
۷۷.....	۲-۶-۳-۱ مرحله مقدار دهی اولیه
۷۸.....	۲-۶-۳-۲ مرحله جستجو
۷۸.....	۲-۶-۳-۳ مرحله کنترل معیار خاتمه

فصل سوم: بهینه یابی سطح مقطع سازه ها با الگوریتم های فرا ابتکاری

۸۰.....	۳-۱ مقدمه
۸۱.....	۳-۲ فرمولبندی مسأله بهینه یابی مقاطع بر اساس الگوریتم های فرا ابتکاری
۸۲.....	۳-۲-۱ متغیر های طراحی
۸۲.....	۳-۲-۲ تابع هدف
۸۳.....	۳-۲-۳ قیود $G1$ و $G2$
۸۵.....	۳-۲-۴ قیود $G3$ و $G4$
۸۶.....	۳-۲-۵ قید $G5$

فصل چهارم: روش پیشنهادی و راهنمای برنامه

۸۸.....	۴-۱ مقدمه
۹۰.....	۴-۲ روش جستجوی چند منظوره مبتنی بر الگوریتم ژنتیک (ژنتیک جزیره ای)
۹۴.....	۴-۲-۱ الگوریتم تولید جمعیت اولیه
۹۴.....	۴-۲-۲ رمز گشایی هر کروموزوم
۹۶.....	۴-۲-۳ ماتریس سختی هر کروموزوم

- ۴-۲-۴ تحلیل سازه مربوط به هر کروموزوم..... ۹۶
- ۴-۲-۵ تابع هدف و تابع جریمه مورد استفاده..... ۹۷
- ۴-۲-۶ تابع هدف اصلاح شده و تابع شایستگی مورد استفاده..... ۹۸
- ۴-۲-۷ مشخصات جزایر درروش جستجوی چند منظوره مبتنی بر الگوریتم ژنتیک..... ۹۹
- ۴-۲-۷-۱ جزیره (۱)..... ۹۹
- ۴-۲-۷-۲ جزیره (۲)..... ۱۰۱
- ۴-۲-۷-۳ جزیره (۳)..... ۱۰۲
- ۴-۲-۷-۴ جزیره (۴)..... ۱۰۳
- ۴-۲-۷-۵ جزیره (۵)..... ۱۰۴
- ۴-۳ روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری..... ۱۰۶
- ۴-۳-۱ معیار مقایسه طرح ها در هنگام مهاجرت..... ۱۱۲
- ۴-۳-۲ سازگار شدن طرح های مهاجر با طرح های جزایر مقصد..... ۱۱۲
- ۴-۳-۳ بررسی همگرایی و کنترل تعداد مراحل نسل سازی..... ۱۱۳
- ۴-۳-۴ مشخصات جزایر درروش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری..... ۱۱۴
- ۴-۳-۴-۱ جزیره (۱)..... ۱۱۴
- ۴-۳-۴-۲ جزیره (۲)..... ۱۱۷
- ۴-۳-۴-۳ جزیره (۳)..... ۱۱۸
- ۴-۳-۴-۴ جزیره (۴)..... ۱۱۹

فصل پنجم: مثال عددی

- ۵-۱ مقدمه..... ۱۲۲
- ۵-۲ مثال (۱)..... ۱۲۲
- ۵-۳ مثال (۲)..... ۱۲۹

۴-۵ مثال (۳)..... ۱۳۶

۵-۵ مثال (۴)..... ۱۴۵

۶-۵ مثال (۵)..... ۱۵۲

فصل ششم: نتیجه‌گیری و پیشنهادات

۱-۶ مقدمه..... ۱۶۰

۲-۶ نتایج..... ۱۶۰

۳-۶ پیشنهادات..... ۱۶۲

فهرست منابع..... ۱۶۳

فهرست اشکال

فصل دوم

- شکل (۱-۲) کدگذاری باینری ۱۸
- شکل (۲-۲) نحوه‌ی بدست آوردن اعداد در مبنای ۲ ۱۹
- شکل (۳-۲) عمل جهش ۲۸
- شکل (۴-۲) آزمایش پل دنبرگ ۳۴
- شکل (۵-۲) آزمایش پل با طول های نامساوی ۳۵
- شکل (۶-۲) رفتار مورچه در یافتن کوتاه ترین مسیر بین لانه و منبع غذا ۳۶
- شکل (۷-۲) مساله کوتاه ترین مسیر در یک گراف ۴۰
- شکل (۸-۲) مقایسه ای بین کار نوازندگان و روش های بهینه یابی ۵۳
- شکل (۹-۲) ساختار حافظه هارمونی ۵۴
- شکل (۱۰-۲) مراحل بهینه یابی الگوریتم جستجوی هارمونی ۵۹
- شکل (۱۱-۲) ترازوی پیچشی کولمب ۶۵
- شکل (۱۲-۲) نمودار E_{ij} در برابر r_{ij} برای یک گوی عایق با باریکناخت ۶۶
- شکل (۱۳-۲) تعیین برآیند نیروی الکتریکی وارد بر یک ذره باردار ۷۲
- شکل (۱۴-۲) مقایسه‌ای بین معادلات $F_{ij} \propto \frac{1}{r_{ij}^2}(a$ و $F_{ij} \propto r_{ij}(b$ وقتی که $r_{ij} < a$ ۷۳
- شکل (۱۵-۲) حرکت ذره باردار به موقعیت جدید ۷۵

فصل چهارم

- شکل (۱-۴) روش جستجوی چند منظوره ۹۰
- شکل (۲-۴) فلوچارت الگوریتم جستجوی چند منظوره مبتنی بر ژنتیک ۹۲
- شکل (۳-۴) فلوچارت روش جستجوی چند منظوره مبتنی بر ژنتیک ۹۳

- شکل (۴-۴) پنجره ورودی اطلاعات جزیره (1)..... ۱۰۰
- شکل (۵-۴) پنجره ورودی اطلاعات جزیره (2)..... ۱۰۱
- شکل (۶-۴) پنجره ورودی اطلاعات جزیره (3)..... ۱۰۲
- شکل (۷-۴) پنجره ورودی اطلاعات جزیره (4)..... ۱۰۳
- شکل (۸-۴) پنجره ورودی اطلاعات جزیره (5)..... ۱۰۴
- شکل (۹-۴) پنجره ورودی اطلاعات GA و فراخونی جزایر ۱۰۵
- شکل (۱۰-۴) فلو چارت روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری..... ۱۰۷
- شکل (۱۱-۴) فلو چارت مربوط به جزیره ۱..... ۱۰۸
- شکل (۱۲-۴) فلو چارت مربوط به جزیره ۲..... ۱۰۹
- شکل (۱۳-۴) فلوچارت مربوط به جزیره ۳..... ۱۱۰
- شکل (۱۴-۴) انتخاب مقدار برای هر متغیر تصمیم در الگوریتم جستجو هارمونی..... ۱۱۰
- شکل (۱۵-۴) فلوچارت مربوط به جزیره ۴..... ۱۱۱
- شکل (۱۶-۴) بررسی معیار خاتمه..... ۱۱۱
- شکل (۱۷-۴) اطلاعات جزیره (۱)..... ۱۱۴
- شکل (۱۸-۴) اندازه سائز رقابت در روش تورنمنت..... ۱۱۵
- شکل (۱۹-۴) مشخصات پیوند در جزیره (۱)..... ۱۱۵
- شکل (۲۰-۴) مشخصات جهش در جزیره (۱)..... ۱۱۶
- شکل (۲۱-۴) اطلاعات جزیره (۲)..... ۱۱۷
- شکل (۲۲-۴) اطلاعات جزیره (۳)..... ۱۱۸
- شکل (۲۳-۴) اطلاعات جزیره (۴)..... ۱۱۹
- شکل (۲۴-۴) پنجره ورودی اطلاعات MSM و فراخونی جزایر..... ۱۲۰

فصل پنجم

- شکل (۵-۱) خرپای دو بعدی ۴۷ عضوی..... ۱۲۳
- شکل (۵-۲) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۴۷ عضوی..... ۱۲۷
- شکل (۵-۳) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خرپای ۴۷ عضوی..... ۱۲۸
- شکل (۵-۴) خرپای فضایی ۷۲ عضوی..... ۱۳۰
- شکل (۵-۵) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۷۲ عضوی..... ۱۳۴
- شکل (۵-۶) بزرگ نمایی شکل (۵-۵) ۱۳۵
- شکل (۵-۷) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خرپای ۴۷ عضوی..... ۱۳۶
- شکل (۵-۸) خرپای دو بعدی ۲۰۰ عضوی با ۲۹ گروه..... ۱۳۷
- شکل (۵-۹) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۲۰۰ عضوی ۱۴۳
- شکل (۵-۱۰) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خرپای ۲۰۰ عضوی ۲۹ گروهی..... ۱۴۴
- شکل (۵-۱۱) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۲۰۰ عضوی ۱۵۱

شکل (۵-۱۲) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای

خرپای ۲۰۰ عضوی ۹۶ گروهی..... ۱۵۲

شکل (۵-۱۳) نمای سه بعدی و جانبی خرپای فضایی ۹۴۲ عضوی..... ۱۵۴

شکل (۵-۱۴) نمای از بالای خرپای فضایی ۹۴۲ عضوی..... ۱۵۵

شکل (۵-۱۵) روند بهینه یابی توسط سایر روش ها برای خرپای ۹۴۲ عضوی..... ۱۵۸

فهرست جداول

فصل پنجم

جدول (۵-۱) بارهای موثر وارد بر خرپای ۴۷ عضوی..... ۱۲۴

جدول (۵-۲) لیست مقاطع خرپای ۴۷ عضوی..... ۱۲۴

جدول (۵-۳) گروه بندی اعضای خرپای ۴۷ عضوی..... ۱۲۴

جدول (۵-۴) پارامتر های مرتبط با روش پیشنهادی در خرپای ۴۷ عضوی..... ۱۲۵

جدول (۵-۵) نتایج طرح بهینه خرپای ۴۷ عضوی..... ۱۲۶

جدول (۵-۶) بارهای موثر وارد بر خرپای فضایی ۷۲ عضوی..... ۱۳۰

جدول (۵-۷) لیست مقاطع خرپای ۷۲ عضوی..... ۱۳۱

جدول (۵-۸) گروه بندی اعضای خرپای ۷۲ عضوی..... ۱۳۱

جدول (۵-۹) پارامتر های مرتبط با روش پیشنهادی در خرپای ۷۲ عضوی..... ۱۳۲

جدول (۵-۱۰) نتایج طرح بهینه خرپای ۷۲ عضوی..... ۱۳۳

جدول (۵-۱۱) گروه بندی اعضای خرپای ۲۰۰ میله ای (۲۹ گروه)..... ۱۳۸

جدول (۵-۱۲) بارگذاری مؤثر بر خرپای ۲۰۰ میله ای (حالت ۱)..... ۱۳۹

جدول (۵-۱۳) بارگذاری مؤثر بر خرپای ۲۰۰ میله ای (حالت ۲)..... ۱۳۹

جدول (۵-۱۴) بارگذاری مؤثر بر خرپای ۲۰۰ میله ای (حالت ۳)..... ۱۴۰

- جدول (۵-۱۵) لیست مقاطع خرپای ۲۰۰ عضوی..... ۱۴۰
- جدول (۵-۱۶) پارامتر های مرتبط با روش پیشنهادی در خرپای ۲۰۰ عضوی..... ۱۴۱
- جدول (۵-۱۷) نتایج طرح بهینه خرپای ۲۰۰ عضوی با ۲۹ گروه..... ۱۴۲
- جدول (۵-۱۸) گروه بندی اعضا ی خرپای ۲۰۰ میله ای (۹۶گروه)..... ۱۴۶
- جدول (۵-۱۹) پارامتر های مرتبط با روش پیشنهادی در خرپای ۲۰۰ عضوی ۹۶ گروهی..... ۱۴۷
- جدول (۵-۲۰) نتایج طرح بهینه خرپای ۲۰۰ عضوی با ۹۶ گروه..... ۱۴۸
- جدول (۵-۲۱) نتایج طرح بهینه خرپای ۹۴۲ عضوی..... ۱۵۶

فصل اول

مقدمه

۱-۱ پیشگفتار

بهینه سازی، یافتن بهترین جواب از بین جواب های موجود برای حل مسئله می باشد. بهینه سازی در تمام زمینه ها همواره از اهمیت بالایی برخوردار بوده است، به طوری که انتخاب و طراحی بهینه در بسیاری از مسائل علمی و فنی باعث تولید بهترین محصول یا جواب ممکن در یک شرایط خاص می شود. به عنوان مثال در زمینه های مهندسی از آنجا که هزینه مصالح یکی از فاکتورهای مهم در ساخت سازه ها می باشد، طراحی بهینه سازه ها به طوری که به کاهش وزن و حجم سازه منجر شود، بسیار مناسب خواهد بود. زیرا هم هزینه ها کاهش می یابد و هم در هنگام زلزله به سازه نیروی کمتری وارد می شود. از دیدگاه علوم ریاضی فرآیند محاسبه حداقل یا حداکثر، تابع هدف بهینه سازی نامیده می شود که در طرح های عملی بر حسب نوع تابع هدف، یکی از مقادیر حدی آن، که به طرح بهینه موسوم است مشخص می شود. بهینه سازی یک سیستم کاربردی همزمان با اقلان شرایط طراحی حاکم بر آن انجام می شود که شرایط طراحی، قید های مساله بهینه سازی می باشند. مساله بهینه سازی تابع هدف با حضور قیود و محدودیت های طراحی، مساله بهینه سازی مقید نامیده می شود. پارامترهای تابع هدف، متغیر های مساله بهینه سازی می باشند. مقادیر این پارامترها ویژگیهای سیستم مورد بحث را نشان می دهند و با تغییر آن ها طرح های مختلفی برای سیستم ایجاد خواهد شد. مقادیر متغیر ها از مجموعه ای تحت عنوان مجموعه مرجع انتخاب می شوند. بدیهی است که مجموعه مرجع بخشی از دامنه تابع هدف می باشد.

بنابراین می توان بهینه سازی را به عنوان فرایند یافتن شرایطی که مقدار بیشینه و یا کمینه یک تابع را به دست می دهد، تعریف نمود. به عبارت دیگر طراحی بهینه عبارتست از: «بهترین طراحی قابل قبول بر اساس یک معیار کیفی و کمی شایستگی از پیش تعریف شده» [۱].

۱-۲ روشهای بهینه سازی

تاریخ علم بهینه سازی به اوایل قرن نوزدهم بر می گردد. مسایل بهینه سازی با روش های متفاوتی مورد تحلیل قرار می گیرند. این روشها شامل شیوه های تحلیلی نظیر حساب تغییرات^۱، حساب دیفرانسیل^۲، روش مضارب لاگرانژ^۳ و روش های عددی مانند برنامه ریزی خطی^۴، روش های جستجوی مستقیم^۵، روش های مبتنی بر گرادیان^۶، روش های تابع جریمه^۷، روشهای دوگانه^۸ و روش های معیار بهینه گی^۹ می باشند [۲].

روش های تحلیلی بر اساس مفهوم مشتق جزئی توابع چند متغیره بیان شده اند. از آنجا که مفهوم مشتق فقط برای توابع پیوسته تعریف می شود، لذا محاسبه بهینه توابع گسسته با این روش ها امکان پذیر نیست. محاسبات روش های تحلیلی از یکی از نقاط دامنه تابع آغاز می شود و با توجه به گرادیان تابع هدف به صورت خطی به نقطه دیگری منتقل می شود. محاسبات مذکور برای نقطه جدید تکرار می شود تا نقاط بعدی بدست آیند. بدیهی است که عملیات تکراری روشهای تحلیلی تا حصول همگرایی ادامه می یابد [۲]. انجام عملیات بهینه سازی بر نقاط خاصی از دامنه تابع هدف و همچنین حرکت خطی میان نقاط مذکور، در بسیاری از موارد باعث همگرایی به نقطه بهینه محلی می شود. این ویژگی در شرایطی که تابع هدف دارای تغییرات شدید در دامنه خود باشد تشدید می یابد. برای جبران این نقطه ضعف الگوریتم های بهینه سازی تحلیلی باید در دفعات با نقاط آغازین مختلف تکرار شود تا پس از مقایسه نتایج آنها اطمینان کامل از مقادیر محاسبه شده حاصل گردد.

¹. Calculus of Variation

². Differential Calculus

³. Lagrange Multipliers

⁴. Linear Programming

⁵. Direct Search Methods

⁶. Gradient Based Methods

⁷. Penalty Function Methos

⁸. Dual Methods

⁹. Optimality Criteria Methods

همچنین روش های عددی از قبیل برنامه ریزی خطی، برنامه ریزی خطی همگن، برنامه ریزی عدد صحیح، برنامه ریزی پویا (دینامیکی) و برنامه ریزی غیر خطی که برای حل مسائل بهینه سازی به کار برده می شوند اطلاعات گرایانی را استفاده می کنند تا محدوده جواب ها نزدیک به یک نقطه شروع اولیه جستجو گردد. در حالت کلی روش های مبتنی بر گرایان سریع تر همگرا می شوند و می توانند جواب ها را با دقت بیشتری در مقایسه با روش های احتمالاتی در انجام دادن وظیفه جستجوی محلی بدست آورند. با این حال برای اجرای موثر این روش ها متغیرها و توابع هزینه تولید شده نیاز به پیوسته بودن دارند. علاوه بر این یک نقطه شروع خوب برای انجام موفقیت آمیز این روش ها واجب است. در بسیاری از مسائل بهینه سازی، نیاز است تا منطقه ممنوعه، محدودیت های جانبی و توابع هزینه محدب در نظر گرفته شوند. بنابر این مسائل بهینه سازی غیر محدب نمی توانند با استفاده از روش های ریاضی مرسوم حل شوند اگر چه برنامه ریزی پویا یا برنامه ریزی غیر خطی مختلط اعداد صحیح و شکل اصلاح شده آن ها بعضی سهولت را در حل مسائل غیر محدب ارائه می کنند، این روش ها در حالت کلی نیاز به تلاش محاسباتی مهم دارند.

مدت هاست که مطالعاتی بر روی الگوریتم های مختلفی که بتوانند از نقاط بهینه محلی بگریزند آغاز شده است. به عنوان یک جایگزین برای روش های متداول ریاضی تکنیک های بهینه سازی فراابتکاری استفاده شده اند تا راه حل های بهینه سراسری یا نزدیک به آن را بدست آورند. به علت قابلیت کاوش آن ها و پیدا کردن نواحی محتمل در فضای جستجو در زمان مناسب این روش ها برای جستجوی سراسری کاملاً مناسب می باشند، علاوه بر این نیاز به توابع هزینه و متغیرهای پیوسته استفاده شده برای روش های بهینه سازی ریاضی را کاهش می دهند. اگر چه این روش ها تقریبی هستند به عنوان مثال راه حل آن ها خوب می باشد اما لزوماً بهینه نمی باشد، آن ها نیاز به مشتقات تابع هدف و قیود ندارند و قوانین انتقال احتمالاتی را به جای قوانین قطعی به کار می گیرند.

از جمله رایج ترین این الگوریتم ها می توان به الگوریتم های جستجوی ممنوعه^۱، تبرید شبیه سازی شده^۲، حرکت پرندگان^۳، کرم شب تاب^۴، ژنتیک^۵، اجتماع مورچگان^۶، جستجوی هارمونی^۷، هارمونی^۷، جستجوی ذرات باردار^۸ و اشاره کرد [۳].

همچنین الگوریتم های مختلفی با استفاده از ترکیب الگوریتم های فراابتکاری شکل گرفته اند که می توان به الگوریتم های ترکیبی اجتماع مورچگان و حرکت پرندگان، جستجوی هارمونی و حرکت پرندگان، اجتماع مورچگان و حرکت پرندگان و جستجوی هارمونی نیز اشاره کرد.

۱-۲-۱ دسته بندی مسائل بهینه سازی سازه

بطور کلی مسائل بهینه یابی سازه ها در چهار رده مختلف دسته بندی می شوند [۴]. که عبارتند از :

۱. بهینه یابی سطح مقطع^۹

۲. بهینه یابی شکل یا هندسه سازه^{۱۰}

۳. بهینه یابی توپولوژی سازه^{۱۱}

۴. بهینه یابی پیکربندی^{۱۲} سازه مشتمل بر ۳ مورد قبل

بهینه سازی سطح مقطع به معنای بهینه کردن ضخامت و سطح مقطع اعضا می باشد.

¹ .Tabu Search

² .Simulated Annealing

³ .Particle Swarm

⁴ .Firefly

⁵ .Genetic Algorithm (GA)

⁶ .Ant Colony Optimization

⁷ .Harmony Search

⁸ .Charge System Search

⁹ .Sizing Optimization

¹⁰ . Geometrical (Shape) Optimization

¹¹ . Topological Optimization

¹² . Configuration

در بهینه‌یابی هندسی آنچه که بهینه می‌گردد هندسه سازه به عبارت دیگر محل قرارگیری نقاط گرهی سازه می‌باشد. باید توجه داشت در این حالت نحوه‌ی اتصال اعضای سازه‌ای (توپولوژی) تغییر نمی‌کند [۵].

در بهینه‌سازی توپولوژی محیط‌های پیوسته، هدف پیدا کردن تعداد مناسب حفره‌ها و محل آن‌ها می‌باشد [۶].

در بهینه‌یابی پیکربندی، با ترکیبی از موارد فوق که در برگیرنده هر سه مورد توپولوژی، مقطع و هندسه است، مواجه شده و بهینه‌یابی آن مد نظر قرار می‌گیرد.

۱-۳ اهداف تحقیق

روش‌های بهینه‌سازی ابزار و تکنیک‌هایی جهت مینیمم یا ماکزیمم کردن توابع هدف ما می‌باشند با توجه به اینکه به دنبال کم کردن وزن و هزینه سازه در عین برآوردن محدودیت‌های طراحی می‌باشیم بنابراین روش‌های بهینه‌سازی ابزار مناسبی می‌باشند. الگوریتم‌های فراابتکاری مجموعه الگوریتم‌هایی برای حل مسائل بهینه‌یابی هستند که بصورت تصادفی اما هدفمند و ساده در فضای جواب مسئله به دنبال جواب بهینه مطلق حرکت می‌نمایند. این روش‌های حرکت معمولاً از طبیعت الهام گرفته می‌شوند؛ زیرا برخی از پدیده‌های طبیعی با وجود تصادفی بودن به طرز جالبی دارای حرکتی به سمت حالت‌های نزدیک به حالت بهینه هستند. به دلیل اهمیت بالای مسائل بهینه‌یابی در علوم مهندسی و استفاده از الگوریتم‌های فرا ابتکاری در حل این مسائل، در این پایان نامه با بهره‌گیری از روش‌های نوین الگوریتم ژنتیک، اجتماع مورچگان، جستجوی هارمونی و جستجوی ذرات باردار شیوه جدیدی جهت بهینه‌یابی سطح مقطع سازه‌های خرپایی، بصورت همزمان، ارائه شده است تا بر اساس آن، فضای طراحی مساله به صورت مناسبی مورد کاوش قرار گیرد و جواب‌های نزدیک به جواب بهینه حاصل گردد. همچنین در این پایان نامه به بررسی کیفیت جواب حاصله و

سرعت همگرایی هر یک از الگوریتم ها و روش پیشنهادی در راه رسیدن به جواب بهینه پرداخته شده است و نتایج حاصل با نتایج مراجع معتبر دیگر مقایسه شده است.

با ارائه برنامه‌ی کامپیوتری نوشته شده، کاربران می‌توانند به طراحی بهینه وزنی سازه‌های بزرگ مقیاس بپردازند. این برنامه با در نظر گرفتن سایر محدودیت‌های طراحی از قبیل قیود تنش، جابه‌جایی و کمانش نوشته شده است.

۴-۱ ساختار پایان نامه

این پایان‌نامه شامل شش فصل می‌باشد که به تفکیک موضوع به شرح ذیل می‌باشد:

۱- در فصل اول به بیان مقدمه‌ای از علم بهینه‌سازی و همچنین خلاصه‌ای از اهداف و انگیزه ارائه این پایان‌نامه پرداخته شده است.

۲- در فصل دوم به تشریح عملکرد الگوریتم‌های فرا ابتکاری ژنتیک، اجتماع مورچگان، جستجوی هارمونی، جستجوی ذرات باردار و پارامترهای مرتبط با آن‌ها، برای حل مسائل بهینه‌سازی پرداخته شده است.

۳- در فصل سوم به مبانی بهینه‌یابی سطح مقطع خرپاها بر اساس الگوریتم‌های فراابتکاری و همچنین بررسی فرمولبندی رایج در این زمینه پرداخته شده است. برای این منظور تابع هدف و قیود حاکم بر مسئله در بهینه‌یابی سطح مقطع خرپاها مورد توجه قرار گرفته است.

۴- در فصل چهارم روش پیشنهادی و راهنمای استفاده از برنامه بهینه‌سازی سطح مقطع سازه تشریح شده است.

۵- در فصل پنجم کارآیی روش پیشنهادی با ذکر چندین مثال مورد بررسی قرار گرفته و نتایج حاصل با دیگر الگوریتم‌های مطرح در مراجع معتبر مقایسه شده است.

۶- در فصل ششم نتایج حاصل و پیشنهاداتی جهت تحقیقات بعدی ارائه شده است و در انتها فهرست مراجع ذکر گردیده است.

فصل دوم

الگوریتم های فرااستگاری

۲-۱ مقدمه

در این فصل ابتدا الگوریتم های فرا ابتکاری مورد بحث قرار گرفته و ویژگی ها و پارامتر های آنها بررسی می گردد و سپس مفاهیم چند الگوریتم فراابتکاری از قبیل الگوریتم ژنتیک، اجتماع مورچگان، جستجوی هارمونی و جستجوی ذرات باردار توضیح داده می شود.

۲-۲ الگوریتم های فرا ابتکاری

الگوریتم های ابتکاری و فرا ابتکاری به دلیل ضعف هایی که در الگوریتم های ریاضی وجود دارد، به وجود آمده اند. اگر بخواهیم چند نمونه از ضعف های روش های ریاضی را مطرح کنیم، می توان به موارد زیر اشاره کرد :

برای توابع با متغیرهای گسسته کارایی شان کاهش می یابد و در بعضی مواقع کارایی ندارند زیرا در روش های بهینه یابی ریاضی ما نیاز به اطلاعات مشتق تابع هدف داریم و وقتی تابع مورد نظر مشتق پذیر نباشد این الگوریتم ها نمی توانند مورد استفاده قرار گیرند. برای توابعی که مشتق گیری از آنها سخت می باشد قابل استفاده نیستند.

برای شروع عملیات بهینه یابی نیاز به مقدار اولیه دارند و برای توابعی که بیش از یک بهینه محلی دارند مقدار نقطه اولیه از اهمیت بالایی برخوردار خواهد بود زیرا مقدار اشتباه برای شروع موجب می شود این روش ها به تله افتاده و جواب نهایی، جواب بهینه مطلق برای مسئله مورد نظر نباشد.

به طور کلی این روش ها برای توابعی که ناشناخته می باشند و ما اطلاعات زیادی از آن ها در دست نداریم، خیلی کارآمد نمی باشند و از آنجا که در دنیای طبیعی اطراف ما، چنین مسائلی زیاد وجود

دارد، محققین به فکر ابداع روش های جدیدی افتاده اند. از جمله این روش ها می توان به الگوریتم های ابتکاری اشاره کرد. این الگوریتم ها برای یافتن یک جواب مناسب برای یک مسئله بهینه یابی از روش سعی و خطا استفاده می کنند تا در یک مدت زمان قابل قبول یک جواب مناسب برای مسئله مورد نظر بیابند. منظور از ابتکاری در این جا همان یافتن یا جستجو با استفاده از سعی و خطا می باشد. اگرچه این الگوریتم ها ممکن است جواب های بهتری نسبت به جواب هایی که برای مسئله موردنظر حدس زده شده است بیابند ولی هیچ تضمینی برای یافتن بهترین جواب یا جواب بهینه با استفاده از این الگوریتم ها وجود ندارد. به طور کلی، الگوریتم های ابتکاری همانند الگوریتم های ریاضی، به عنوان الگوریتم بهینه یاب محلی محسوب می گردند زیرا فقط در یک زیر فضا از کل فضای جستجو متمرکز شده و به بهینه یابی می پردازند. مزیت این الگوریتم ها نسبت به روش های خطی و غیرخطی ریاضی در این است که نیازی به اطلاعات مشتق تابع هدف ندارند. ضعف اساسی این الگوریتم ها در این می باشد که کاربر می بایست زیر فضای جستجو را برای الگوریتم انتخاب کند و اگر بهینه مطلق در این زیر فضا قرار نداشته باشد الگوریتم به تله خواهد افتاد و جواب نهایی، جواب بهینه نخواهد بود. از این رو این الگوریتم ها نیز برای مسائلی از دنیای طبیعی اطراف ما، که هیچ شناختی از آنها وجود ندارد خیلی کارآمد نیستند اما در مواقعی که بخواهیم در یک زیر فضای خاص از کل فضای جستجو بهترین جواب را در مدت زمان کم بیابیم، این الگوریتم ها بسیار مفید می باشند.

به دلایل ذکر شده در بالا، در سال های اخیر، محققین به اجبار به فکر ابداع الگوریتم های جدیدتر با کارایی بالاتر از الگوریتم های ابتکاری افتاده اند. این الگوریتم های جدید که الگوریتم های فرا ابتکاری نامیده می شوند همانطور که از نامشان مشخص است، نوع پیشرفته الگوریتم های ابتکاری می باشند. اگرچه این الگوریتم ها نیز همانند الگوریتم های ابتکاری از روش های سعی و خطا برای

یافتن جواب استفاده می کنند، اما در لغت فرا اکتشاف به معنی یافتن جواب با استفاده از روش ها و تکنیک های قوی تر و سطح بالاتر می باشد. در سال های اخیر، کلمه فرا ابتکاری ارجاع داده شده است به تمام الگوریتم های نوین سطح بالا، شامل الگوریتم های تکاملی^۱ که از جمله ی آن ها می توان به الگوریتم ژنتیک، جستجوی ممنوعه، اجتماع مورچگان، اجتماع پرندگان، الگوریتم زنبور عسل^۲، الگوریتم کرم شب تاب، الگوریتم جستجوی هارمونی و... اشاره کرد [۳]. همانطور که در بالا توضیح داده شد، الگوریتم های فرا ابتکاری همانند الگوریتم های ابتکاری از روش سعی و خطا برای انجام عملیات جستجو استفاده می کنند. اما نکته مهم هوشمند بودن این سعی و خطاست. در واقع مبتکرین این روش سعی می کنند با استفاده از روش های هوشمند سعی و خطا، این الگوریتم ها را به سمت یافتن جواب بهینه هدایت کنند. هر چه پارامترهای مورد استفاده در این الگوریتم ها هوشمندتر باشد احتمال به تله افتادن این الگوریتم ها کاهش یافته و احتمال آن که جواب نهایی به جواب بهینه نزدیک تر باشد افزایش خواهد یافت.

در تمام الگوریتم های فرا ابتکاری نوین دو مولفه بسیار مهم وجود دارد که این الگوریتم ها را شکل می دهند:

- تنوع^۳
- تشدید^۴

چنین اصطلاحات علمی از الگوریتم جستجوی ممنوعه گرفته شده است [۷]. برای این که یک الگوریتم فرا ابتکاری بتواند مناسب و موثر عمل نماید، می بایست بتواند به اکتشاف کل فضای جستجو بپردازد تا بتواند یک دامنه وسیع و متنوع از جواب ها که مستعد، بودن جواب بهینه هستند را تولید

^۱ Evolutionary Algorithms (EA)

^۲ Bee Algorithm (BA)

^۳ Diversification

^۴ Intensification

کند و هم زمان می بایست جستجوی خود را در همسایگی جواب های مستعد افزایش دهد. در واقع الگوریتم های فرا ابتکاری با استفاده از مولفه تنوع خود، سعی می کنند به طور مناسب و موثر کل فضای جستجو را مورد بررسی قرار دهند تا بتوانند یک مجموعه متنوعی از جواب های مستعد را بیابند. همچنین مولفه تشدید، الگوریتم های فرا ابتکاری را مجبور می کند که از حل های مستعد بدست آمده در جستجو های قبلی خود برای استخراج جواب بهینه نهایی استفاده نماید و جستجوی خود را در همسایگی آن جواب ها افزایش دهد. اصولا الگوریتم های فرا ابتکاری تشدید را از طریق نخبه گرایی یا استفاده از یک حافظه انجام می دهند [۷-۸].

آنچه که در الگوریتم های فرا ابتکاری از اهمیت بالایی برخوردار است، ایجاد یک تعادل مناسب بین این دو مولفه ظاهرا متضاد می باشد منظور از ایجاد تعادل مناسب و ترکیب کردن این دو پارامتر، در واقع این می باشد که الگوریتم بتواند همزمان هم جستجوی کلی و هم جستجوی محلی را در فضای جستجوی مورد نظر انجام دهد، و این عمل می بایست در هر تکرار الگوریتم، صورت گیرد. با این تفاوت که در تکرارهای اولیه شدت جستجوی کلی و در تکرارهای نهایی شدت جستجوی محلی می بایست بیشتر باشد.

وقتی ما با یک مسئله بهینه یابی در دنیای طبیعی اطرافمان برخورد می کنیم، اصولا شناخت خیلی زیادی از خصوصیات و رفتار آن نخواهیم داشت. در نتیجه ما یک فضای جستجویی خواهیم داشت که هیچ دید و حسی نسبت به آن، در ما وجود ندارد. همانند آن می ماند که بخواهیم در یک سرزمین وسیع و ناشناخته به دنبال عمیق ترین دره آن سرزمین بگردیم و عمق آن دره را محاسبه کنیم. اولین چیزی که به ذهن ما می رسد آن است که به تمام نقاط این سرزمین سر زده و عمیق ترین دره آن را بیابیم. اما آن چیزی که اهمیت دارد آن است که ما بتوانیم یک مدیریت مناسبی برای جستجوی خود داشته باشیم و به صورت کاملا تصادفی به جستجو پردازیم. زیرا همواره زمان از اهمیت بالایی برخوردار است. اگر ما بخواهیم به صورت تصادفی به اکتشاف خود پردازیم، مجبور خواهیم بود زمان بسیار زیادی را صرف این کار نماییم. مخصوصا در مسائل بهینه یابی مهندسی، که

زمان از اهمیت بسیار بالاتری برخوردار است و همواره محققین تلاش می کنند تا در کمترین زمان به بهینه ترین جواب برای یک مسئله بهینه یابی دست یابند. از طرفی دیگر، اگر ما به تمام نقاط آن سرزمین سرزنیم، در نتیجه نمی توانیم اطمینان حاصل نماییم که دره ی کشف شده، عمیق ترین دره آن سرزمین خواهد بود. پس همان طور که مشاهده می نماییم، جستجوی مناسب کل یک فضا در حداقل زمان ممکن از اهمیت بسیار بالایی برخوردار می باشد. این همان کاریست که در الگوریتم های فرا ابتکاری جستجوی کلی نامیده می شود و می بایست بیشتر با کمک مولفه تنوع این الگوریتم ها انجام شود. اما نکته ای که در اینجا حائز اهمیت می باشد آن است که ما به صرف انجام یک عملیات تصادفی برای جستجوی کلی، نمی توانیم به مقصود خود دست یابیم. این بدین معنی می باشد که جستجوی کلی ای که الگوریتم های فرا ابتکاری انجام می دهند می بایست یک جستجوی هوشمند باشد، تا در حداقل زمان به بهینه ترین جواب دست یابند، نه آن که کاملاً تصادفی و بی هدف به جستجوی کلی بپردازند.

مولفه تنوع باید این اطمینان را در ما ایجاد کند که کل فضای حل به صورت مفید و موثر مورد بررسی قرار گرفته است. زیرا در این صورت است که می توان اطمینان حاصل کنیم که الگوریتم در تله قرار نگرفته است. اگر تنوع در الگوریتم های فرا ابتکاری خیلی قوی باشد، سبب می گردد الگوریتم همانند یک جستجوگر تصادفی عمل نماید و این باعث افزایش زمان می گردد زیرا سرعت همگرایی الگوریتم کاهش می یابد به همین خاطر است که می بایست دو مولفه تنوع و تشدید را با هم ترکیب کرد. در واقع وقتی ما تشدید را به تنوع اضافه می کنیم یعنی یک مدیریت زمان مناسبی را به وجود آورده ایم. زیرا تشدید، الگوریتم را مجبور می کند که از اطلاعات بدست آمده از مولفه تنوع (یعنی اطلاعات حاصل شده از جستجوی کلی) استفاده بهتری کند و با مدیریت مناسب تری به جستجو بپردازد. اما از طرفی اگر تنوع در الگوریتم فرا ابتکاری خیلی ضعیف باشد، الگوریتم نمی تواند به شکلی مناسب کل فضای حل را بررسی کند و اصطلاحاً به تله خواهد افتاد، زیرا الگوریتم با استفاده از مولفه تشدید، برای انجام عملیات جستجوی خود، در جایی متمرکز می شود که جواب بهینه مطلق در آن

ناحیه ممکن است وجود نداشته باشد و در نتیجه الگوریتم به جواب های بی معنی دست خواهد یافت. در مورد مولفه تشدید نیز همین قضایا وجود دارد. در واقع یک تشدید مناسب سبب می گردد الگوریتم به صورت مناسبی به جستجوی محلی پرداخته و با استفاده مناسب از تاریخچه جستجو و تجارب قبلی، جواب های بهینه تری را استخراج کند. در واقع هدف این مولفه آن است که در مواقع لازم، با محدود کردن جستجوی تصادفی، سرعت همگرایی را افزایش دهد و الگوریتم را مجبور کند با متمرکز شدن در مکان های مناسب تر، عملیات جستجوی بیشتری را در آن مکان ها انجام دهد. این مولفه تشدید است که سبب می شود الگوریتم با مدیریت مناسب تری به جستجو پردازد و باعث هوشمندتر شدن الگوریتم می گردد. اما اگر تشدید در الگوریتم های فرا ابتکاری خیلی قوی باشد، منجر به تولید جواب های حل نامناسبی خواهد شد که به اصطلاح می گویند، الگوریتم همگرایی نارس داشته است و به تله افتاده است. زیرا الگوریتم نتوانسته است کل فضای حل را مورد بررسی قرار دهد و برعکس اگر تشدید ضعیف باشد سرعت همگرایی کاهش خواهد یافت.

به همین دلیل، همواره نیازمند ایجاد تعادل بهینه بین این دو مولفه می باشیم و ایجاد چنین تعادلی خود نیازمند یک پروسه بهینه یابی می باشد. از این رو، آنالیز حساسیت پارامترهایی که وظیفه ایجاد تنوع و تشدید را دارند، در الگوریتم های فرا ابتکاری از اهمیت بالایی برخوردار است.

۲-۳ مفاهیم الگوریتم ژنتیک

۲-۳-۱ تاریخچه الگوریتم ژنتیک

ایده اصلی الگوریتم های تکاملی در سال ۱۹۶۰ توسط ریچنبرگ^۱ مطرح شد که تحقیق وی در مورد استراتژیهای تکاملی^۲ بود [۹]. بعدها نظریه وی توسط محققان زیادی مورد بررسی قرار گرفت تا منجر به طرح الگوریتم ژنتیک شد. الگوریتم های ژنتیک در حقیقت، روش جستجوی کامپیوتری بر

^۱ . Rechenberg

^۲ . Evolutionary Strategies

پایه الگوریتم های بهینه سازی و بر اساس ساختار ژن^۱ ها و کروموزوم^۲ ها است که توسط جان هولند^۳ در سال ۱۹۷۵ در دانشگاه میشیگان مطرح شد و پس از وی توسط جمعی از دانشجویانش مثل گلدبرگ^۴ و آنن آربور^۵ توسعه یافت. در سال ۱۹۹۲ جان کوزا^۶ از الگوریتم ژنتیک برای حل و بهینه سازی مسائل مهندسی پیشرفته استفاده کرد و توانست برای اولین بار روند الگوریتم ژنتیک را به زبان کامپیوتر درآورد و برای آن یک زبان برنامه نویسی ابداع کند که به این روش برنامه نویسی، برنامه نویسی ژنتیک^۷ می گویند. نرم افزاری که توسط وی ابداع گردید به نرم افزار LISP مشهور است که هم اکنون نیز کاربرد فراوانی در حل و بهینه سازی مسائل مهندسی دارد [۱۰]. از آن پس تا به امروز کتاب های متعددی توسط افرادی چون گلدبرگ، دب^۸ و جنکینز^۹ به رشته تحریر در آمده است و پس از آن بسیاری از محققین مقالات متنوعی درباره بسط دامنه کاربرد GA، بهبود نتایج و افزایش سرعت محاسبات ارائه نموده اند. این الگوریتم امروزه به دلیل کاربرد فراوان، از جایگاه ویژه ای در میان روش های بهینه سازی برخوردار است.

در یک نگاه کلی اصطلاحاتی که از مباحث زیست شناسی به مبحث الگوریتم ژنتیک وارد شده است را می توان به شرح زیر دسته بندی نمود :

۱- تولید مثل^{۱۰} که به تولید کروموزوم های جدید با ویژگی های والدین خود، منجر می شود.

۲- جهش^{۱۱} که سبب ایجاد تغییرات ناگهانی در DNA می شود.

۳ - مناسب بودن که برای یک موجود زنده میزان شایستگی آن در ادامه حیات است.

^۱ . Gene

^۲ . Chromosome

^۳ . John Holland

^۴ . Goldberg

^۵ . Ann Arbor

^۶ . John Koza

^۷ . Genetic Programming (GP)

^۸ . Deb

^۹ . Jenkins

^{۱۰} . Cross Over

^{۱۱} . Mutation

۲-۳-۲ بهینه یابی با الگوریتم های ژنتیک

الگوریتم ژنتیک با الهام از نظریه د/روین^۱ درباره حیات بهترین ها شکل گرفته است که بر اساس اساس آن بهترین ها حق بقا دارند، بنابراین می توان گفت [۹].

« GA بر اساس اصل " ادامه حیات بهترین ها " و " تکثیر نوع برتر " پی ریزی شده است. »
این روش هوشمند بطور موفقیت آمیزی طرح بهینه عمومی^۲ را بدون در نظر گرفتن فرضیات محدود کننده ای از قبیل پیوسته بودن فضای جستجو و یا وجود مشتقات پیدا می کند. فرآیند GA بر روی متغیرهای طراحی رمز شده که رشته ای از کاراکترها با طول محدود هستند، انجام می گیرد. مزیت کار با متغیرهای کد شده در این است که اصولاً کدها قابلیت تبدیل فضای پیوسته به فضای گسسته را دارند [۱۱].

در بهینه یابی از طریق GA چهار مرحله اساسی انجام می پذیرد [۱۲]:

۱. تولید تصادفی مجموعه ای از طرح ها، که بدان جمعیت اولیه گفته می شود. هر عضو این جمعیت یک کروموزوم می باشد که بصورت کد^۳ بوده و رشته^۴ نامیده می شود. هر رشته متناسب با تعداد متغیرهای طراحی، به چندین زیر رشته^۵ تقسیم می شود. زیر رشته مجموعه ای از بیت^۶ ها می باشد که بصورت زنجیر در کنار هم چیده شده اند. هر بیت هم هم ارز با یک ژن در الفبای ژنتیک می باشد. تعداد بیت های هر زیررشته به نحوی تعیین می شوند تا بتوان کلیه اطلاعات متغیر طراحی را مابین حدود بالا و پایین در مرحله رمز گشایی بدست آورد.

^۱ . Darwin

^۲ . Global Optimum

^۳ . Code

^۴ . String

^۵ . Substring

^۶ . Bite

۲. با رمزگشایی، مقدار متغیرهای طراحی در هر رشته ارزیابی و مقادیر تابع هدف تعیین می گردد.

۳. مسأله بهینه یابی مقید با تعریف تابع پنالتی^۱ و ترکیب آن با تابع هدف و ایجاد تابع هدف اصلاح شده^۲، به یک مسأله بهینه یابی نامقید تبدیل می شود. در ادامه با تعریف تابع شایستگی، به هر کروموزوم، یک مقدار شایستگی اختصاص داده می شود. سپس با الهام از قوانین تکامل طبیعی و روش هایی که در ادامه ارائه می شود، بهترین کروموزوم ها بر مبنای میزان شایستگی انتخاب می شوند.

۴. در این مرحله، جمعیت جدید بر مبنای عملگرهای GA که در ادامه به شرح تفصیلی آن پرداخته شده است، تشکیل می شود. پس از آن جمعیت فرزندان^۳ برای نسل^۴ بعد به کار گرفته می شود. این مراحل تا ارضای شرط خاتمه و یا به تعداد نسل های پیش بینی شده برای کسب بهینه عمومی تکرار می شود.

اساساً برای پیاده سازی مراحل GA به شکلی صحیح می بایست سه مفهوم زیر مشخص شود:

۱- تعریف تابع هدف به شکلی صحیح

۲- تعریف فضای ژنتیک بر اساس قیود مسئله

۳- تعریف و پیاده سازی صحیح عملگرهای GA

در فرآیندهای به ظاهر ساده فوق، نکات عمیق و مهمی نهفته است که در بخش های بعد به طور اجمالی بحث شده است.

^۱ . Penalty Function

^۲ . Modified Objective Function

^۳ . Offspring

^۴ . Generation

۲-۳-۱-۲ کد گذاری کروموزوم ها

در حل مسائل بهینه‌سازی روش‌های متنوعی جهت نمایش پارامترها و اطلاعات مسئله (کروموزوم‌ها) به کار برده شده است که انتخاب هرکدام از این روش‌ها باید با توجه به نوع مساله و فضای جستجوی مورد نیاز برای حل مساله بهینه‌سازی صورت پذیرد. به طور مثال می‌توان سیستم کدگذاری به صورت رشته‌ای، آرایه^۱، درخت^۲ یا لیست^۳، مقداری یا ارزشی^۴، جایگشتی^۵ صورت گیرد. در این روش‌های کدگذاری می‌توان از اعداد یا حروف استفاده کرد. معمول‌ترین روش در کدگذاری عددی، کدگذاری باینری (دو دویی) می‌باشد [۱۳]. شیوه‌های مختلفی برای کدگذاری وجود دارد [۱۰]، که در اینجا به شرح کد گذاری دودویی پرداخته شده است.

۲-۳-۱-۲-۱ کد گذاری باینری^۶

از آن جایی که این نوع کدگذاری، متداولترین نوع کدگذاری می‌باشد و همچنین از این روش در این پایان‌نامه استفاده شده به شرح این روش می‌پردازیم. در این روش کدگذاری، هر کروموزوم شامل یک رشته از بیت‌های ۰ و ۱ می‌باشد. شکل (۲-۱).

Chromosome A	101101100011
Chromosome B	010011001100

شکل (۲-۱) - نمایش کروموزوم ها به صورت باینری

^۱.Array

^۲.Tree

^۳.List

^۴.value

^۵.Permutation

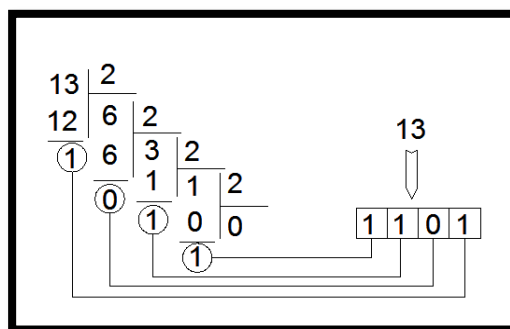
^۶. Binary

نخستین گام برای کد کردن پارامترهای تصمیم‌گیری مساله، تعیین طول رشته‌ی آن پارامتر خاص می‌باشد، که این طول به شمار مقدارهای گسسته آن متغیر خاص و سامانه کدگذاری بستگی دارد و از رابطه‌ی زیر محاسبه می‌گردد:

$$2^n = N \quad (1-2)$$

که در آن N تعداد متغیرهای طراحی و n طول مورد نیاز رشته است.

به عنوان مثال برای یک مساله خاص که دارای ۱۶ پارامتر است، زیر رشته تعریف شده بایستی ۴ بیتی باشد که برای بدست آوردن رشته باینری معادل یک پارامتر مشخص، در ابتدا کد مورد نظر، و در ادامه، خارج قسمت را به طور متوالی بر عدد دو تقسیم می‌نماییم تا رشته معادل تعیین گردد. شکل (۲-۲).



شکل (۲-۲) - نحوه‌ی بدست آوردن اعداد در مبنای ۲

اما اگر تعداد متغیرهای طراحی توانی از دو نباشد. طول رشته مورد نظر کوچکترین عدد صحیحی می‌باشد که در رابطه ذیل صادق است.

$$2^n > N \quad (2-2)$$

بنابراین با ادغام رابطه (۱-۲) با رابطه (۲-۲) رابطه کلی (۳-۲) به دست می‌آید طول رشته مورد نظر

کوچکترین عدد صحیحی است که در رابطه ذیل صادق باشد.

$$2^n \geq N \quad (3-2)$$

رابطه (۳-۲) را می‌توان به صورت ذیل نیز نشان داد. اگر تعداد متغیرهای طراحی توانی از دو نباشد از رابطه (۴-۲) در غیر این صورت از رابطه (۵-۲) استفاده می‌شود.

$$n = [\log_2 N] + 1 \quad (4-2)$$

$$n = \log_2 N \quad (5-2)$$

در معادله (۴-۲) علامت [] به معنی جزء صحیح می‌باشد.

طبق رابطه (۱-۲) با توجه به اینکه یک رشته n بیتی، تعداد N عدد از ۰ تا $2^n - 1$ را کد می‌نماید؛ لذا مقدار عددی هر زیررشته A در هنگام رمزگشایی^۱ برابر است با [۱۲].

$$A = \sum_{k=0}^{k=n-1} 2^k * S_k \quad (6-2)$$

هر بیت S می‌تواند اعداد ۰ و ۱ را اختیار نماید و به صورت رشته زیر نمایش داده می‌شود:

$$S_{n-1} \dots S_3 S_2 S_1 S_0 \quad (7-2)$$

اگرچه سیستم دودویی گسترده‌ترین شیوه کدگذاری است اما دارای مشکلاتی نیز می‌باشد که بدان اشاره می‌گردد، نخست آنکه تغییرات کوچک در پارامترها می‌تواند تغییرات بزرگ در رشته را بوجود آورد.

به عنوان نمونه دو عدد (۱۰۰۰۰۰) و (۰۱۱۱۱۱) به لحاظ مقدار، تنها دارای اختلاف واحد هستند. ولی در حالت کد شده بسیار متفاوت می‌باشند، این اختلاف در هنگام اعمال عملگرهای ژنتیک معمول مشکل آفرین خواهد بود، به بیان دیگر، یک تغییر کوچک در فضای کد شده، مشابه آن تغییر در مقدار حقیقی نیست. مشکل دیگر سیستم‌های دودویی، هنگامی به وجود خواهد آمد که

^۱.Decoding

تعداد پارامترهای مساله دقیقاً با توان‌های عدد ۲ برابر نباشد؛ به طور مثال در یک مساله خاص (برای مثال عدد ۶)، تعداد پارامترهای تولیدی عددی بین ۴ و ۸ می باشد. در این صورت به احتمال زیاد، در طی مراحل مختلف اجرای الگوریتم، مقداری تولید می شود که معادل تعداد پارامتر مربوطه نیست. برای رفع این مشکل برای رمزگشایی به طریق ذیل عمل می کنیم [۱۳].

$$A = X^L + \left[\frac{X^U - X^L}{2^n - 1} * \sum_{k=0}^{k=n-1} 2^K * S_k \right] \quad (۸-۲)$$

که X^L حد پایین متغیر طراحی، X^U حد بالای متغیر طراحی و n طول رشته مورد نظر می باشد.

۲-۳-۲ شایستگی هر کروموزوم

همان طور که پیشتر بیان شد، GA، مسئله بهینه یابی مقید را با تعریف تابع پناستی که وابسته به میزان نقض قیود است و ترکیب آن با تابع هدف و ایجاد تابع هدف اصلاح شده، به یک مسئله بهینه یابی نامقید تبدیل می نماید. در ادامه با کمک مقادیر تابع هدف اصلاح شده تابع شایستگی تعریف می گردد. سپس با توجه به تابع شایستگی، به هر کروموزوم یک مقدار شایستگی اختصاص می یابد. اما آنچه که در روند فوق حائز اهمیت است؛ چگونگی محاسبه تابع پناستی و تابع هدف اصلاح شده است تا با کمک این توابع، تابع شایستگی تعریف شود. بنابراین در ادامه به بررسی توابع فوق الذکر پرداخته می شود.

۲-۳-۲-۱ تابع پناستی

تابع پناستی یا جریمه، همان طور که از نامش مشخص است، مقداری است که به عنوان جریمه به تابع هدف اعمال می شود؛ این جریمه به دلیل نقض قیود می باشد. در واقع به واسطه تعریف تصادفی متغیر یا متغیرهای طراحی قیود مسئله نقض می شود، بنابراین کروموزوم ناقض قیود جریمه

می شود. طبیعی است که هر کروموزومی قیود مسئله را بیشتر نقض نماید، مقدار تابع پنالتی متناظر آن بیشتر است و بالطبع جریمه بیشتری به مقدار تابع هدف مربوط به این کروموزوم اعمال می شود.

محققین روابط متعددی را برای تابع پنالتی و چگونگی محاسبه مقدار جریمه ارائه نموده اند [۱۲]. در این پایان نامه از تابع پنالتی راجیو و کریشنامورتی^۱ [۱۴] که در ذیل آمده استفاده شده است.

$$f_{Penalty} = F(X) * K * C \quad (۹-۲)$$

که

$$C = \sum_{q=1}^Q \max[0, g_q(X)] \quad (۱۰-۲)$$

در روابط فوق X بردار متغیر طراحی، Q تعداد کل قیود حاکم بر مسئله، $f_{Penalty}$ تابع جریمه، $F(X)$ تابع هدف، $g_q(X)$ میزان نقض سازه در ارتباط با هر یک از قیود می باشند و ثابت K با توجه به مسئله در نظر گرفته می شود. در گام بعدی برای رسیدن به مقدار شایستگی هر کروموزوم می بایست تابع هدف اصلاح شده و در پی آن تابع شایستگی محاسبه شود؛ هر کدام از این توابع نیز بر اساس دیدگاه های مختلف به طرق گوناگون مطرح شده است که در ذیل به شرح اهم این دیدگاه ها بر حسب کاربردشان پرداخته شده است.

۲-۲-۳-۲ تابع شایستگی^۲

تابع شایستگی معمولاً براساس تابع هدف اصلاح شده تعریف می شود و بایستی مقدار آن مثبت باشد. تابع هدف اصلاح شده نیز عموماً براساس مقادیر تابع جریمه و تابع هدف تنظیم می شود.

^۱ S.Rajeev & C.S.Krishnamoorthy

^۲ . Fitness Function

محققین GA روابط متنوع و گوناگونی را برای تابع هدف اصلاح شده و تابع شایستگی ارائه داده اند که برخی از آن ها عبارتند از:

۱- گلد برگ در کتاب خود، مرجع [۱۵]، تابع هدف اصلاح شده را مطابق رابطه (۱۱-۲) محاسبه می نماید؛ اما برای تابع شایستگی پیشنهاد می کند که تابع هدف اصلاح شده هر کروموزوم از یک مقدار بزرگ کم شود تا مقدار شایستگی هر کروموزوم بصورت مثبت محاسبه گردد. وی این مقدار ثابت بزرگ را به عهده طراح می گذارد.

$$\phi(X) = F(X) + f_{Penalty} \quad (11-2)$$

۲- راجیو و کریشنا مورتی، تابع هدف اصلاح شده و تابع شایستگی را بشرح زیر پیشنهاد می نمایند [۱۶] و [۱۴]:

$$\phi(X) = F(X) + f_{Penalty} \quad (12-2)$$

$$R(X) = [\phi_{\max}(X) + \phi_{\min}(X)] - \phi(X) \quad (13-2)$$

در روابط فوق $\phi(X)$ تابع هدف اصلاح شده، $R(X)$ تابع شایستگی، $f_{Penalty}$ تابع جریمه، $F(X)$ تابع هدف، $\phi_{\max}(X)$ حداکثر مقادیر تابع هدف اصلاح شده در جمعیت حاضر، $\phi_{\min}(X)$ حداقل مقادیر تابع هدف اصلاح شده در جمعیت حاضر می باشند.

در هر صورت مطلب زیر در تمامی روابط و نظریات مربوط به تابع شایستگی حاکم است:

۱- در مسئله کمینه سازی، کروموزومی با کمترین مقدار تابع هدف اصلاح شده، بیشترین مقدار شایستگی را به خود اختصاص می دهد؛ لذا شانس بیشتری برای حضور در جمعیت بعدی دارد.

۲- مقادیر تابع شایستگی، همواره مقداری مثبت است.

در نهایت امر تابع شایستگی به ازای هر کروموزوم محاسبه و میزان شایستگی هر کروموزوم بدست

می آید. حال زمان آن فرا رسیده است، تا با کمک مقادیر تابع شایستگی بهترین ها برای نسل بعد برگزیده شوند. این مهم در فرآیند کلی به نام انتخاب به شرح زیر تبلور می یابد.

۲-۳-۲ انتخاب^۱

پس از برآزش کروموزوم‌های موجود در هر جمعیت نوبت به پدیده‌ی انتخاب کروموزوم‌های والد می‌رسد که براساس شایستگی یک کروموزوم منطبق است، هرچند شیوه‌های متعدد در انتخاب کروموزوم تا به امروز مطرح گشته است ولی، ویژگی‌های مشترک در تمامی آن‌ها، انطباق روش بر اصل انتخاب طبیعی داروین است، بر این اساس که جواب‌های بهتر (کروموزوم‌های با شایستگی بیشتر) برای مساله از شانس بیشتری برای زاد و ولد برخوردارند. عکس این مطلب نیز صادق است، یعنی کروموزوم‌هایی با مقدار شایستگی کمتر، احتمال کمتری برای انتقال در استخر تولید مثل خواهند داشت در تمامی فرایندهای انتخاب هرگاه کروموزومی منتخب گردد، بدان معنی است که این کروموزوم شایستگی حضور در استخر تولید مثل^۲ و یا حضور مستقیم در نسل بعدی را دارا است، به بیان ساده اینکه تعداد فرزندان که از هر فرد به وجود خواهد آمد در این مرحله تعیین می‌گردد. اما تعیین شیوه‌ها و تکنیک‌های انتخاب، بستگی به نوع مساله و شرایط حاکم بر آن دارد که هر کدام بنا به موقعیت و شرایط محیطی فضای مساله کارایی خاص خود را دارد. مهمترین و رایج‌ترین روش‌های انتخاب، عبارتند از:

۱. روش چرخ گردان^۳

۲. روش رتبه بندی^۴

۳. روش مسابقه^۱

^۱ . Selection(Reproduction)

^۲ . Mating Pool

^۳ . Roulette Wheel

^۴ . Rank

که این روش ها به تفصیل در مرجع [۱۲]، شرح داده شده است.

۲-۳-۲ مفهوم برگزیده (شایسته سالاری)

استراتژی نخبه گرایی یا مفهوم برگزیده، نخستین بار توسط فردی به نام دی جانگ^۱ در سال ۱۹۷۵ مطرح شد. بر اساس این روش بهترین کروموزوم ها (یک کروموزوم) بر مبنای شایستگی در جمعیت جدید بدون آنکه وارد استخر تولید مثل شود؛ کپی می شوند. این کار برای آن است تا بهترین کروموزوم نسل حاضر، در نسل بعدی حضور داشته باشند.

به کمک استراتژی فوق اطمینان حاصل می گردد که بهترین فرزند هر نسل نسبت به بهترین فرزند نسلهای گذشته دارای نمره شایستگی پایین تری نخواهد بود. این روش اغلب به صورت ترکیبی با سایر روش های انتخابی ارائه می شود. استفاده از مفهوم برگزیده باعث افزایش کارایی GA می گردد، زیرا مانع از گم شدن جواب های خوب به دست آمده می شود. استراتژی نخبه گرایی تأثیر بسزایی در همگرایی الگوریتم داشته و استفاده از آن در مقالات گوناگون، بررسی و توصیه شده است.

۲-۳-۳ عملگرهای الگوریتم ژنتیک

پس از مراحل انتخاب کروموزوم های والد بر طبق هرکدام از روش های اشاره شده در بالا، نوبت به تولید نسل بعدی می رسد، مجموعه فعالیت هایی که در حوضچه تولید مثل انجام می گردد تا نسل های بعدی تولید شده، هر یک بخشی از اطلاعات کروموزوم های مولد خود را دارا باشند توسط عملگرهای ژنتیک انجام می گیرد.

سپس طی فرآیند بکارگیری عملگرهای الگوریتم ژنتیک، رشته های بهتر و جدید تولید و تکثیر خواهند گردید و در نتیجه این ساختار الگوریتم سبب جستجوی بیشتر و بهتر در فضای طراحی

^۱ . Tournament

^۲ . Kenneth De Jong

می گردد. الگوریتم ژنتیک دارای عملگر های مختلفی می باشد، که به برخی از آن ها در این بخش اشاره می گردد.

معمولاً در یک الگوریتم ژنتیک ساده^۱ از دو عملگر اصلی زیر پس از فرآیند انتخاب استفاده می شود:

۱. ادغام

۲. جهش

۲-۳-۲-۵-۱ عملگر ادغام

در مرحله انتخاب، جمعیتی از بهترین کروموزوم ها شکل می گیرد. در فرآیند ادغام، هدف تولید رشته های بهتر و جدید (تولید فرزندان) می باشد. فرآیند ادغام، با هدف تعویض ویژگی طرح ها در میان اعضای جمعیت، به قصد بهبود شایستگی نسل بعد انجام می شود. این مشابه با انتقال صفات ژنتیکی در فرآیندهای زایش موجودات زنده است که توسط DNA و RNA صورت می پذیرد. پیوند سبب جستجوی بیشتر در فضای طراحی می گردد.

عملگر ادغام یک عملگر ترکیبی است که شامل سه مرحله است. در مرحله اول یک جفت رشته به صورت تصادفی انتخاب می شود. در مرحله دوم یک زیررشته، از کروموزوم ها به طور تصادفی در طول رشته انتخاب خواهد شد. بایستی توجه داشت که طول زیررشته در هر دو کروموزوم، ثابت باشد. سرانجام در سومین مرحله زیررشته منتخب در دو رشته والد جابه جا می شود.

روش های مختلف و متنوعی برای عمل ادغام وجود دارد. بر اساس نظریه دب^۲ نمی توان گفت که کدامیک از روش های ادغام بهتر است [۹]. در هر حال مشهورترین روش های ادغام بشرح زیر خلاصه می شوند:

۱- روش ادغام تک نقطه ای یا مکانی^۱

^۱ . Simple Genetic Algorithm

^۲ . Deb

۲- روش ادغام دو نقطه‌ای^۲

۳- روش ادغام چند نقطه‌ای^۳

۴- روش ادغام یکنواخت^۴

۵- روش ادغام متغیر

که این روش‌ها نیز به تفصیل در مرجع [۱۲]، شرح داده شده است.

۲-۳-۲-۵-۱-۱ نرخ ادغام

در انجام فرآیند ادغام با تعریف ثابتی به نام نرخ ادغام (پیوند)، شانس حضور کروموزوم در فرآیند ادغام بررسی می‌شود. نرخ ادغام بیانگر احتمال ادغام است که آن را با p_c نشان می‌دهند و مقدار آن بین 0 و 1 است. با فرض احتمال ادغام p_c می‌توان گفت که $p_c * 100$ درصد از رشته‌های موجود در استخر تولید مثل در عملیات ادغام به کار رفته‌اند و $(1-p_c) * 100$ درصد از جمعیت باقی می‌مانند. نحوه انجام کار به این صورت است که با انتخاب مقدار نرخ پیوند، برای هر کروموزوم یک عدد تصادفی در محدوده صفر تا یک تولید می‌شود. اگر عدد تصادفی تولید شده از نرخ ادغام کمتر باشد، این کروموزوم برای آمیزش با کروموزوم بعدی که شرایط فوق را دارد، انتخاب می‌شود. انتخاب نرخ ادغام مناسب نیز، اغلب مشکل‌ساز است. اگر خیلی زیاد باشد، باعث می‌شود که فرصت تطابق در کروموزوم از دست برود و اگر خیلی کم باشد، تعداد فرزندان تولید شده کافی نخواهد بود [۱۲].

۲-۳-۲-۵-۲ جهش

از دیگر مراحل رایج در عملیات GA که تکامل جمعیت برای نسل بعد را به دنبال خواهد

¹ . Single-Sight Cross Over

² . Two-Point Cross Over

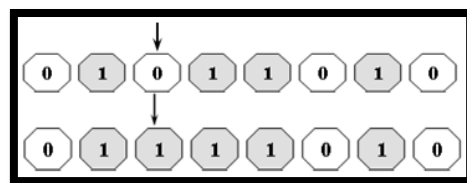
³ . Multi-Point Cross Over

⁴ . Uniform Cross Over

داشت، استفاده از اپراتور جهش است. پس از عمل ادغام رشته ها، نوبت به عمل جهش یا موتاسیون می رسد. نقش جهش در الگوریتم ژنتیک، بازگرداندن مواد ژنتیکی گم شده و یا پیدا نشده به داخل جمعیت است. بدین سان با استعانت از اپراتور جهش از همگرایی زودرس الگوریتم ژنتیک و کسب جواب های بهینه محلی جلوگیری می شود. به عبارت دیگر در یک کلام می توان گفت اپراتور جهش، سبب جستجوی بهتر فضای طراحی می شود.

۲-۳-۲-۵-۱ عمل جهش در کدگذاری دودویی

عمل جهش یک بیت، شامل تبدیل عدد صفر (0) به یک (1) و بالعکس است که بر اساس یک احتمال کوچک مثل P_m به صورت بیت به بیت صورت می گیرد. عمل جهش به این ترتیب است که یک عدد تصادفی بین صفر (0) تا یک (1) تولید می شود؛ اگر عدد تولید شده کوچکتر از P_m باشد مقدار خروجی را برابر درست^۱ و گرنه برابر غلط^۲ در نظر گرفته می شود. اگر برای هر بیت مقدار خروجی درست باشد، بیت تغییر می کند و گرنه بیت بدون تغییر باقی خواهد ماند. بیت های یک رشته بصورت مستقل جهش می یابند، به این معنا که جهش یک بیت بر روی احتمال جهش سایر بیت ها تأثیر نمی گذارد. این عمل در یک الگوریتم ژنتیک ساده به منزله ی یک عملگر ثانویه و به منظور حفظ اطلاعاتی که در حال از دست رفتن است، تلقی می گردد. به کمک این عملگر می توان امید داشت که کروموزوم های خوب که در مراحل انتخاب و یا تکثیر حذف شده اند، دوباره احیا شوند.



شکل (۲-۳) - عمل جهش را در بیت ۳ نشان می دهد

۲-۳-۲-۵-۲ نرخ جهش

^۱ . True

^۲ . False

(P_m) یا همان احتمال جهش در کروموزوم‌ها معمولاً در حدود ۰.۰۰۱ تا ۰.۰۱ در نظر گرفته می‌شود [۱۴]، با این احتمال اندک، این عملگر تضمین می‌نماید که احتمال رخداد هر نقطه در فضای مساله هیچ‌گاه صفر نگردد. در بعضی از مراجع این نرخ برای یک جمعیت با اندازه ی ۲۰ تا ۲۰۰ مقداری بین ۰.۰۰۵ تا ۰.۰۵ نیز توصیه شده است [۹]. به هر تقدیر باید توجه داشت، نرخ های جهش پایین سبب همگرایی سریع و عدم تحرک در فضای جستجو می گردد، از سوی دیگر نرخ های جهش بالا سبب می گردد که عملیات GA به سمت روند جستجوی تصادفی میل نماید که گاه موجب از بین رفتن کروموزوم های نخبه‌ی تولید شده در نسل‌های ابتدایی و میانی می‌شود که جهت رفع این نقیصه، بهره جویی از الگوریتم‌های نخبه‌گرا توصیه می‌گردد.

نرخ جهش را می‌توان ثابت در نظر گرفت و یا پیشنهاد می‌شود آن را متناسب با تعداد نسل‌های تولید شده قرار داد. در این صورت دو حالت وجود دارد که به شرح زیر است:

۱- در این حالت نرخ جهش بصورت افزایشی می باشد. یعنی در ابتدا جهش با احتمال کمی اعمال شده و در خلال عملیات GA، متناسب با نسبت شماره نسل به تعداد کل نسل ها، افزایش می یابد. یعنی:

$$P_m = P_m^{Min} + (P_m^{Max} - P_m^{Min}) \frac{t}{T} \quad (14-2)$$

۲- در این حالت نرخ جهش بصورت کاهشی می باشد. یعنی در ابتدا جهش با بیشترین احتمال اعمال می شود و در خلال فرآیند GA، رشد نزولی خود را متناسب با نسبت شماره نسل به تعداد کل نسل ها، درپیش می گیرد. یعنی:

$$P_m = P_m^{Max} - (P_m^{Max} - P_m^{Min}) \frac{t}{T} \quad (15-2)$$

در روابط (۱۴-۲) و (۱۵-۲)، P_m مقدار نرخ جهش در هر نسل، P_m^{Max} حداکثر مقدار نرخ جهش در

فرآیند GA، P_m^{Min} حداقل مقدار نرخ جهش در فرآیند GA، t شماره نسل حاضر و T تعداد کل نسل سازی می باشند.

۲-۳-۶ دستور خروج از الگوریتم

برخی از شرط های متداول توقف عملیات GA عبارتند از [۱۷]:

۱. رسیدن به جواب^۱: این ساده ترین فکر برای همگرایی است. به این معنی که اگر تصور

شد الگوریتم به مقدار کروموزوم مناسب رسیده است، الگوریتم متوقف شود.

۲. عدم پیشرفت^۲: یعنی الگوریتم GA پس از n بار تکرار با همان کروموزوم های قبلی

ادامه پیدا کند؛ در این صورت الگوریتم متوقف شود. در این حالت ممکن است الگوریتم

در یک نقطه کمینه نسبی، به دام افتاده باشد.

۳. روش آماری^۳: اگر انحراف از معیار^۴ مقادیر شایستگی جمعیت به یک مقدار مشخصی

رسید، الگوریتم متوقف می شود. این بدان مفهوم است که مقادیر شایستگی جمعیت با

یکدیگر تفاوت چندانی نداشته باشند.

۴. تعداد تکرارها^۵: اگر با هیچ کدام از موارد فوق جواب نداد، شرط توقف براساس تعداد

تکرار بنا می شود.

در هر صورت باید اذعان داشت که ممکن است GA نیز، همانند دیگر روش های بهینه سازی، به جواب

بهینه نرسد و یا در نقطه کمینه نسبی گرفتار شود. در هر صورت اگر الگوریتم به یک حل بهینه همگرا

نشود، می بایست پارامترهای آن را تغییر داده و مجدداً الگوریتم اجرا شود. این پارامترها نقش بسزایی

در رسیدن به جواب بهینه دارند.

¹ . Correct Answer

² . No Improvement

³ . Statistics

⁴ . Standard Deviation

⁵ . Number Of Iterations

۲-۴ الگوریتم اجتماع مورچگان

۲-۴-۱ مقدمه ای بر الگوریتم اجتماع مورچگان

مورچه ها، در روی زمین از حدود ۱۰۰ میلیون سال پیش زندگی می کرده اند و در حال حاضر جمعیتی در حدود 10^{16} مورچه در دنیا وجود دارد. بعد ها تخمین زده شد که مجموع وزن مورچه ها در حدود مجموع وزن انسان های روی زمین می باشد. اغلب این حشرات، موجوداتی اجتماعی می باشند و در کلونی هایی (به یک مجموعه موجودات که در کنار هم یک زندگی اجتماعی دارند کلونی، جرگه یا توده گفته می شود) به اندازه ای از ۳۰ تا میلیون ها فرد زندگی می کنند. رفتارهای پیچیده کلونی مورچگان انسان ها را تشویق کرده است که مطالعات متعددی در خصوص رفتارهای اجتماعی و فردی آنها داشته باشند. رفتارهای گروهی از مورچگان در ارتباط با جستجو برای غذا، نظم کارگران، سازماندهی گورستان، مواظبت از بچه ها و ساختن لانه مورد مطالعه قرار گرفته است. مارایس یکی از اولین افرادی بود که کلونی موریانه ها را مورد بررسی قرار داد. وی در کتابش تحت عنوان "روح مورچه" (که در سال ۱۹۳۷ بعد از مرگش چاپ شد)، آزمایشات انجام شده و مشاهدات خود را در ارتباط با رفتار اجتماعی موریانه ها شرح داد.

در ادامه بیولوژیست فرانسوی به نام گریس، مکانیسم ارتباطی موریانه ها را مورد بررسی قرار داد. او نشان داد که یک فرمی از ارتباطات غیر مستقیم، که نام آن را استیگمرجی گذاشت، بین موریانه ها وجود دارد. در سال های نزدیک تر، دنبرگ و همکارانش یک نمونه از استیگمرجی را بررسی کردند و نام آن را ارتباطات فرومونی گذاشتند [۱۸]. بعد از این مطالعه ها، اولین الگوریتم که رفتار مورچه ها را در جستجوی غذا نمایش می داد، توسط دوریگو ارائه شد و در ادامه توسط سایر محققین توسعه داده شد [۲۰-۱۹]. در حالیکه اغلب تحقیقات در ارتباط با توسعه مدل ها و الگوریتم هایی از رفتار

جستجوی غذای مورچگان انجام می شد، مدل های دیگری نیز در ارتباط با سایر رفتارهای مورچگان، شامل مدیریت کارگران، سازماندهی گورستان و... انجام می شد.

۲-۴-۲ متاهیوریستیک بهینه سازی کلونی مورچگان

یکی از اولین رفتارهایی که توسط محققین مورد مطالعه قرار گرفت، توانایی مورچه ها در یافتن کوتاه ترین مسیر بین یک منبع غذا و لانه آنها بود. بر اساس این مطالعات، اولین الگوریتم از رفتار مورچه ها در جستجوی غذا توسط دوریگو ارائه گردید. بعد از آن، الگوریتم های دیگری نیز توسعه داده شد. به طور کلی الگوریتم هایی که به عنوان نتیجه ای از مطالعه رفتار مورچه ها در جستجو برای غذا توسعه داده شد، به عنوان نمونه هایی از متاهیوریستیک بهینه سازی کلونی مورچگان (ACO-MH) شناخته می شود.

۲-۴-۲-۱ رفتار در جستجوی غذای مورچگان

چگونه مورچه ها می توانند کوتاه ترین مسیر بین منبع غذا و لانه خود را بدون هیچ گونه مکانیسم هدایتی، دیداری، مرکزی یا فعال پیدا کنند؟ مطالعات رفتارهای بعضی از گونه های واقعی مورچه ها نشان داده است که یک الگوی تصادفی و شانسی در حرکت مورچه ها برای یافتن غذا وجود دارد. به محض اینکه یک منبع غذا در محلی در حوالی لانه مورچه ها قرار داده می شود، الگوی تصادفی به مرور سازماندهی بیشتری یافته و در نهایت منجر به رسیدن به یک مسیر مشابه توسط مورچه ها، خواهد انجامید و به صورت جادویی دیده می شود که مورچه ها کوتاه ترین مسیر را پیدا کرده اند. این رفتار عجیب نتیجه تاثیر مورچه هایی که منبع غذا را پیدا کرده اند بر سایر مورچه ها می باشد. این تاثیر ممکن است در بعضی از گونه های مورچه ها به صورت مستقیم و بعضی گونه های دیگر به صورت غیر مستقیم باشد. نشان داده شده است که در اغلب گونه های مورچه این تاثیر یا به کارگیری مورچه های دیگر به صورت غیر مستقیم اتفاق می افتد.

مورچه هایی که غذا را یافته اند با سایر مورچه ها از طریق یک مکانیسم ارتباطی به نام اثر فرومون ارتباط برقرار می نمایند. زمانی که یک مورچه یک منبع غذا را پیدا می کند و مقداری غذا به خانه می برد در مسیر حرکت خود ماده شیمیایی به نام فرومون منتشر می نماید. بقیه مورچه ها که به دنبال غذا می گردند، در هنگام انتخاب مسیر به غلظت فرومون مسیر توجه می نمایند. مسیرهای با غلظت فرومون بیشتر، احتمال بیشتری برای انتخاب دارند. هر چقدر مورچه های بیشتری از یک مسیر عبور نمایند، غلظت فرومون آن مسیر افزایش یافته و در نتیجه، مورچه های بیشتری نسبت به آن مسیر علاقه مند می شوند.

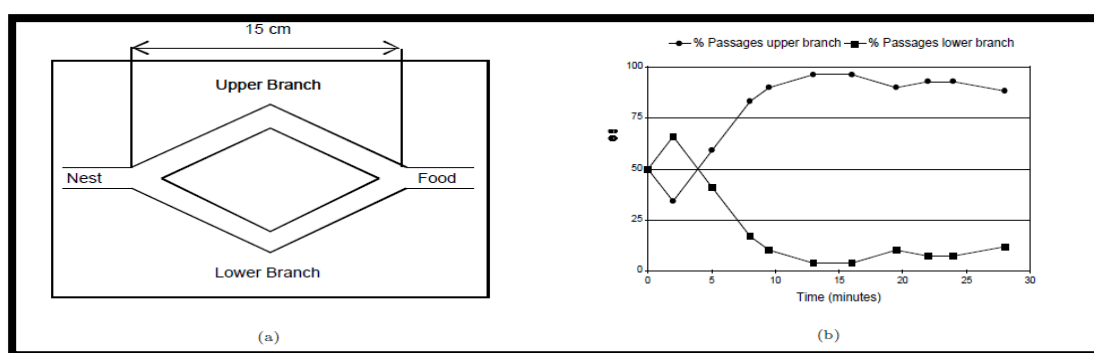
۲-۲-۴-۲ آزمایشات پل

دنبه‌رگ و همکارا نش رفتار گونه ای مورچه آرژانتینی را به منظور توسعه یک مدل برای شرح رفتار مورچه ها بررسی کردند. در آزمایشگاه آنها، همان طور که در شکل (۲-۴) نمایش داده شده است، لانه با یک پل دارای دو شاخه از منبع غذا جدا شده است. در ابتدا، هردو شاخه از هر فرومونی پاک بودند. بعد از یک دوره زمانی محدود، اغلب مورچه ها از یک مسیر استفاده می نمودند در حالی که هر دو مسیر با همدیگر برابر بودند. انتخاب یکی از مسیرها، ناشی از نوسانات تصادفی در انتخاب مسیر، منجر به غلظت بالا در یکی از مسیرها شده بود. همان طوری که در قسمت b این شکل نمایش داده شده است، درصد انتخاب مسیرها در ابتدای آزمایش برابر بوده است. با ادامه آزمایش، نوساناتی در درصد انتخاب مسیر به وجود آمده است. به عبارت دیگر، درصد انتخاب یک مسیر کم و زیاد شده است اما با ادامه آزمایش، پس از سپری شدن مقدار محدودی از زمان، یک مسیر نسبت به مسیر دیگر ترجیح داده شده است.

بر اساس این آزمایش که به آزمایش پل باینری نیز معروف است، یک مدل قراردادی ساده برای مشخص کردن خصوصیات انتخاب مسیر توسعه داده شد. به همین منظور، فرض گردید که مورچه ها مقدار مشابهی از فرومون را منتشر می نمایند و آن فرومون ها تبخیر نمی شوند. فرض کنید $n_A(t)$ و

$n_B(t)$ تعداد مورچه ها در مسیر A و B را در مرحله زمانی t نمایش می دهند که با سرعت ثابت v در حال حرکت می باشند، پاستیل و همکارانش نشان دادند که احتمال این که مورچه بعدی مسیر A را در مرحله زمانی $t+1$ انتخاب نماید، از طریق رابطه زیر قابل محاسبه است:

$$p_A(t+1) = \frac{(c + n_A(t))^\alpha}{(c + n_A(t))^\alpha + (c + n_B(t))^\alpha} = 1 - p_B(t+1) \quad (۲-۱۶)$$



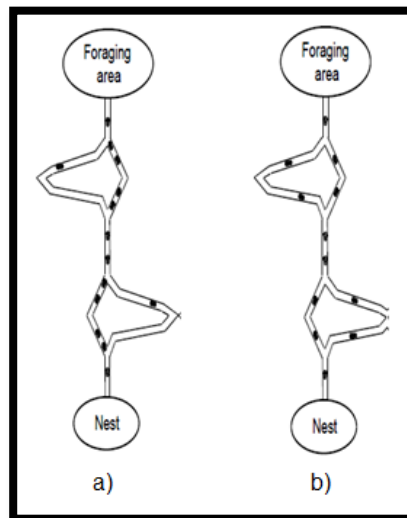
شکل ۲-۴ - آزمایش پل دمبرگ

که در آن C درجه جذابیت یک شاخه بررسی شده را مشخص نموده که از تقسیم سرعت (v) بر طول مسیر طی شده توسط مورچه بدست می آید و α یک درجه توجه، به فرومون منتشر شده است. مقادیر بالای α ، منجر به احتمال بیشتری برای انتخاب مسیری که مقدار غلظت فرومون بیشتری دارد، می گردد. مقادیر بالای C موجب می گردد که انتشار فرومون بیشتری برای انتخاب یک مسیر به صورت غیر تصادفی، لازم باشد. به طور مثال اگر C صفر باشد و α بزرگ باشد، مسیر با غلظت بیشتر فرومون، حتی اگر به مقدار کمی بیشتر باشد، با احتمال بیشتری انتخاب می گردد.

تصمیم گیری بر اساس معادله (۲-۱۶) به این صورت انجام می گردد که یک عدد تصادفی با توزیع یکنواخت به صورت $U(0,1)$ تولید می گردد، اگر این عدد کوچکتر از $p_A(t+1)$ باشد، آنگاه مسیر A انتخاب شده و در غیر این صورت مسیر B انتخاب می گردد.

گوس و همکارانش آزمایش پل باینری را توسعه دادند. درآزمایش آنها، همان طوری که در شکل

(۵-۲) دیده می شود یکی از پل ها طولانی تر از دیگری می باشد. در ابتدا مسیر ها به طور تصادفی انتخاب شده و تقریباً در هر دو مسیر تعداد مورچه ها برابر می باشند (شکل (۵-۲) قسمت a) بعد از مدتی تعداد بیشتری از مورچه ها، مسیر کوتاهتر را انتخاب می نمایند و این عمل ادامه می یابد (شکل (۵-۲) قسمت b) .

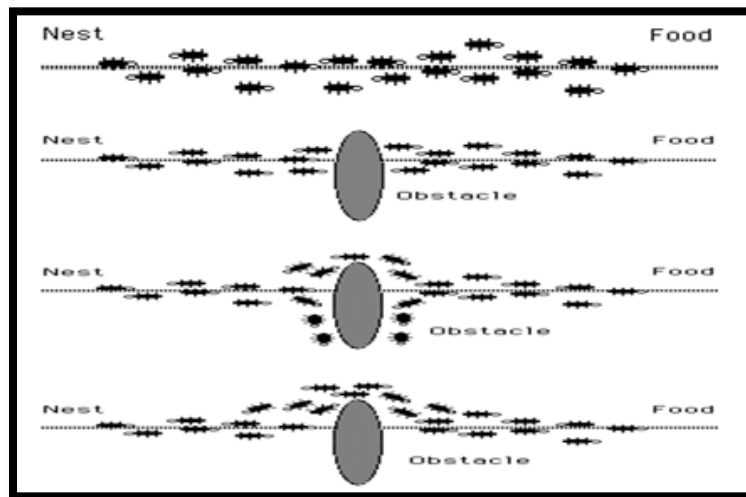


شکل ۵-۲- آزمایش پل با طول های نامساوی

علت انتخاب مسیر کوتاه تر این است که، مورچه هایی که مسیر کوتاهتر را انتخاب می نمایند، زودتر از مورچه هایی که مسیر طولانی تر را انتخاب کرده اند، به لانه برمی گردند. در نتیجه، در صورتی که از هر مسیر یک مورچه با هم عبور کرده باشند، مورچه مسیر کوتاه تر، زودتر برگشته و در نتیجه فرومون مسیر کوتاه تر بیشتر است. آنها متوجه شدند که احتمال انتخاب مسیر کوتاه تر، با میزان نسبت طول دو مسیر رابطه دارد. شکل (۶-۲)، نمایی دیگر از رفتار مورچه ها را در یافتن کوتاهترین مسیر بین لانه و منبع غذا را نمایش می دهد.

با وجود این که کلونی مورچه ها رفتار پیچیده ای دارند، یک مورچه تکی از یک رفتار خیلی ساده تبعیت می کند یک مورچه را می توان به عنوان یک عامل که به محرک ها پاسخ می دهد، در نظر گرفت مورچه غلظت های فرومون در مسیر ها را مشاهده می کند و یک اقدام را بر اساس محرک

فرومون به وجود می آورد. یک مورچه مصنوعی این رفتار ساده از مورچه های واقعی را در قالب یک الگوریتم، مدل می نماید[۱۸].



شکل ۲-۶- رفتار مورچه در یافتن کوتاه ترین مسیر بین لانه و منبع غذا

۲-۴-۳ از مورچه های طبیعی تا روش فرا کاوشی کلونی مورچگان

الگوریتم های ACO از رفتار واقعی جامعه مورچه ها الهام گرفته شده است. اساس الگوریتم های ACO همکاری و مشارکت عوامل محاسباتی (مورچه ها) و ارتباط آن ها از طریق اثر فرومون مصنوعی که به آن تاثیر گذاری نیز گفته می شود، می باشد. مورچه ها برای حرکت از یک نقطه به نقطه دیگر با توجه به دو دسته اطلاعات حرکت خود را انجام می دهند:

اطلاعات کاوشی^۱: که نشان دهنده مطلوبیت نسبی مسیرها با توجه به تابع هدف مسئله می باشد. در مسئله ای که هدف کمینه کردن هزینه ها می باشد اطلاعات کاوشی، عکس هزینه مسیر پیموده شده تعریف می شود. این اطلاعات توسط مورچه ها در طول الگوریتم تغییر نمی یابد. اطلاعات اثر فرومون مصنوعی^۲: میزان فرومون که نشان از جذابیت مسیر برای مورچه های مصنوعی

^۱.Heuristic information

^۲.(Artificial)Pheromon trail information

دارد، بستگی به میزان عبور سایر مورچه ها در تکرار های قبلی از این مسیرها دارد. این مقادیر در طول اجرای الگوریتم بسته به کیفیت جواب های پیدا شده توسط مورچه ها تغییر می یابد.

۴-۲-۴-۲ شبا هتها و تفاوت های بین مورچه های واقعی و مصنوعی

جامعه مورچه های واقعی و مصنوعی دارای شبا هتهای زیر هستند [۲۱]:

- ۱- استفاده از یک جامعه که با اثر گذاری متقابل و همکاری وظیفه خود را انجام می دهند.
 - ۲- مورچه های مصنوعی و طبیعی محیط اطراف خود را با ارتباط تاثیر گذاری^۱ بر اساس فرومون تغییر می دهند در مورد مورچه های مصنوعی، اثر فرومون مصنوعی اطلاعات عددی است که به صورت محلی در دسترس می باشد.
 - ۳- جستجوی نزدیک ترین مسیر از لانه تا منبع غذایی
 - ۴- مورچه های مصنوعی و واقعی در حرکت از یک مرحله به مرحله دیگر از اطلاعات فرومون استفاده می کنند.
- از آنجایی که این مشخصات به تنهایی اجازه توسعه الگوریتم های مفید برای مسائل پیچیده را نمی دهد بنابراین مورچه های مصنوعی در یک جهان مجزا زندگی می کنند و مشخصات دیگری نیز دارند که در مورچه های واقعی یافت نمی شود [۲۱]:
- ۱- مورچه های مصنوعی در حرکت از یک مرحله به مرحله دیگر علاوه بر اطلاعات فرومون از اطلاعات کاوشی نیز استفاده می کنند.
 - ۲- آنها دارای حافظه ای هستند که مسیر پیموده شده توسط مورچه را ذخیره می کند.

¹ stigmergy.

۳- میزان فرومون گذاشته شده به کیفیت جواب یافت شده توسط مورچه بستگی دارد.

۴- یکی از تفاوت‌های اساسی، زمان گذاشتن فرومون می باشد، مورچه های مصنوعی معمولاً بعد از ساختن جواب کامل اقدام به گذاشتن فرومون می کنند.

۵- مکانیسم تبخیر برای جلوگیری از سکون الگوریتم در یک جواب بهینه نسبی می باشد. مکانیسم آن در الگوریتم ACO متفاوت از طبیعت می باشد. تبخیر فرومون به مورچه ها اجازه می دهد تا گذشته خود را فراموش کنند و در یک فضای جدید به جستجو ادامه دهند.

۶- به منظور بهبود عملکرد سیستم، الگوریتم های ACO دارای توانایی های اضافی هستند. توان جلوگیری^۱ و لیست کاندیداها^۲ نمونه هایی از این قابلیت ها است که برای بهبود عملکرد الگوریتم به خدمت گرفته می شوند.

۲-۴-۵ استیگمرجی و فرومون مصنوعی

به طور کلی، استیگمرجی ساختاری از رابطه های بین حیوانات است. عنوان استیگمرجی در ابتدا توسط گریس (۱۹۵۹) به صورت ارتباطات درونی بین مورچه ها که از طریق تغییراتی در محیط به وجود می آید، تعریف گردید. او این موضوع را از روی مطالعه رفتار موریانه ها در ساختن لانه کشف کرد. کلمه استیگمرجی از دو کلمه یونانی زیر گرفته شده است:

• Stigma، به معنای علامت

• Ergon، به معنای کار یا شغل

مورچه ها سیگنال هایی را مشاهده می کنند که یک اقدام یا پاسخ خاص را بر می انگیزد. این اقدام خود محرکی برای سایر مورچه ها می باشد. دو شکل از استیگمرجی را می توان به صورت (۱) با

^۱Lookahead

^۲Candidate list

استفاده از تغییرات فیزیکی محیط و^(۲) مبتنی بر علامت تعریف کرد. استیگمرجی با استفاده از تغییرات فیزیکی، به ایجاد ارتباط از طریق تغییراتی در مشخصه های فیزیکی محیط گفته می شود. برای مثال، مورچه ها در فعالیت های ساخت لانه و تمیز کردن لانه، از این سیستم ارتباطی استفاده می نمایند. استیگمرجی مبتنی بر علامت، به صورت یک مکانیسم علامت دهی از طریق انتشار مواد شیمیایی توسط مورچه ها می باشد. رفتار مورچه ها در هنگام جستجو برای غذا، مثالی از این سیستم ارتباطی می باشد.

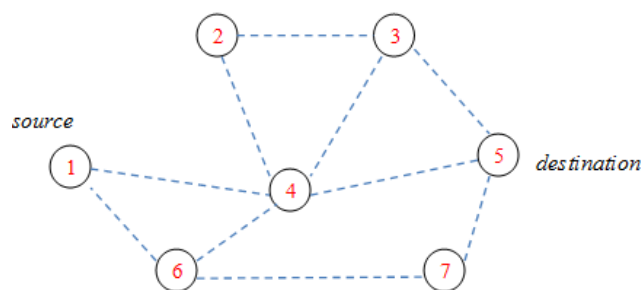
الگوریتم های مورچه با الهام از رفتار های واقعی مورچه ها به صورت سیستم هایی مبتنی بر جمعیت طراحی شده اند. مشارکت بین مورچه ها در یک الگوریتم مورچه، از مکانیسم ارتباطی مورچه های واقعی الهام گرفته شده است

همان طور که قبلا شرح داده شد، مورچه ها این قابلیت را دارند که کوتاه ترین مسیر بین لانه و منبع غذا را پیدا کنند. به محض این که مورچه ها از منبع غذا به سمت لانه حرکت می کنند، مقداری از فرومون توسط مورچه ها منتشر می گردد. مورچه های بعدی، مسیرها را به صورت احتمالی بر پایه مقدار فرومون موجود در مسیر ها انتخاب می نمایند. مسیر دارای غلظت فرومون بیشتر، با احتمال بیشتری انتخاب می گردد. بعضی از گونه های مورچه ها وجود دارند که مقدار فرومون را متناسب با کیفیت غذا منتشر می نمایند.

در طول زمان، مسیر های کوتاه تر به علت اینکه مورچه ها سریع تر به لانه بر می گردند، مقدار غلظت فرومون بیشتری خواهند داشت. فرومون در طول زمان تبخیر می گردد، و به همین علت غلظت فرومون موجود در مسیر طولانی تر با سرعت بیشتری نسبت به مسیر کوتاه تر، کاهش می یابد. فرومون مصنوعی تقلیدی از مشخصه های فرومون واقعی است و نشان دهنده شهرت یک راه حل در یک مساله بهینه سازی می باشد.

۶-۲-۴-۲ شکل ساده ای از بهینه سازی کلونی مورچگان

اولین الگوریتم مورچه که توسعه داده شد، سیستم مورچه بود واز آن موقع توسعه های متعددی در سیستم مورچه بوجود آمده است. این الگوریتم ها دارای فرایندهای تصمیم گیری نسبتا پیچیده تری نسبت به آنچه در بخش ۲-۲-۴-۲ ارائه گردید، می باشند. در این قسمت، ساختار ساده ای از الگوریتم کلونی مورچگان شرح داده می شود. مساله عمومی، یافتن کوتاه ترین مسیر بین دو راس در یک گراف $G=(V,E)$ که V مجموعه راس ها و E ماتریسی است که ارتباطات بین راس ها را بیان می کند را در نظر می گیریم. این گراف $n_G = |V|$ راس دارد. طول l^k ، که توسط مورچه k ام ساخته شده است، به صورت فاصله مبدا تا مقصد تعریف می گردد. یک گراف نمونه همراه با یک مسیر انتخابی در شکل (۷-۲) نمایش داده شده است. طول مسیر انتخابی برابر ۲ است. در این حالت، یک غلظت فرومون τ_{ij} ، به هر یال (i,j) از گراف، اختصاص داده می شود.



شکل ۷-۲- مساله کوتاه ترین مسیر در یک گراف

در بهینه سازی کلونی مورچگان ساده (SACO)، در ابتدا به هر یال یک مقدار تصادفی به عنوان فرومون اولیه، $\tau_{ij}(0)$ ، تخصیص داده می شود. در اصل، در ابتدای الگوریتم به علت نبود غلظت فرومون یا مقدار خیلی کم آن، مورچه ها به طور تصادفی مسیر ها را انتخاب می نمایند. تعدادی از مورچه ها، $(k=1,2,\dots,n_k)$ ، در راس مبدا قرار داده می شود. در هر تکرار SACO، هر مورچه به صورت پله به پله یک مسیر (راه حل) را بین مبدا و مقصد می سازد. در هر راس، هر مورچه فرآیند

تصمیم گیری را برای تعیین یال بعدی مسیر خود به کار می برد. اگر مورچه k در راس i قرار داشته باشد، راس بعدی $j \in N_i^k$ را براساس یک تابع احتمال گذر به شکل زیر تعیین می نماید:

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha}{\sum_{j \in N_i^k} \tau_{ij}^\alpha(t)} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \quad (17-2)$$

که در آن N_i^k مجموعه یال های قابل انتخاب برای مورچه k در راس i با توجه به محدودیت های مسئله می باشد. اگر برای هر راس i و مورچه k ، $N_i^k = \emptyset$ ، آنگاه راس پیشین راس i در نظر گرفته می شود (در صورتی که مورچه به مسیر بسته برسد، بر می گردد). این مساله منجر به ایجاد یک حلقه می گردد، ولی مشکلی را ایجاد نمی کند زیرا پس از رسیدن به مقصد، در هنگام معرفی مسیر بین مبدا و مقصد، حلقه ها حذف می گردند. در معادله بالا α یک مقدار ثابت مثبت برای شدت بخشیدن به اثر غلظت فرومون است. مقدار بزرگ α اهمیت غلظت فرومون را افزایش داده و تاثیر رفتار های تصادفی را کمتر می کند و منجر به همگرایی سریع به سمت یک بهینه محلی می گردد.

بعد از آنکه تمامی مورچه ها یک مسیر کامل بین مبدا تا مقصد را ساختند و حلقه های احتمالی حذف شدند، هر مورچه رد پای خود را به سمت راس مبدا دنبال می نماید و مقداری فرومون، در هر یال (i,j) روی مسیر طی شده مطابق با قاعده زیر منتشر می نماید:

$$\Delta \tau_{ij}^k(t) \propto \frac{1}{L^k(t)} \quad (18-2)$$

$L^k(t)$ معرف طول مسیر ساخته شده توسط مورچه k ام در مرحله زمانی t می باشد. در نتیجه، میزان فرومون هر مسیر مطابق با قاعده زیر به هنگام می گردد:

$$\tau_{ij}(t+1) = \tau_{ij} + \sum_{k=1}^{n_k} \Delta \tau_{ij}^k(t) \quad (19-2)$$

که در آن n_k تعداد مورچه ها می باشد. با استفاده از معادله (۲-۱۸)، مقداری چگالی جدید روی هر یال تولید می گردد و این مقدار بستگی به کیفیت مسیر هایی دارد که توسط مورچه ها با استفاده از آن یال ساخته شده است. به عبارت دیگر، هر مورچه مقداری فرومون در هر یال متناسب با کیفیت مسیرش به جا می گذارد. بعد از یک تکرار، میزان فرومون موجود در یک یال با توجه به مورچه های عبوری از آن یال تغییر می یابد. مقدار چگالی اضافه شده با مقدار قبلی چگالی فرومون آن مسیر طبق معادله (۲-۱۹) به هنگام می گردد.

در SACO مقدار کیفیت هر راه حل با یک تابع ساده به صورت عکس طول مسیر طی شده بیان می گردد. البته معیارهای دیگری نیز می توان تعریف کرد. اگر $x^k(t)$ نشان دهنده یک راه حل در مرحله زمانی t باشد، آنگاه $f(x^k(t))$ کیفیت آن راه حل را بیان می کند. اگر مقدار فرومون انتشاری رابطه عکس با طول مسیر داشته باشد (شبیه معادله (۲-۱۸)) در آن صورت، مقدار بیشتر $f(x^k(t))$ (که نشان دهنده بد تر بودن راه حل ساخته شده است) موجب می گردد که مقدار کم تری فرومون در آن مسیر منتشر گردد. بنابر این، یک مسیر طولانی موجب می گردد که تمامی اجزای آن دارای جذابیت کمتری شوند (فرومون کمتری در آن ها منتشر گردد).

در SACO قواعد توقف متعددی می توان استفاده کرد. برای مثال چند قاعده در زیر آمده است:

- توقف زمانی که تعداد ماکزیممی از تکرار ها انجام گرفت.
- توقف زمانی که یک راه حل قابل قبول ساخته شد. $(f(x^k(t)) \leq \varepsilon)$
- توقف زمانی که تمامی مورچه ها (یا اغلب آن ها) از یک مسیر مشابه پیروی کنند.

در آزمایشات اولیه مساله پل باینری نشان داده شد که مورچه ها به سرعت به یک راه حل همگرا می شوند و پس از اینکه مقدار کمی زمان سپری گردید، مسیرهای جایگزین را انتخاب می کنند. برای اجبار مورچه ها در کاوش بیشتر و جلوگیری از همگرایی زودرس، اجازه داده شد که مقدار شدت فرومون در یک مسیر در هر تکرار از الگوریتم، قبل از ساخت مسیر، دچار تبخیر شود. در

این حالت، مقداری از شدت فرومون هر مسیر کاهش (تبخیر) پیدا می کند. در این حالت نحوه به هنگام سازی مقدار شدت فرومون طبق معادله زیر انجام می گردد:

$$\tau_{ij}^k = (1-\rho)\tau_{ij} \quad (20-2)$$

که در آن $\rho \in [0,1]$ یک مقدار ثابت است که نرخ تبخیر فرومون در هر مرحله را نمایش می دهد. این تبخیر منجر به فراموش شدن درصدی از وقایع گذشته (تصمیمات گرفته شده در گذشته یا به عبارت دیگر مسیرهای انتخاب شده در گذشته) می گردد و به عبارت دیگر ρ یک پارامتر کنترلی تاریخچه جستجو است. برای مقادیر بالای ρ ، نرخ بالای تبخیر، توجه به گذشته کم شده و جستجو بیشتر وضعیت تصادفی و شانسی به خود می گیرد. در صورتی که $\rho = 1$ باشد، جستجو کاملاً تصادفی و شانسی خواهد بود.

در این جا یادآوری این نکته ضروری است که ارائه راه حل نتیجه رفتار مشارکتی ای است که از رفتار ساده مورچه های تکی ناشی می گردد. هر مورچه یال بعدی مسیر خود را بر اساس اطلاعات دریافتی از سایر مورچه ها، در قالب انتشار فرومون، انتخاب می کند. از طرفی دیگر، این نکته هم ضروری است که هر مورچه از اطلاعات دریافتی از حوزه محلی خود، برای تصمیم گیری استفاده می نماید. دوریگو و دی کارو در آزمایشات خود دریافتند که:

- SACO برای گراف های ساده خیلی خوب عمل می نماید و کوتاه ترین مسیر در اغلب مواقع انتخاب می گردد.
- برای گراف های بزرگ تر، انتخاب پارامتر ها باید با حساسیت صورت گیرد و در صورت استفاده از پارامتر های نامناسب، کیفیت الگوریتم نزول خواهد کرد.
- تبخیر در گراف های پیچیده نقش مهمی را ایفا می کند. اگر $\rho = 0$ و تبخیری وجود نداشته باشد، الگوریتم به سمت جواب خوبی همگرا نخواهد شد، اگر تبخیر فرومون زیاد باشد،

الگوریتم در اغلب مواقع برای برای مسائل پیچیده به سمت یک راه حل بهینه محلی همگرا می شود.

- برای مقادیر کم α ، الگوریتم به طور کلی به سمت کوتاه ترین مسیر همگرا می شود. برای مسایل پیچیده، مقدار بالای α منجر به همگرایی نامناسب الگوریتم می گردد.

بر اساس این مطالعات، اهمیت کاوش و تبعیت (تنوع و تشدید) روی SACO آشکار گشت. در هنگام طراحی الگوریتم، باید مکانیسمی به کار گرفته شود که اطمینان حاصل کند که مورچه ها به طور نابهنگام به سمت یک بهینه محلی نامناسب همگرا نشوند و کاوش های بیشتری را انجام دهند. کاوش، رفتار مورچه در یافتن مسیرهای جدید (از روی تصادف) و تبعیت، رفتار مورچه ها در پیروی از مسیرهای با غلظت فرومون بیشتر را بیان می کند. به عبارتی دیگر تنوع هر راه حل با تصادفی بودن و استفاده از قانون انتخاب مسیر توسط هر مورچه بیان گردیده و تشدید با تغییرات غلظت فرومون و میزان تبخیر اشاره می گردد.

الگوریتم SACO که در قسمت های قبلی مورد بحث قرار گرفت موفقیت های خوبی در یافتن کوتاه ترین مسیر به دست آورد. عملکرد این الگوریتم را می توان با استفاده از بعضی از تغییرات کوچک از قبیل اضافه کردن اطلاعات هیوریستیکی (کاوشی) برای تعیین احتمال انتخاب یک مسیر، حافظه با لیست ممنوع، قواعد مختلف بهنگام سازی فرومون با استفاده از اطلاعات محلی و سراسری، بهبود بخشید. در ادامه الگوریتم ACO با به کارگیری این تغییرات شرح داده خواهد شد.

۷-۲-۴-۲ سیستم مورچه

اولین الگوریتمی که توسط دوریگو توسعه داده شد به یک سیستم مورچه (AS) شبیه بود. AS توسعه ای را روی SACO با تغییر احتمال گذر، p_{ij}^k به منظور در نظر گرفتن

اطلاعات هیوریستیکی و اضافه نمودن یک حافظه شبیه به لیست ممنوع، به وجود می آورد.

در AS احتمال حرکت از راس i به راس j به صورت زیر بیان می گردد:

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t) \cdot \eta_{ij}^\beta(t)}{\sum_{u \in N_i^k} \tau_{iu}^\alpha(t) \cdot \eta_{iu}^\beta(t)} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \quad (21-2)$$

که در آن τ_{ij} شدت فرومون مسیر (i,j) و η_{ij} جذابیت حرکت (اطلاعات کاوشی) که توسط بعضی از هیوریستیک ها محاسبه می گردد می باشند.

تابع احتمال گذر (21-2) با تابع احتمال گذر (2-17) الگوریتم SACO در دو جنبه زیر متفاوت است:

- احتمال گذر مورد استفاده در AS توازنی را بین غلظت فرومون، τ_{ij} ، (به عبارت دیگر،

تاریخچه حرکت های موفق قبلی) و اطلاعات هیوریستیکی، η_{ij} ، (که جذابیت حرکت را بیان

می کند) برقرار می نماید. این امر منجر به توازن بین کاوش و تبعیت (تنوع و تشدید)

می گردد. بهترین توازن بین کاوش و تبعیت از طریق انتخاب مناسب پارامتر های α و β به

دست می آید. اگر $\alpha = 0$ باشد، اطلاعات فرومونی مورد استفاده قرار نمی گیرد و سابقه

گذشته ی جستجو فراموش می گردد و الگوریتم حالت جستجوی تصادفی را به خود

می گیرد. اگر $\beta = 0$ باشد، جذابیت حرکت ها صرفنظر می گردد و الگوریتم ساختار SACO

را به خود می گیرد، اطلاعات هیوریستیکی نحوه انتخاب مسیرها را دچار تغییراتی می نماید.

نحوه تعریف آن بسته به مساله مورد بررسی می باشد. برای مثال برای یک مساله حداقل

سازی مسیر این تابع به صورت زیر قابل تعریف است:

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (22-2)$$

که در آن d_{ij} فاصله (یا هزینه) بین راس i و j است.

- مجموعه N_i^k ، مجموعه راس های امکان پذیر را برای مورچه k زمانی که در راس i قرار دارد، معرفی می نماید. مجموعه راس های امکان پذیر، تنها شامل همسایه های بعد از راس i می باشند. برای جلوگیری از ایجاد حلقه، ممکن است N_i^k به صورت مجموعه تمامی راس های همسایه ای که تاکنون توسط مورچه k ملاقات نشده اند (از آن ها عبور نشده است) تعریف گردد. برای این منظور، یک لیست ممنوع برای هر مورچه تعریف می گردد. هر زمانی که یک مورچه یک راس جدید را ملاقات می نماید، آن راس تنها یکبار ملاقات می شود، راس های قرار گرفته در لیست ممنوع، از مجموعه N_i^k ، خارج می گردند.
- منیزو و کلورنی از یک فرمول متفاوت برای احتمال گذر استفاده کردند:

$$p_{ij}^k = \begin{cases} \frac{\alpha \tau_{ij}(t) + (1-\alpha) \eta_{ij}(t)}{\sum_{u \in N_i^k(t)} \alpha \tau_{iu}(t) + (1-\alpha) \eta_{iu}(t)} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \quad (23-2)$$

که در آن، α اهمیت نسبی غلظت فرومون $(\tau_{ij}(t))$ را نسبت به جذابیت $(\eta_{ij}(t))$ نشان می دهد. این فرمول به علت عدم نیاز به پارامتر β ، ساده تر از قبلی است. در این الگوریتم، معادلات تبخیر فرومون و بهنگام سازی فرومون شبیه به SACO است. روش های مختلفی برای نحوه انتشار فرومون توسط یک مورچه در یک مسیر وجود دارد که در زیر به بعضی از آن ها اشاره می شود:

- وابسته به کیفیت راه حل. در این حالت، مقدار فرومون منتشر شده متناسب با معکوس کیفیت مسیر ایجاد شده توسط یک مورچه است. در این فرمول Q ، مقدار ثابت مثبتی است:

$$\Delta \tau_{ij}^k(t) = \frac{Q}{f(x^k(t))} \quad (24-2)$$

- مقدار ثابت. در این حالت برای تمامی یال های قرار گرفته در مسیر مورچه، مقدار ثابت و برابری از فرومون منتشر می گردد.

$$\Delta \tau_{ij}^k(t) = Q \quad (25-2)$$

- وابسته به طول یال. در این حالت مقدار فرومون منتشر شده وابسته به طول یال قرار گرفته در مسیر مورچه می باشد.

$$\Delta \tau_{ij}^k(t) = \frac{Q}{d_{ij}} \quad (26-2)$$

دوریگو و همکارانش با انجام آزمایشی پیرامون مساله فروشنده دوره گرد و مساله تخصیص درجه دو همراه با ساختار های متفاوت انتشار فرومون، نشان دادند که روش انتشار فرومون وابسته به کیفیت راه حل، به دلیل این که از اطلاعات سراسری در ارتباط با کیفیت راه حل استفاده می نماید، نتایج بهتری را به وجود می آورد. آن ها همچنین از یک استراتژی نخبه گرایی برای توزیع فرومون استفاده نمودند. در این استراتژی، مقداری فرومون اضافه روی یال های کوتاه ترین مسیر انتخابی توسط مورچه ها، منتشر می گردد. در نتیجه، در این حالت فرمول انتشار فرومون به صورت زیر تغییر می یابد:

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \Delta \tau_{ij}(t) + n_e \cdot \Delta \tau_{ij}^e(t) \quad (27-2)$$

$$\Delta \tau_{ij}^e(t) = \begin{cases} \frac{Q}{f(\tilde{x}(t))} & \text{if } (i, j) \in \tilde{x}(t) \\ 0 & \text{otherwise} \end{cases} \quad (28-2)$$

که در آن e تعداد مورچه های نخبه بوده و $\tilde{x}(t)$ بهترین مسیر را معرفی می کند. در این حالت مقدار تابع حداقل برابر $f(\tilde{x}(t)) = \min_{k=1,2,3,\dots,n_k} \{f(x^k(t))\}$ می باشد.

۸-۲-۴-۲ سیستم کلونی مورچه ها

سیستم کلونی مورچه (ACS) توسط دوریگو و گامباردلا برای توسعه عملکرد AS ارائه

شد. ACS در چهار زمینه با AS متفاوت است. در آن:

- از یک قاعده متفاوت انتقال که قاعده شبه تصادفی نسبی^۱ نامیده می شود، استفاده می گردد.

- از قاعده متفاوتی برای به هنگام سازی فرومون استفاده می گردد.

- به هنگام سازی فرومون محلی به کار رفته است.

- از لیست کاندیدا برای طرف داری کردن از راس های خاصی استفاده می گردد.

قاعده انتقال ACS برای ایجاد یک بالانس قوی بین قابلیت های کاوش و تبعیت (تنوع و تشدید) الگوریتم توسعه داده شد. براساس آن، مورچه k که در راس i قرار دارد، راس بعدی j را برای حرکت از طریق قاعده زیر انتخاب می کند:

$$j = \begin{cases} \arg \max_{u \in N_i^k(t)} \{ \tau_{iu}^\alpha(t) \cdot \eta_{iu}^\beta(t) \} & \text{if } r \leq r_0 \\ J & \text{if } r > r_0 \end{cases} \quad (29-2)$$

که در آن $r \sim u(0,1)$ و $r_0 \in [0,1]$ پارامتری است که توسط کاربر معرفی می شود و $J \in N_i^k(t)$ راسی است که به طور تصادفی بر اساس احتمال زیر انتخاب می گردد:

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t) \cdot \eta_{ij}^\beta(t)}{\sum_{u \in N_i^k} \tau_{iu}^\alpha(t) \cdot \eta_{iu}^\beta(t)} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \quad (30-2)$$

پارامتر r_0 برای توازن بین کاوش و تبعیت استفاده می گردد. اگر $r \leq r_0$ باشد، الگوریتم تمایل پیدا می کند بهترین راس را انتخاب کند (تبعیت) و اگر $r > r_0$ باشد الگوریتم به کاوش می پردازد. بنابراین، انتخاب مقادیر کوچک r_0 ، موجب می گردد که تبعیت کمتر اتفاق افتاده و بیشتر کاوش صورت گیرد. بر خلاف AS در ACS تنها مورچه ای که دارای بهترین جواب است اجازه انتشار

¹ . Pseudo-random-proportional rule

فرومون مجدد را در مسیر طی شده دارد. این مورچه می تواند « بهترین هر تکرار^۱ » یا « بهترین در کل^۲ » باشد. فرومون از طریق قاعده بهنگام سازی سراسری^۳ زیر بهنگام می گردد:

$$\tau_{ij}(t+1) = (1-\rho_1)\tau_{ij}(t) + \rho_1\Delta\tau_{ij}(t) \quad (31-2)$$

$$\Delta\tau_{ij}(t) = \begin{cases} \frac{Q}{f(x^+(t))} & \text{if } (i,j) \in x^+(t) \\ 0 & \text{otherwise} \end{cases} \quad (32-2)$$

قاعده به هنگام سازی ACS به تشویق مورچه ها به سمت جستجو در اطراف بهترین راه حلی که تاکنون پیدا شده، می پردازد. این استراتژی از خاصیت تبعیت حمایت بیشتری کرده و بعد از این که تمامی مورچه ها مسیر را به انتها رساندند، به کار گرفته می شود.

اگر بروزرسانی بر اساس بهترین هر تکرار صورت گیرد $x^+(t)$ بهترین مسیر یافته شده در طول تکرار جاری t ، می باشد و اگر بروزرسانی بر اساس بهترین سراسری صورت گیرد، $x^+(t)$ بهترین مسیر یافته شده از اولین تکرار الگوریتم تاکنون است.

با استراتژی بهترین سراسری، فرآیند جستجو به دلیل استفاده از اطلاعات سراسری بیشتر، از مقبولیت بیشتری برخوردار است.

تبخیر فرومون نیز متفاوت از AS است. بر اساس معادله (31-2) برای مقادیر کوچک ρ_1 ، غلطت های فرومون مسیر به آرامی تبخیر می شود، در حالی که تاثیر بهترین مسیر تعدیل می گردد. در حالت دیگر برای مقادیر بزرگ ρ_1 ، فرومون منتشره قبلی به سرعت تبخیر شده و تاثیر بهترین مسیر تقویت می گردد.

علاوه بر قاعده به هنگام سازی سراسری، ACS از قاعده به هنگام سازی محلی زیر استفاده می کند.

¹. Iteration best

². Iteration best

³. Global Updating

$$\tau_{ij}(t+1) = (1-\rho_2)\tau_{ij}(t) + \rho_2\Delta\tau_{ij} \quad (33-2)$$

که در آن ρ_2 مقداری در محدوده (۰ و ۱) بوده و $\Delta\tau_{ij}$ می تواند سه حالت داشته باشد:

۱. استفاده از الگوریتم Q^1 که در آن :

$$\Delta\tau_{ij} = \gamma \cdot \max_{u \in N_i^k} \tau_{ij} \quad (34-2)$$

حداکثر فرومون موجود در مسیرهایی که می توان از گره i به آن ها رفت را انتخاب و در پارامتر γ ($0 < \gamma < 1$) ضرب کرده، به عنوان $\Delta\tau_{ij}$ انتخاب می کنند.

۲. $\Delta\tau_{ij} = \tau_0$ ، که در آن τ_0 مقدار فرومون اولیه می باشد.

۳. $\Delta\tau_{ij} = 0$ ، که در این حالت قاعده بهنگام سازی محلی انجام نمی گیرد.

نتایج بدست آمده از سه مقدار قبل نشان می دهد که استفاده از $\Delta\tau_{ij} = 0$ جواب های بدتری

نسبت به دو حالت قبل دارد [۲۲].

قاعده به هنگام سازی محلی برای آن است که اگر مورچه ای از یک مسیر عبور کرد مقدار فرومون موجود بر روی آن مسیر را کاهش دهد تا احتمال عبور سایر مورچه ها از آن مسیر کمتر گردد [۲۲].

همچنین ACS معنی مجموعه همسایگی را تغییر می دهد. مجموعه همسایه ها $N_i^k(t)$ ، شامل فهرستی از راس های کاندیدا می باشد. این راس های کاندیدا، راس هایی می باشند که نسبت به سایر راس ها ترجیح داده می شوند. فرض کنید $|N_i^k(t)| < n_l$ تعداد راس های در لیست کاندیدا را نمایش می دهد. راس های n_l راس هایی می باشند که به راس i نزدیک (از لحاظ مسافت یا هزینه) بوده و بر حسب فاصله آن ها مرتب می شوند. زمانی که لیست خالی است، راس j از راس های باقیمانده $N_i^k(t)$ انتخاب می گردد. یک راس غیر کاندیدا می تواند بر اساس معادله (۲-۳۰)، یا نزدیک ترین راس از $N_i^k(t)$ انتخاب گردد.

¹. Q-learning

۲-۴-۹ سیستم مورچگان رتبه بندی شده

سیستم مورچه رتبه بندی شده (AS_{rank}) که در این پایان نامه از آن استفاده شده است برای اولین بار توسط بولنهاایمر^۱ و همکارانش ارائه گردید [۲۳]. در این روش در هر تکرار، مورچه ها بر حسب میزان هدفی که بدست آورده اند رتبه بندی شده و سپس تعدادی از بهترین مورچه ها متناسب با رتبه ای که دارند فرومون می ریزند. همچنین بهترین مورچه از کل تکرار های قبلی نیز فرومون می ریزد. بدین ترتیب رابطه به هنگام سازی فرومون در این الگوریتم به صورت معادله (۲-۳۵) می باشد.

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{\mu=1}^{w-1} \Delta\tau_{ij}^{\mu} + w \cdot \Delta\tau_{ij}^{bs} \quad (2-35)$$

در این رابطه w مشخص کننده تعداد مورچه هایی هستند که می توانند فرومون بریزند که معمولاً درصدی کوچک از جمعیت مورچه ها در نظر گرفته می شود. همچنین $\Delta\tau_{ij}^{bs}$ مقدار فرومون مربوط به بهترین مورچه در کل تکرار ها و $\Delta\tau_{ij}^{\mu}$ مقدار فرومون متناظر با مورچه دارای رتبه μ است که این مقایر به صورت روابط زیر بیان می گردند.

$$\Delta\tau_{ij}^{bs} = \frac{1}{c^{bs}} \quad (2-36)$$

$$\Delta\tau_{ij}^{\mu} = \frac{w - \mu}{c^{\mu}} \quad (2-37)$$

دراین روابط c^{bs} و c^{μ} به ترتیب طول مسیر طی شده توسط بهترین مورچه کلی و مورچه با رتبه μ می باشند. تحقیقات بولنهاایمر و همکارانش بیانگر این است که جواب های بدست آمده از این الگوریتم به مقدار قابل توجهی بهتر از پاسخ های بدست آمده از الگوریتم سیستم مورچگان بوده است.

¹ . Bullnheimer

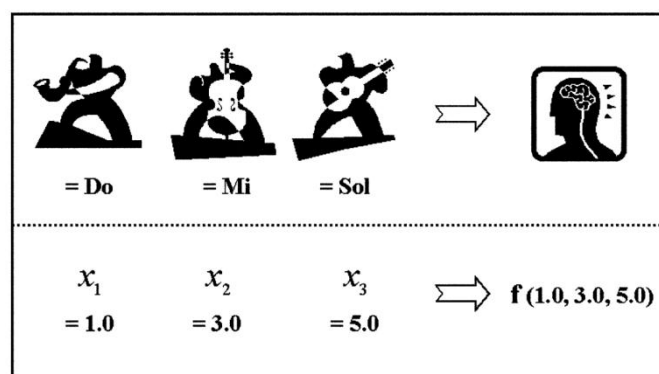
۲-۵ الگوریتم جستجوی هارمونی

الگوریتم فرا ابتکاری جستجوی هارمونی که اخیرا توسط گیم و همکاران ایجاد شده است، از قوانینی که نوازندگان برای جستجوی یک هارمونی زیبا در موسیقی استفاده می کنند، الهام گرفته است. قبل از آن که با جزئیات بیشتری به توصیف الگوریتم جستجوی هارمونی بپردازیم، ابتدا به شباهت هایی که بین کار نوازندگان و روش های بهینه یابی وجود دارد می پردازیم.

نوازندگان تلاش می کنند با استفاده از یک سری قوانین موسیقی، یک هارمونی زیبا را بیابند که زیبا بودن آن بر اساس استاندارد زیبا شناختی صوتی ارزیابی می گردد. دقیقا به همان صورتی که روش های بهینه یابی تلاش می کنند تا یک جواب بهینه مطلق را بیابند که بهینه بودن آن بر اساس مقدار تابع هدف ارزیابی می شود. هر هارمونی نواخته شده توسط نوازندگان در واقع ترکیبی از نت هایی است که هر نوازنده با ساز خود می نوازد، درست همانطور که مقدار تابع هدف بر اساس مجموعه مقادیری که به هر متغیر تصمیم آن اختصاص داده می شود، تعیین می گردد.

شکل (۲-۸) مقایسه ای بین کار نوازندگان و روش های بهینه یابی را با جزئیات بیشتر نشان می دهد. در موسیقی، هر نوازنده از محدوده مجاز نت های سازش، یک نت را می نوازد و ترکیب این نت های نواخته شده توسط نوازندگان تشکیل یک هارمونی جدید را می دهد. اگر ترکیب نت های تولید شده منجر به یک هارمونی زیبا شود، این هارمونی در ذهن هر نوازنده ذخیره می شود. این عمل احتمال تولید هارمونی زیبا تر در مرحله بعدی را افزایش می دهد. به طور مشابه در روش های بهینه یابی، مقدار هر متغیر تصمیم تابع هدف از درون محدوده مجاز آن متغیر انتخاب می شود و اگر ترکیب این مقادیر منجر به یک مقدار بهینه برای تابع هدف مفروض گردد، این بردار جواب به همراه مقدار تابع هدف مربوطه اش در یک حافظه ذخیره می شود، و این عمل نیز احتمال انتخاب یک بردار

جواب بهینه تر را در تکرار بعدی افزایش می دهد.



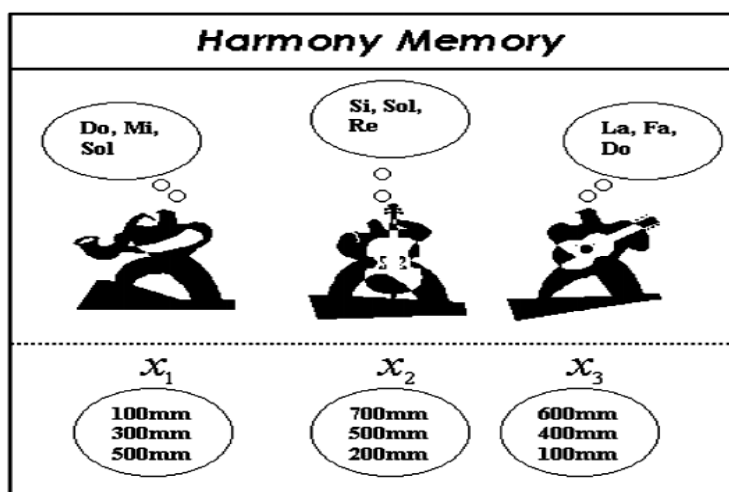
شکل (۲-۸) - مقایسه ای بین کار نوازندگان و روش های بهینه یابی [۲۴]

قسمت اصلی الگوریتم جستجوی هارمونی، حافظه هارمونی^۱ می باشد که در شکل (۲-۹) ساختار آن نشان داده شده است. یک جاز سه نفره را در نظر بگیرید که از ساکسوفون، باس و گیتار تشکیل شده است. فرض کنید نت های برتر در حافظه هر نوازنده به این صورت باشد که: نوازنده ساکسوفون، {Do, Mi, Sol}؛ نوازنده باس، {Si, Sol, Re}؛ و نوازنده گیتار، {La, Fa, Do} اگر نوازنده ساکسوفون، باس و گیتار به ترتیب و به صورت تصادفی از حافظه خود {Sol}، {Si} و {Do} را انتخاب کرده و بنوازند، آنگاه هارمونی جدیدی تولید خواهد شد که به صورت {Sol, Si, Do} می باشد. حال اگر هارمونی جدید تولید شده بهتر از بدترین هارمونی در درون حافظه هارمونی باشد، این هارمونی جدید در حافظه ذخیره شده و بدترین هارمونی از حافظه حذف می شود. این عمل تکرار می گردد تا بهترین هارمونی پیدا گردد.

در مسائل بهینه یابی، هر نوازنده می تواند با هر متغیر تصمیم جایگزین گردد و نت های برتر ذخیره شده در حافظه هر نوازنده نیز می تواند با مقادیر مناسب برای هر متغیر تصمیم جایگزین گردد. اگر فرض کنیم که سه متغیر تصمیم داریم و مقادیر انتخابی برای آنها به این صورت باشد که: متغیر

¹ Harmony memory (HM)

اول {۱۰۰} میلیتر از میان {۱۰۰، ۳۰۰، ۵۰۰} میلیتر، متغیر دوم {۵۰۰} میلیتر از میان {۷۰۰، ۵۰۰، ۲۰۰} میلیتر و متغیر سوم {۴۰۰} میلیتر از میان {۶۰۰، ۴۰۰، ۱۰۰} میلیتر، آنگاه بردار حل جدید تولید شده به صورت {۱۰۰، ۵۰۰، ۴۰۰} میلیتر می باشد.



شکل (۲-۹) - ساختار حافظه هارمونی [۲۴]

حال اگر این بردار حل جدید نسبت به بردارهای موجود در حافظه هارمونی، مقدار تابع هدف را بهینه تر کند، این بردار حل در حافظه ذخیره شده و بدترین بردار حل از حافظه پاک می شود. این عمل تا زمانی که به حداکثر مقدار برسیم تکرار می شود. در مسائل بهینه یابی مهندسی، تابع هدف می تواند نماینده کمترین هزینه، کمترین زمان یا کمترین خطا و یا بیشترین سود و.. باشد.

شباهت بین کار نوازندگان در موسیقی و عملکرد روش های بهینه یابی را می توان به صورت

زیر خلاصه کرد:

ساز	←	متغیر تصمیم
دامنه نت	←	دامنه متغیر تصمیم
نت انتخاب شده	←	مقدار متغیر تصمیم

هارمونی	←	بردار حل
علم زیبا شناختی	←	تابع هدف
تمرین	←	تکرار
تجربه	←	حافظه هارمونی

برای آنکه جستجوی هارمونی را بهتر درک کنیم، ابتدا باید ببینیم نوازندگان موسیقی با استفاده از چه قوانینی یک هارمونی را تولید می کنند. وقتی یک نوازنده فالبداهه می نوازد، او سه انتخاب در پیش رو دارد: (۱) نواختن یک نت دقیقاً از درون حافظه خود، (۲) نواختن نت مجاور یکی از نت های در درون حافظه خود و (۳) نواختن یک نت به صورت کاملاً تصادفی از میان نت های مجاز ساز خود. گیم و همکاران در سال ۲۰۰۱ این سه انتخاب نوازندگان را برای حل مسائل بهینه یابی فرموله کردند، و سه انتخاب نوازندگان به ترتیب قانون: در نظرگیری حافظه^۱، تنظیم نت^۲ و انتخاب تصادفی^۳ نامیده شدند.

به این ترتیب وقتی هر متغیر در الگوریتم جستجوی هارمونی یک مقدار را برای خود بر می گزیند، آن از یکی از این سه قانون پیروی می کند: (۱) انتخاب یک مقدار از درون حافظه الگوریتم جستجوی هارمونی (در نظر گرفتن حافظه)، (۲) انتخاب مقدار مجاور یکی از مقادیر درون حافظه الگوریتم (تنظیم نت) و (۳) انتخاب کاملاً تصادفی یک مقدار برای متغیر مورد نظر از میان دامنه مجاز آن متغیر (انتخاب تصادفی).

همان طور که ذکر شد قانون اول انتخاب، در الگوریتم جستجوی هارمونی، قانون در نظرگیری حافظه نام دارد. دلیل اهمیت استفاده از حافظه هارمونی در آن است که ما اطمینان حاصل کنیم که

1Memory considerations
2 Pitch adjustments
3 Randomization

هارمونی های خوب به عنوان المان هایی از هارمونی جدید در نظر گرفته می شوند. در واقع در این حافظه، تمام هارمونی های که ذخیره می شوند، نخبه ترین هارمونی هایی هستند که تا تکرار مربوطه، الگوریتم به آن ها دست یافته است. به منظور استفاده موثر از حافظه، الگوریتم جستجوی هارمونی یک پارامتری به نام نرخ در نظرگیری حافظه^۱ (HMCR) را اتخاذ می کند. اگر مقدار این پارامتر خیلی کم در نظر گرفته شود، نت های کمی از هارمونی های نخبه انتخاب خواهند شد و این سبب می شود که الگوریتم به کندی همگرا شود. و اگر مقدار این پارامتر خیلی زیاد در نظر گرفته شود، آنگاه فقط نت های درون حافظه همواره انتخاب می شوند و نت های دیگر به خوبی کشف نمی شوند و این عمل منجر به حل مناسبی نخواهد گردید. ما به طور معمول همواره مقداری بین ۰.۷ ~ ۰.۹۹ را برای این پارامتر انتخاب می کنیم. اگر مقدار این پارامتر را یک اتخاذ کنیم، آنگاه احتمال آنکه یک نتی غیر از نت های ذخیره شده در حافظه انتخاب گردد، وجود نخواهد داشت و این قابل قبول نیست به همین خاطر بیشترین مقداری که برای این پارامتر می تواند اتخاذ کرد ۰.۹۹ خواهد بود.

قانون دوم انتخاب در الگوریتم جستجوی هارمونی قانون تنظیم نت می باشد. الگوریتم برای اعمال این قانون، از دو پارامتر به نام های نرخ تنظیم نت^۲ (PAR) و پهنای باند^۳ (bw) بهره می برد. همان گونه که در موسیقی، تنظیم نت به معنی تغییر فرکانس است در الگوریتم جستجوی هارمونی نیز به معنی یک تغییر جزئی در مقدار انتخاب شده برای متغیر مربوطه می باشد [۲۵]. در حالت نظری، نت می تواند هم به صورت خطی و هم به صورت غیرخطی تنظیم گردد، اما در این الگوریتم، تنظیم خطی مورد استفاده قرار گرفته است. بنابراین خواهیم داشت:

$$x_{new} = x_{old} + bw \times \varepsilon \quad (38-2)$$

^۱ Harmony memory considering rate (HMCR)

^۲ Pitch adjusting rate (PAR)

^۳ Bandwidth (bw)

به طوریکه x_{old} نت موجود ذخیره شده در حافظه هارمونی می باشد و x_{new} نت جدید بعد از اعمال قانون تنظیم نت می باشد. در واقع این عمل باعث می گردد با اضافه کردن یک مقدار تصادفی کوچک به نت موجود در حافظه، یک نت جدید تولید گردد [۲۴]. در اینجا ε یک عدد تصادفی با توزیع یکنواخت در محدوده ی $[-1, 1]$ می باشد. تنظیم نت در الگوریتم جستجوی هارمونی همانند عملگر جهش در الگوریتم ژنتیک عمل می کند. الگوریتم، نرخ تنظیم نت را برای کنترل میزان تنظیم اتخاذ می کند. انتخاب مقدار برای هر دو پارامتر در این قانون از اهمیت بالایی برخوردار است، زیرا آن ها پارامترهای بسیار مهمی در تنظیم دقیق بردارهای جواب ذخیره شده در حافظه می باشند و همچنین در تنظیم سرعت همگرایی الگوریتم برای یافتن جواب بهینه مطلق نیز مفید واقع می گردند. در مسائل با متغیرهای گسسته نت جدید از بین مقادیر گسسته متغیرهای تصمیم، در همسایگی متغیر مربوط به x_{old} (نت ذخیره شده در حافظه) انتخاب می گردند.

قانون سوم در این الگوریتم، قانون انتخاب تصادفی می باشد، که باعث افزایش تنوع و گوناگونی بردارهای حل می گردد. هر چند در الگوریتم جستجوی هارمونی قانون تنظیم نت نیز عملکرد مشابهی با قانون انتخاب تصادفی دارد، اما آن به یک محدوده خاصی از کل فضای جستجو محدود شده است و می توان گفت همانند یک جستجوگر محلی عمل می کند.

به طور خلاصه احتمال استفاده الگوریتم از هر یک از این سه قانون را می توان به صورت زیر

تعریف کرد [۲۶]:

احتمال انتخاب از حافظه برابر است با اندازه نرخ در نظرگیری حافظه (HMCR)

احتمال اعمال قانون تنظیم نت برابر است با ضرب نرخ در نظرگیری حافظه در نرخ تنظیم

نت (HMCR×PAR)

احتمال انتخاب تصادفی برابر است با یک منهای نرخ در نظرگیری حافظه (1- HMCR)

۲-۵-۱ مراحل بهینه یابی با جستجوی هارمونی

در شکل (۲-۱۰) مراحل بهینه یابی با استفاده از الگوریتم فرا ابتکاری جستجوی هارمونی نشان داده شده است، همان طور که ملاحظه می گردد این الگوریتم شامل پنج مرحله می باشد [۲۶]:

مرحله (۱): مقداردهی اولیه به مسئله بهینه یابی و پارامترهای الگوریتم

مرحله (۲): مقداردهی اولیه به حافظه هارمونی

مرحله (۳): تولید یک هارمونی جدید با استفاده از سه قانون الگوریتم جستجوی هارمونی

مرحله (۴): به روز رسانی حافظه هارمونی

مرحله (۵): تکرار مرحله ۳ و ۴ تا رسیدن به تکرار نهایی

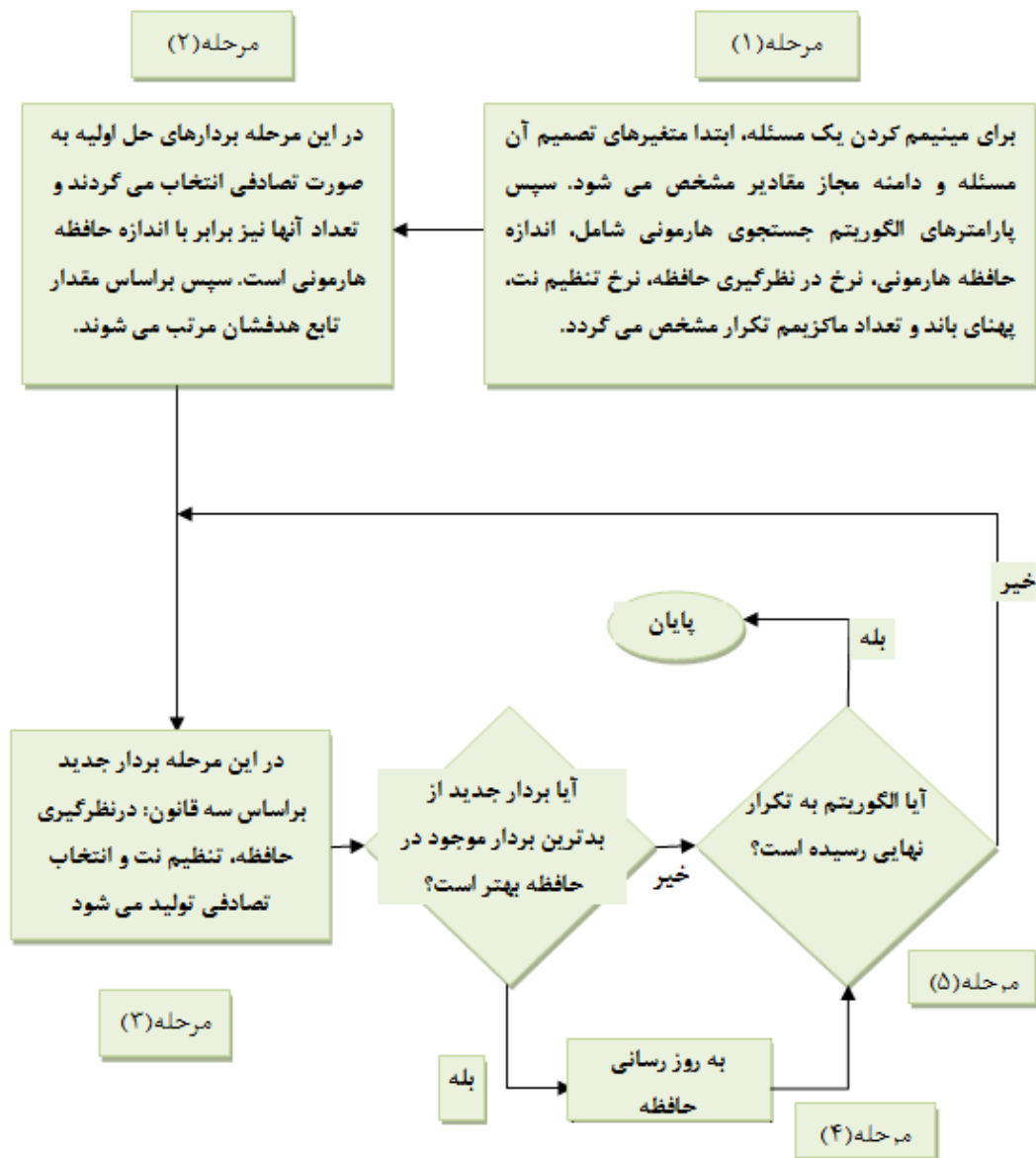
۲-۵-۱-۱ مقداردهی اولیه به مسئله بهینه یابی و پارامترهای الگوریتم

در ابتدا، مسئله بهینه یابی به صورت زیر تعریف می شود:

$$\text{Minimize } f(x) \text{ s.t. } x_i \in X_i, \quad i = 1, 2, \dots, N \quad (2-39)$$

به طوری که $f(x)$ تابع هدف می باشد که مقدار بهینه برای آن باید محاسبه گردد، x_i مجموعه ای از متغیرهای تصمیم می باشد، X_i مجموعه ای از محدوده مقادیر ممکن برای هر متغیر تصمیم می باشد، به طوری که خواهیم داشت، $x_{iL} \leq x_i \leq x_{iU}$ (حد پایین و x_{iU} حد بالا برای هر متغیر تصمیم است) و N تعداد متغیرهای تصمیم می باشد. همچنین پارامترهای الگوریتم جستجوی هارمونی برای حل مسئله بهینه یابی نیز در این مرحله مقداردهی می شوند. این پارامترها شامل اندازه حافظه هارمونی^۱ (تعداد بردارهای حل در حافظه هارمونی)، نرخ در نظرگیری حافظه، نرخ تنظیم نت، مقدار پهنای باند و ماکزیمم تعداد تکرار می باشند.

¹ Harmony memory size (HMS)



شکل (۲-۱۰) - مراحل بهینه یابی الگوریتم جستجوی هارمونی

۲-۱-۵-۲ مقداردهی اولیه به حافظه هارمونی

در این مرحله ماتریس حافظه هارمونی، نشان داده شده در معادله (۲-۴۰) با بردارهای حلی که به صورت تصادفی تولید شده اند و بر اساس مقدار تابع هدفشان $(f(x))$ مرتب گردیده اند، پر می شود.

$$HM = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_{N-1}^1 & x_N^1 \\ x_1^2 & x_2^2 & \cdots & x_{N-1}^2 & x_N^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{HMS-1} & x_2^{HMS-1} & \cdots & x_{N-1}^{HMS-1} & x_N^{HMS-1} \\ x_1^{HMS} & x_2^{HMS} & \cdots & x_{N-1}^{HMS} & x_N^{HMS} \end{bmatrix} \quad (2-40)$$

۳-۱-۵-۲ تولید یک هارمونی جدید

در این مرحله، یک بردار جدید همانند $x' = (x'_1, x'_2, \dots, x'_N)$ با استفاده از سه قانون الگوریتم جستجوی هارمونی (در نظرگیری حافظه، تنظیم نت و انتخاب تصادفی) تولید می شود، به عنوان مثال، الگوریتم ممکن است برای متغیر اول (x'_1) ، یا یک مقدار از حافظه یعنی $(x_1^1 - x_1^{HMS})$ اختصاص دهد یا یک مقدار به صورت تصادفی از محدوده مجاز متغیر اول (X_1) برای آن انتخاب کند و یا بر روی مقداری که از حافظه انتخاب شده است، قانون تنظیم نت را اعمال نماید. به این نکته باید توجه داشت، که قانون تنظیم نت فقط به روی متغیرهایی اعمال می گردد که الگوریتم، مقدار آنها را از درون حافظه انتخاب کرده است و برای متغیرهایی که مقدارشان به صورت تصادفی انتخاب شده است قانون تنظیم نت اعمال نخواهد شد. احتمال اعمال هر یک از این قوانین برای انتخاب یک مقدار جدید برای متغیرها، در بالا ارائه گردیده است.

۴-۱-۵-۲ به روز رسانی حافظه هارمونی

به روز رسانی حافظه هارمونی. در این مرحله، اگر مقدار تابع هدف بدست آمده از بردار تولید شده جدید، بهتر از مقدار تابع هدف بدترین بردار موجود در حافظه باشد، بدترین بردار جای خود را به بردار جدید می دهد و حافظه بر اساس مقدار تابع هدف مجددا مرتب می گردد.

۵-۱-۵-۲ بررسی معیار توقف

در این مرحله اگر الگوریتم به تکرار نهایی برسد، به عملیات جستجو پایان می دهد. در غیر این صورت، مراحل ۳ و ۴ تکرار می گردد.

۲-۵-۲ بررسی پارامترهای الگوریتم جستجوی هارمونی

حال که الگوریتم جستجوی هارمونی مورد بررسی قرار گرفت، می خواهیم به چند سوال در مورد این الگوریتم پاسخ دهیم. الگوریتم جستجوی هارمونی چگونه تنوع و تشدید (کاوش و تبعیت) را در حل مسائل بهینه یابی به کار می گیرد؟ الگوریتم جستجوی هارمونی با استفاده از چه پارامترهایی عملیات جستجوی محلی و با استفاده از چه پارامترهایی عملیات جستجوی کلی را انجام می دهد؟ همان طور که قبلا نیز مطرح شد، الگوریتم جستجوی هارمونی از سه قانون برای تولید بردار جواب جدید استفاده می کند. برای اعمال این سه قانون، الگوریتم از دو پارامتر استفاده می کند. این دو پارامتر، نرخ در نظر گیری حافظه و نرخ تنظیم نت نامیده می شوند. که به ترتیب آنها را با نام های HMCR و PAR نشان می دهیم. الگوریتم جستجوی هارمونی برای اعمال تنوع بر روی دسته جواب های ذخیره شده در حافظه هارمونی از دو قانون استفاده می کند:

- قانون تنظیم نت
- قانون انتخاب تصادفی

در این الگوریتم، انتخاب تصادفی یک مقدار برای هر متغیر تصمیم، به هیچ عنوان هوشمند نمی باشد و تنها یک عمل کاملاً تصادفی می باشد که باعث افزایش تنوع دسته جواب ها می شود. اما قانون تنظیم نت برعکس انتخاب تصادفی، یک عملگر کاملاً هوشمند می باشد، که بوسیله پارامتر PAR اعمال می گردد. الگوریتم جستجوی هارمونی علاوه بر انتخاب تصادفی (غیر هوشمند) مقادیر برای هر متغیر تصمیم، از یک عملگر هوشمند به نام تنظیم نت نیز برای اعمال تنوع در تولید دسته جواب ها استفاده می کند. اما نکته ای که می بایست به آن دقت شود آن است که الگوریتم جستجوی هارمونی با استفاده از قانون تنظیم نت یک مقدار جزئی به مقادیر موجود در حافظه اضافه می کند، به

همین خاطر نمی توان این عملگر (قانون تنظیم نت) را به عنوان یک جستجوگر کلی به حساب آورد بلکه فقط یک جستجوگر محلی به حساب می آید. در صورتی که انتخاب تصادفی به عنوان یک جستجوگر کلی در الگوریتم جستجوی هارمونی عمل می کند. در واقع این الگوریتم از دو نوع تنوع برای انجام عملیات جستجوی خود استفاده می کند، یکی با استفاده از قانون انتخاب تصادفی که تنوع از نوع کلی می باشد و دیگری با استفاده از قانون تنظیم نت که تنوع از نوع جزئی می باشد، زیرا الگوریتم با استفاده از این قانون (تنظیم نت)، تنوع را فقط حول بردار جواب های خوب که در حافظه هارمونی ذخیره شده اند، اعمال می کند و به صورت خیلی جزئی بعضی از مقادیری که از حافظه انتخاب شده اند را تغییر می دهد.

اما از سوی دیگر، این الگوریتم برای اعمال تشدید بر روی دسته جواب ها از یک قانون استفاده می کند:

- قانون در نظر گیری حافظه

این قانون هم، همانند قانون تنظیم نت یک عملگر کاملاً هوشمند به حساب می آید که به وسیله پارامتر HMCN اعمال می گردد. اگر فرض کنیم که مقدار آن ۰.۹۵ باشد، الگوریتم به احتمال ۹۵ درصد برای انتخاب یک مقدار برای هر متغیر تصمیم از یکی از مقادیر ذخیره شده، متناظر با آن متغیر، در درون حافظه هارمونی استفاده می کند. و به احتمال ۵ درصد یک مقدار تصادفی برای متغیر مورد نظر از دامنه مجاز آن، انتخاب می کند. در واقع این قانون، بیانگر یک نوع نخبه گرایی می باشد. زیرا الگوریتم را مجبور می کند که در تشکیل بردار جواب جدید از مولفه های بردارهای ذخیره شده در حافظه هارمونی استفاده کند و همان طور که می دانیم با افزایش تعداد تکرارها، به کیفیت بردارهای ذخیره شده در حافظه افزوده می شود، به همین خاطر این عمل، بیانگر نخبه گرایی می باشد. اگر مقدار این پارامتر کم باشد، سبب می گردد الگوریتم به کندی همگرا شود. همان طور که

قبلا نیز ذکر شد، زیاد بودن مقدار تشدید در الگوریتم های فرا ابتکاری باعث ایجاد همگرایی زودرس خواهد شد. اما به دلیل آنکه در الگوریتم جستجوی هارمونی بر روی متغیرهایی که مقادیرشان از داخل حافظه انتخاب شده است به احتمال PAR عمل تنظیم نت صورت می گیرد، از نارس شدن جواب ها جلوگیری می شود، که بسیار حائز اهمیت می باشد. به همین خاطر حتی انتخاب یک مقدار بالا برای پارامتر (HMCR) نیز مشکل ساز نخواهد شد.

این پارامتر یکی از هوشمندترین عملگرهای این الگوریتم می باشد. اگرچه مولفه تشدید در الگوریتم جستجوی هارمونی توسط این پارامتر اعمال می گردد، اما نکته قابل توجه در این است که، چون مقادیر هر متغیر تصمیم با یک توزیع یکنواخت از دامنه مجازشان انتخاب می شوند در نتیجه پراکندگی بین مقادیر ذخیره شده برای هر متغیر در حافظه هارمونی در تکرارهای ابتدایی بسیار زیاد می باشد، به همین خاطر انتخاب هر مقدار برای متغیرهای تصمیم از درون حافظه در تکرار ابتدایی همانند یک جستجوی کلی برای آن متغیر می باشد در نتیجه این عمل باعث می شود در تکرارهای ابتدایی حتی پارامتر HMCR که برای تشدید طراحی شده است به عنوان یک عملگر تنوع عمل نماید که بسیار مفید می باشد. اما در تکرارهای انتهایی دقیقا برعکس خواهد شد. از آنجا که در تکرارهای انتهایی، مقادیر همه متغیرهای تصمیم به سمت مقدار بهینه خود همگرا شده اند، در نتیجه پراکندگی مقادیر هر متغیر در حافظه به صورت چشمگیری کاهش می یابد. به همین خاطر انتخاب هر مقداری از درون حافظه برای هر متغیر، در تکرارهای انتهایی همانند یک جستجوی محلی برای آن متغیر می باشد. این همان تعادل مناسبی می باشد که الگوریتم جستجوی هارمونی به بهترین نحو ممکن در پارامتر HMCR خود ایجاد کرده است.

۲-۶ الگوریتم جستجوی ذرات باردار

این بخش یک الگوریتم بهینه سازی جدید را مبنی بر برخی اصول و قوانین از فیزیک و مکانیک معرفی می کند که CSS نامیده می شود. در این بخش از قوانین حاکم کلمب از الکترواستاتیک و قوانین نیوتن از مکانیک استفاده شده است. CSS یک روش چند عنصره است که هر عنصر یک ذره باردار می باشد. ذرات باردار می توانند بر اساس مقادیر شایستگی و فاصله جدایی شان بر یکدیگر اثر بگذارند. مقدار نیروی برآیند با استفاده از قوانین الکترواستاتیک و چگونگی جابه جایی آن ها با استفاده از قوانین مکانیکی نیوتن تعیین می شود. CSS می تواند در همه زمینه های بهینه سازی استفاده گردد. مخصوصاً آن برای حوزه ناهموار و غیر محدب مناسب می باشد. CSS به اطلاعات گرایان و پیوسته بودن فضای جستجو نیاز ندارد.

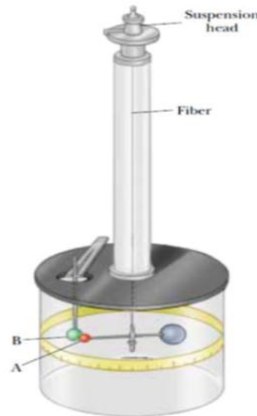
در ادامه سعی بر این داریم تا به شرح کامل روش جستجوی ذرات باردار با استفاده از قواعد و اصول فیزیکی و مکانیکی کلمب و نیوتن بپردازیم.

۲-۶-۱ مروری بر قوانین الکتریکی و مکانیکی

۲-۶-۱-۱ قوانین الکتریکی

در فیزیک فضای اطراف یک بار الکتریکی یک میدان الکتریکی ایجاد می کند که یک نیرو بر اجسام باردار الکتریکی دیگر وارد می کند. میدان الکتریکی یک بار نقطه ای توسط قانون کلمب نشان داده شده است. کلمب تایید کرد که نیروی الکتریکی بین دو گوی باردار کوچک با مربع معکوس فواصل جدایی بین آن ها متناسب است. نیروی الکتریکی بین گوی های باردار A و B در شکل (۲-۱۱) باعث می شود که گوی ها به سمت همدیگر جذب شوند یا همدیگر را دفع نمایند و نتیجه این

حرکت باعث پیچش فیبر معلق می گردد. از آنجایی که لنگر باز گرداننده فیبر تابیده با زاویه ای که به واسطه آن فیبر دوران می کند متناسب است، اندازه گیری این زاویه، یک اندازه گیری کمی نیروی الکتریکی جاذبه یا دافعه را فراهم می کند.



شکل (۲-۱۱) - ترازوی پیچشی کولمب

آزمایشات کلمب نشان می دهد که نیروی الکتریکی بین دو ذره باردار ثابت:

- نسبت معکوس با مربع فاصله جدایی بین ذرات و راستای خطی که آن ها را به هم متصل می کند دارد.
- متناسب با بارهای تولید شده q_i و q_j در دو ذره می باشد.
- نیرو جاذبه است اگر بار ها علامت مخالف داشته باشند و دافعه است اگر بارها علامت های یکسان داشته باشند.

با توجه به ملاحظات بالا قوانین کلمب بزرگی نیروی الکتریکی بین دو ذره باردار را به صورت

رابطه (۲-۴۱) بیان می کند [۲۷]:

$$f_{ij} = k_e \frac{q_i q_j}{r_{ij}^2} \quad (۲-۴۱)$$

که k_e یک ثابت است که ثابت کلمب نامیده می شود. r_{ij} فاصله بین دو بار می باشد. یک کره

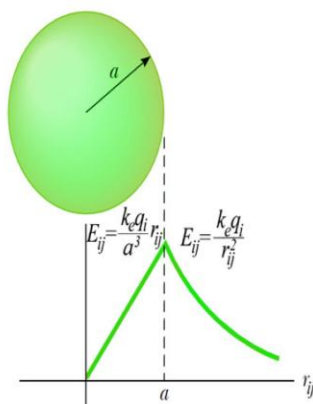
توپر عایق با شعاع a را در نظر بگیرید که چگالی حجمی بار یکنواختی دارد و یک بار مثبت q را حمل می‌کند. میدان الکتریکی E_{ij} در یک نقطه بیرون گوی به صورت معادله (۴۲-۲) تعریف می‌گردد [۲۷].

$$E_{ij} = k_e \frac{q_i}{r_{ij}^2} \quad (۴۲-۲)$$

بزرگی میدان الکتریکی در یک نقطه درون گوی می‌تواند با قانون گاوس بدست آید که این قانون به صورت معادله (۴۳-۲) بیان می‌گردد [۲۷].

$$E_{ij} = k_e \frac{q_i}{a^3} r_{ij} \quad (۴۳-۲)$$

توجه داشته باشید که این معادله نشان می‌دهد که هنگامی که $r_{ij} \rightarrow 0$ آنگاه میدان الکتریکی نیز به سمت صفر میل می‌کند ($E_{ij} \rightarrow 0$). بنابراین این معادله مشکلی که در $r_{ij} = 0$ وجود خواهد داشت را برطرف می‌کند و میدان الکتریکی درون گوی با $(1/r_{ij}^2)$ متناسب نمی‌شود. این بدین معنی است که اگر $E_{ij} \propto (1/r_{ij}^2)$ باشد، میدان در $r_{ij} = 0$ بی نهایت خواهد بود که به صورت فیزیکی غیر ممکن است. از این رو میدان الکتریکی درون گوی به صورت خطی با r_{ij} تغییر می‌کند. میدان بیرون گوی همان گونه است که یک بار نقطه‌ای q_i در $r_{ij} = 0$ قرار گیرد، همچنین بزرگی میدان الکتریکی برای یک نقطه درون یا بیرون گوی وقتی که $r_{ij} = a$ می‌شود برابر است. نمودار E_{ij} در برابر r_{ij} در شکل (۱۲-۲) نشان داده شده است.



شکل (۱۲-۲) - نمودار E_{ij} در برابر r_{ij} برای یک گوی عایق با بار یکنواخت

به منظور محاسبه میدان الکتریکی در یک نقطه (r_j) ناشی از یک گروه از بارهای نقطه‌ای، اصل برهم نهی برای میدان‌هایی که به طور مستقیم از برهم نهی نیروهای الکتریکی پیروی می‌کنند به کار می‌رود، بنابراین میدان الکتریکی یک گروه بار می‌تواند به صورت رابطه (۴۴-۲) بیان گردد.

$$E_j = \sum_{i=1, i \neq j}^N E_{ij} \quad (44-2)$$

که N تعداد کل ذرات باردار می‌باشد و E_{ij} برابر است با معادله (۴۵-۲):

$$E_{ij} = \begin{cases} k_e \frac{q_i}{a^3} r_{ij} \rightarrow \text{if } r_{ij} < a \\ k_e \frac{q_i}{r_{ij}^2} \rightarrow \text{if } r_{ij} > a \end{cases} \quad (45-2)$$

به منظور بدست آوردن بزرگی و راستای یک نیروی برآیند روی یک بار q_j در موقعیت r_j ناشی از میدان الکتریکی بار q_i در موقعیت r_i شکل برداری کامل مورد نیاز می‌باشد که به صورت رابطه (۴۶-۲) بیان می‌گردد:

$$F_{ij} = E_{ij} q_j \frac{r_i - r_j}{\|r_i - r_j\|} \quad (46-2)$$

برای چندین ذره باردار معادله (۴۶-۲) می‌تواند به صورت معادله (۴۷-۲) خلاصه شود:

$$F_j = k_e q_j \sum_{i, i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) \frac{r_i - r_j}{\|r_i - r_j\|} \quad (47-2)$$

۲-۱-۶-۲ قوانین مکانیک برداری

مکانیک برداری یا مکانیک کلاسیک، حرکت اجسام را مطالعه می‌کند. در مطالعه حرکت، اجسام محرک صرفنظر از اندازه‌شان به عنوان یک ذره توصیف می‌شوند. در حالت کلی یک ذره یک نقطه توده مانند می‌باشد که اندازه‌اش بی‌نهایت کوچک است. اگر موقعیت ذرات در فضا در هر زمان

مشخص باشد، حرکت یک ذره به طور کامل معلوم است. جابه‌جایی یک ذره با تغییر در موقعیتش تعریف می‌شود، در نتیجه حرکت ذره از یک موقعیت ابتدایی r_{old} به یک نقطه نهایی r_{new} ، جابه‌جایی آن با فرمول (۴۸-۲) داده می‌شود:

$$\Delta r = r_{new} - r_{old} \quad (۴۸-۲)$$

شیب خط مماس بر موقعیت ذره نشان دهنده سرعت ذره می باشد که برابر است با:

$$V = \frac{r_{new} - r_{old}}{t_{new} - t_{old}} = \frac{\Delta r}{\Delta t} \quad (۴۹-۲)$$

وقتی سرعت یک ذره با زمان تغییر می‌کند گفته می‌شود که ذره شتاب دار است. شتاب ذرات با تغییر سرعت تقسیم بر فاصله زمانی که تغییر سرعت در آن رخ می‌دهد تعریف می‌گردد، که در معادله (۵۰-۲) نشان داده شده است.

$$a = \frac{V_{new} - V_{old}}{\Delta t} \quad (۵۰-۲)$$

با استفاده از معادلات (۴۸-۲) تا (۵۰-۲) جابه‌جایی هر جسم به عنوان تابعی از زمان به طور تقریبی از زمان از رابطه (۵۱-۲) بدست می‌آید:

$$r_{new} = \frac{1}{2} a \cdot \Delta t^2 + V_{old} \cdot \Delta t + r_{old} \quad (۵۱-۲)$$

قانون دیگری که در این جا استفاده می‌گردد قانون دوم نیوتن است که به سوال زیر پاسخ می‌دهد: وقتی یک نیروی برآیند غیر صفر بر یک جسم وارد می‌شود، چه اتفاقی رخ می‌دهد؟ شتاب یک جسم (a) به طور مستقیم با نیروی خالصی که بر آن اعمال می‌گردد و به طور معکوس با جرم (m) آن متناسب است.

$$F = m \cdot a \quad (۵۲-۲)$$

۲-۶-۲ روش جستجوی بار

در این بخش یک الگوریتم بهینه‌سازی موثر جدید با استفاده از قوانین فیزیکی و مکانیکی فوق الذکر معرفی می‌گردد که CSS نامیده می‌شود. در CSS هر کاندیدای راه حل X_i که شامل تعدادی از متغیرهای تصمیم‌گیری است (به عنوان مثال $X_i = \{x_{ij}\}$) به عنوان یک ذره باردار در نظر گرفته می‌شوند. ذره باردار توسط میدان الکتریکی عناصر دیگر تحت تاثیر قرار می‌گیرند. مقدار نیروی برآیند با استفاده از قوانین الکترواستاتیک و چگونگی حرکت با استفاده از قوانین مکانیک برداری تعیین می‌گردد. به نظر می‌رسد که یک عنصر با نتایج خوب باید نیروی قوی‌تری از عناصری که نتایج آنها بد می‌باشند وارد کند، به گونه‌ای که مقدار بار با در نظر گرفتن مقدار تابع هدف $fit(i)$ تعیین خواهد شد، به منظور معرفی کردن CSS قوانین زیر در نظر گرفته می‌شوند [۲۸-۲۹].

□ قانون ۱

بسیاری از الگوریتم‌های تکاملی طبیعی یک تعداد از راه‌حل‌هایی که از طریق تصادفی انتخاب و نتیجه می‌شوند را حفظ می‌کنند [۳۰]. به طور مشابه CSS تعدادی از ذرات باردار را در نظر می‌گیرد. هر ذره باردار یک بزرگی (مقدار) برای (q_i) دارد و بنابراین یک میدان الکتریکی در فضای اطرافش ایجاد می‌گردد. بزرگی بار با در نظر گرفتن کیفیت راه‌حل آن همان‌گونه که در رابطه (۲-۵۳) آمده است، تعریف می‌شود [۲۸]:

$$q_i = \frac{fit(i) - fit_{worst}}{fit_{best} - fit_{worst}} \rightarrow i = 1, 2, 3, \dots, N \quad (2-53)$$

که در آن fit_{best} و fit_{worst} بدترین و بهترین شایستگی همه ذرات تا این مرحله می‌باشند، $fit(i)$ مقدار تابع هدف یا شایستگی عنصر i را بیان می‌کند و N تعداد کل ذرات باردار می‌باشد. فاصله جدایی r_{ij} بین دو ذره باردار به صورت رابطه (۲-۵۴) تعریف می‌گردد [۲۸].

$$r_{ij} = \frac{\|X_i - X_j\|}{\|(X_i + X_j) / 2 - X_{best}\| + \varepsilon} \quad (54-2)$$

که در آن X_i و X_j موقعیت i امین و j امین ذره باردار می‌باشند، X_{best} موقعیت بهترین ذره باردار موجود است و ε یک عددی کوچک بزرگ تر از صفر، برای اجتناب از یکتایی می‌باشد.

□ قانون ۲

موقعیت ابتدایی ذرات باردار به طور تصادفی در فضای جستجو توسط رابطه (۵۵-۲) تعیین می‌گردد.

$$x_{i,j}^{(0)} = x_{i,\min} + rand.(x_{i,\min} - x_{i,\max}) \rightarrow i = 1, 2, 3, \dots, n \quad (55-2)$$

که در آن $x_{i,j}^{(0)}$ مقدار i امین متغیر را برای j امین ذره باردار تعیین می‌کند، $x_{i,\max}$ و $x_{i,\min}$ حداقل و حداکثر مقدار مجاز برای i امین متغیر می‌باشند، $rand$ یک عدد تصادفی در بازه $[0, 1]$ است و n تعداد متغیرها می‌باشد. سرعت اولیه ذرات باردار صفر در نظر گرفته می‌شود.

$$v_{i,j}^{(0)} = 0 \rightarrow i = 1, 2, 3, \dots, n \quad (56-2)$$

□ قانون ۳

سه شرط می‌تواند در رابطه با نوع نیروهای جاذبه در نظر گرفته شود [۲۸]:

۱. هر ذره باردار می‌تواند بر روی ذره باردار دیگر اثر بگذارد، به عنوان مثال یک ذره باردار بد

می‌تواند بر روی یک ذره باردار خوب اثر بگذارد و بالعکس ($p_{i,j} = 1$)

۲. یک ذره باردار می‌تواند ذره دیگری را جذب کند، اگر مقدار بار الکتریکی آن از دیگری بهتر

باشد. به عبارتی یک ذره باردار خوب یک ذره باردار بد را جذب می‌کند با در نظر گرفتن تابع

احتمال رابطه (۵۷-۲):

$$p_{ij} = \begin{cases} 1 \rightarrow fit(j) > fit(i) \\ 0 \rightarrow else \end{cases} \quad (57-2)$$

۳. همه ذرات باردار خوب می‌توانند همه ذرات باردار بد را جذب کنند و فقط برخی از عناصر

بد می‌توانند عناصر خوب را جذب کنند با در نظر گرفتن تابع احتمال (۵۸-۲):

$$p_{ij} = \begin{cases} 1 \rightarrow \frac{fit(i) - fit_{best}}{fit(j) - fit(i)} > rand \vee fit(j) > fit(i) \\ 0 \rightarrow else \end{cases} \quad (58-2)$$

مطابق شرایط بالا، وقتی یک عنصر خوب یک عنصر بد را جذب می‌کند، قابلیت تبعیت (تشدید)

برای الگوریتم فراهم شده است و بالعکس اگر یک ذره باردار بد یک ذره باردار خوب را جذب کند کاوش (تنوع) فراهم شده است. وقتی یک ذره باردار به سمت یک عنصر خوب می‌رود آن باعث بهبود عملکردش می‌گردد و همچنین اصل سازگاری آن تضمین شده است. حرکت یک ذره باردار خوب به سمت یک ذره باردار بد ممکن است باعث از دست رفتن راه‌حل‌های خوب قبلی و یا افزایش هزینه‌های محاسباتی برای پیدا کردن یک راه‌حل خوب شود. برای حل این مشکل می‌توان یک حافظه بار که بهترین راه حل موجود تاکنون را ذخیره کند در نظر گرفته شود. بنابراین به نظر می‌رسد که شرط سوم از شرایط بالا بهترین قانون می‌باشد، به دلیل این که قابلیت کاوش قوی و بهره‌وری (تبعیت) کارآمد را فراهم می‌نماید.

□ قانون ۴

مقدار نیروی برآیند روی یک ذره باردار با استفاده از معادله (۴۶-۲) تعیین می‌شود که به

صورت کلی توسط رابطه (۵۹-۲) بدست می‌آید:

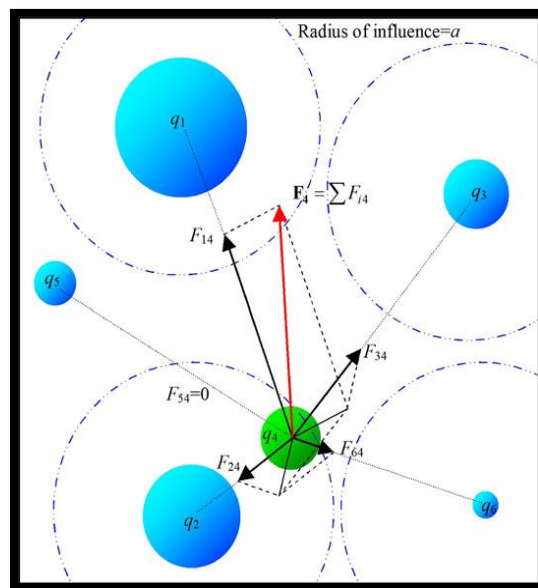
$$F_j = q_j \sum_{i, i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) p_{ij} \alpha_{ij} (X_i - X_j), \begin{cases} j = 1, 2, \dots, N \\ i_1 = 1, i_2 = 0 \Leftrightarrow r_{ij} < a \\ i_1 = 0, i_2 = 1 \Leftrightarrow r_{ij} > a \end{cases} \quad (59-2)$$

در این فرمول α_{ij} پارامتری است که با احتمال k_t نوع نیروی جاذبه و دافعه را مشخص می‌نماید. k_t

یک پارامتر برای کنترل اثر این نیروها می‌باشد. اگر نیرو جاذبه باشد $\alpha_{ij} = 1$ و در غیر این صورت

$\alpha_{ij} = -1$ می گردد. همان طور که در شکل (۲-۱۳) نشان داده شده F_j نیروی برآیند وارد بر j امین

ذره باردار می باشد.



شکل (۲-۱۳) - تعیین برآیند نیروی الکتریکی وارد بر یک ذره باردار

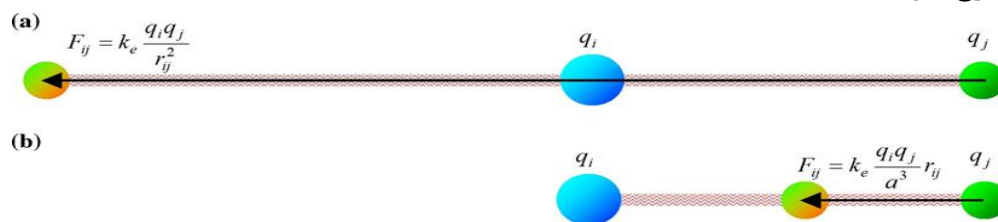
در این الگوریتم هر ذره باردار به عنوان یک گوی باردار با شعاع a در نظر گرفته می شود که چگالی حجمی بار یکنواختی دارد (پخش بار در آن یکنواخت است). در اینجا بزرگی a واحد قرار داده می شود [۲۸]. برای مثال های پیچیده مقدار مناسب برای a باید با در نظر گرفتن فضای جستجو تعیین گردد. معادله زیر را به عنوان یک فرمول کلی می توان استفاده کرد.

$$a = 0.1 * \max(\{x_{i,\max} - x_{i,\min} \mid i = 1, 2, 3, \dots, n\}) \quad (۲-۶۰)$$

مطابق این قانون در تکرارهای اولیه که عناصر دور از یکدیگر می باشند بزرگی نیروی برآیند وارد شده بر یک ذره باردار متناسب با معکوس مربع فاصله جدایی بین ذرات می باشد. بنابر این قدرت کاوش در این شرایط به علت انجام جستجوی بیش تر در تکرار های اولیه زیاد است. آن ضروری است که بهره وری (تبعیت) الگوریتم افزایش یابد و کاوش رفته رفته کاهش پیدا کند، بعد از تعدادی جستجو که ذرات باردار در یک فضای کوچک جمع می شوند و فاصله جدایی بین ذرات باردار کوچک

می شود (مثلا ۰.۱)، نیروی برآیند به جای متناسب بودن با مربع معکوس فاصله جدایی، با فاصله جدایی ذرات متناسب می شود.

مطابق شکل (۲-۱۴) اگر معادله اول $f_{ij} \propto (1/r_{ij}^2)$ برای $r_{ij} = 0.1$ استفاده شود، داریم $(F_{ij} = 100.k_e q_i q_j)$ که در مقایسه با نیرویی که بر ذره باردار در $r_{ij} = 2$ و $(F_{ij} = 0.25.k_e q_i q_j)$ وارد می شود یک مقدار بزرگی است و این نیروی بزرگ باعث می شود که ذرات به جای این که به همدیگر نزدیک تر گردند از یکدیگر دورتر شوند در حالی که معادله دوم $(F_{ij} \propto r_{ij})$ تضمین می کند که همگرایی رخ خواهد داد.



شکل (۲-۱۴) - مقایسه ای بین معادلات (a) $F_{ij} \propto (1/r_{ij}^2)$ و (b) $F_{ij} \propto r_{ij}$ وقتی که $r_{ij} < a$

بنابراین پارامتر a مرحله جستجوی سراسری و مرحله جستجوی محلی را تفکیک می کند. به عنوان مثال وقتی که اکثریت عناصر در یک فضای با شعاع a جمع می گردند جستجوی سراسری تمام می شود و فرآیند بهینه سازی با بهبود نتایج قبلی ادامه می یابد و جستجوی محلی شروع می گردد. از طرف دیگر با استفاده از این قواعد و اصول، تعادل بین کاوش و تبعیت کنترل می شود. باید توجه شود که این قانون گام های رقابتی الگوریتم را در نظر می گیرد. از آنجایی که نیروی برآیند متناسب با بزرگی بار است یک شایستگی بهتر (q_i بزرگتر) می تواند یک نیروی جاذبه قوی تری را ایجاد کند، به گونه ای که تمایل حرکت به سمت یک ذره خوب بیشتر از تمایل حرکت به سمت یک ذره بد می شود [۲۸].

□ قانون ۵

موقعیت و سرعت جدید هر ذره باردار با در نظر گرفتن معادلات (۲-۴۹)، (۲-۵۱) و (۲-۵۲)

تعیین می شود که به صورت کلی در روابط (۲-۶۱) و (۲-۶۲) نشان داده شده است:

$$X_{j,new} = rand_{j_1,k_a} \cdot \frac{F_j}{m_j} \Delta t^2 + rand_{j_2,k_v} V_{j,old} \Delta t + X_{j,old} \quad (۶۱-۲)$$

$$V_{j,new} = \frac{X_{j,new} - X_{j,old}}{\Delta t} \quad (۶۲-۲)$$

که در آن k_a ضریب شتاب، k_v ضریب سرعت برای کنترل اثر سرعت قبلی است، $rand_{j_1}$ و $rand_{j_2}$ دو عدد تصادفی اند که به طور یکنواخت در بازه (۰ و ۱) توزیع شده‌اند. در اینجا m_j جرم ذره باردار j ام است که با q_j برابر است. Δt یک بازه زمانی است که واحد قرار داده می‌شود [۲۸].

اثر سرعت پیشین و نیروی برآیند وارد بر یک ذره باردار بر مبنای مقدار k_v و k_a می‌تواند به ترتیب کاهش یا افزایش یابد. جستجوی بیش از اندازه در تکرارهای اولیه ممکن است قابلیت کاوش را بهبود بخشد، به هر حال مقدار کاوش باید همان گونه که قبلاً شرح داده شده است به تدریج کاهش یابد. از آن جایی که k_a یک پارامتر وابسته به نیروی جاذبه است انتخاب یک مقدار بزرگ برای این پارامتر ممکن است باعث همگرایی سریع شود و بالعکس یک مقدار کوچک می‌تواند زمان محاسبات را افزایش دهد. در حقیقت k_a یک پارامتر کنترلی برای تبعیت (بهره‌وری) است. انتخاب یک تابع که سیر صعودی دارد می‌تواند عملکرد الگوریتم را بهبود بخشد. علاوه بر این، جهت سرعت قبلی برای یک ذره باردار لزوماً هم جهت نیروی برآیند نیست بنابراین آن نتیجه می‌شود که ضریب سرعت k_v فرآیند کاوش را کنترل می‌کند و می‌تواند یک تابع کاهنده انتخاب شود، از این رو k_v و k_a به توسط روابط (۶۳-۲) و (۶۴-۲) تعریف می‌شوند [۲۸]:

$$k_v = 0.5(1 - \frac{iter}{iter_{max}}) \quad (۶۳-۲)$$

$$k_a = 0.5(1 + \frac{iter}{iter_{max}}) \quad (۶۴-۲)$$

که $iter$ تعداد تکرار واقعی می‌باشد، $iter_{max}$ حداکثر تعداد تکرار می‌باشد با این معادلات k_v به صورت خطی به سمت صفر کاهش می‌یابد در حالی که k_a وقتی که تعداد تکرارها بالا می‌رود به سمت یک افزایش می‌یابد. با این شیوه تعادل بین کاوش و نرخ سرعت همگرایی ذخیره می‌شود. با در

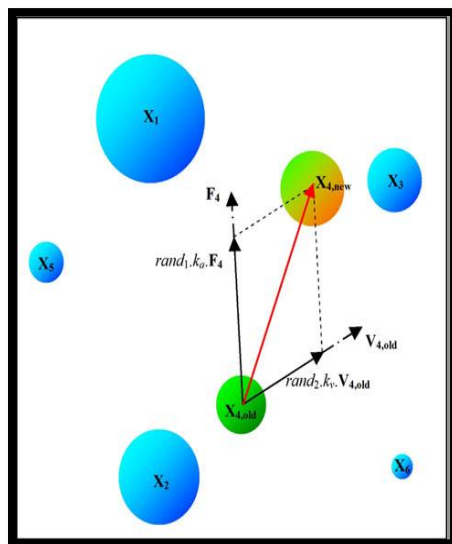
نظر گرفتن این پارامترها معادلات (۶۱-۲) و (۶۲-۲) می‌توانند به صورت کلی به شکل روابط (۶۵-۲) و (۶۶-۲) نوشته شوند:

$$X_{j,new} = 0.5rand_{j_1} \cdot (1 + \frac{iter}{iter_{max}}) \cdot \sum_{i, i \neq j} (\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2) p_{ij} (X_i - X_j) \quad (۶۵-۲)$$

$$+ 0.5 \cdot rand_{j_2} \cdot (1 - \frac{iter}{iter_{max}}) V_{j,old} + X_{j,old}$$

$$V_{j,new} = X_{j,new} - X_{j,old} \quad (۶۶-۲)$$

شکل (۱۵-۲) حرکت یک ذره باردار به مکان جدیدش را با استفاده از این قاعده نشان می‌دهد. قوانین ۵ و ۶ مراحل همکاری ذرات باردار را فراهم می‌کند.



شکل (۱۵-۲) - حرکت ذره باردار به موقعیت جدید

□ قانون ۶

در نظر گرفتن حافظه که بردارهای بهترین ذره باردار و مقادیر توابع هدف وابسته به آن‌ها را ذخیره کند می‌تواند عملکرد الگوریتم را بدون افزایش هزینه‌های محاسباتی بهبود بخشد. برای انجام این هدف حافظه بار^۱ CM استفاده می‌شود تا تعدادی از بهترین راه‌حل‌های موجود را ذخیره کند. اندازه CM (CMS)، $\frac{N}{4}$ (تعداد ذرات باردار) گرفته شده است [۲۸]. مزایای دیگر CM شامل استفاده از این حافظه برای راهنمایی ذرات باردار می‌باشد، به عبارتی دیگر بردارهای ذخیره شده در

^۱ ChargeMemory

CM می‌توانند ذرات باردار فعلی را بر طبق معادله (۲-۵۹) جذب کنند. در صورتی که فرض می‌شود تعدادی مشابه از بدترین ذرات موجود نمی‌توانند بقیه را جذب کنند.

□ قانون ۷

دو مشکل اساسی در رابطه با بسیاری از الگوریتم‌های تکاملی وجود دارد اولین مشکل تعادل بین کاوش و تبعیت (بهره‌وری) در شروع، در طول و درانتهای جستجو می‌باشد و مشکل دوم چگونگی سروکار داشتن با عواملی است که محدوده متغیرها را نقض می‌کنند.

اولین مشکل به طور طبیعی از طریق کاربرد قوانین شرح داده شده بالا حل می‌شود. به هر حال به منظور حل مشکل دوم یکی از ساده‌ترین روش‌ها استفاده از مقادیر نزدیک مرز برای متغیرهای نقض شده است و دیگر اینکه شخص می‌تواند ذرات نقض شده را مجبور کند تا به موقعیت اولیه قبلی خودشان باز گردند یا شخص می‌تواند مقادیر ماکزیمم سرعت را کاهش دهد تا به ذرات کمتری اجازه دهد تا حدود متغیرها را نقض کنند. با اینکه این روش‌ها ساده می‌باشند آنها به اندازه کافی موثر نیستند و ممکن است منجر به کاهش کاوش در فضای جستجو شوند. این مشکل از قبل مورد توجه قرار گرفته و با استفاده از روش‌های جابه‌جایی مبتنی بر Harmony search حل شده است. طبق این مکانیزم هر مولفه‌ای از بردارهای راه‌حل که محدوده متغیرها را نقض کند می‌تواند دوباره از طریق حافظه بار احیا گردد [۳۱-۳۲]. انتخاب یک مقدار جدید برای آن مولفه طبق حالت زیر انجام می‌گیرد:

۱. با احتمال CMCR یک مقدار از حافظه بار انتخاب می‌گردد. CMCR (نرخ در نظرگیری

حافظه بار) که بین صفر و یک متغیر است.

۲. با احتمال (CMCR-۱) یک مقدار به صورت تصادفی از محدوده مجاز متغیرها برای مولفه مذکور انتخاب می‌گردد.

۳. در صورتی که یک مقدار از CM (حافظه بار) برای مولفه مورد نظر انتخاب گردد با احتمال

PAR فرآیند تعدیل گام‌ها انجام می‌شود، که PAR نرخ انتخاب یک مقدار از همسایگی بهترین

ذرات باردار قرار داده می‌شود.

□ قانون ۸

معیار خاتمه در این روش یکی از شرایط زیر است:

- ♦ تعداد حداکثر تکرار: فرآیند بهینه‌سازی بعد از یک مقدار ثابتی از تکرار پایان می‌یابد برای مثال ۱۰۰۰ تکرار.
- ♦ تعداد تکرارها بدون بهبود: فرآیند بهینه‌سازی بعد از این که تعداد ثابتی از تکرار بدون هیچ بهبودی انجام گردد پایان می‌یابد.
- ♦ خطا در تابع هدف حداقل: اختلاف بین بهترین تابع هدف و بهینه کلی از یک حد مورد انتظار کمتر باشد.
- ♦ اختلاف بین بهترین و بدترین ذره باردار: فرآیند بهینه‌سازی متوقف می‌گردد اگر اختلاف بین مقادیر هدف بهترین و بدترین ذره باردار از یک دقت مشخصی کمتر گردد.
- ♦ حداکثر فاصله بین ذرات باردار: حداکثر فاصله بین ذرات باردار از یک مقدار از پیش مشخص شده کمتر باشد.

۲-۶-۳ مختصری از مراحل الگوریتم جستجوی ذرات باردار

با توجه به گام‌های ذکر شده در بالا می‌توان الگوریتم بهینه‌سازی جستجوی ذرات باردار را بطور

مختصر شرح داد:

۲-۶-۳-۱ مرحله مقدار دهی اولیه

□ گام ۱

مقدار دهی اولیه: به پارامترهای الگوریتم CSS مقدار اولیه داده می‌شود، یک رشته از ذرات باردار با موقعیت‌های تصادفی و سرعت‌های وابسته به آنها (قوانین ۱ و ۲) مقدار دهی می‌شوند.

□ گام ۲

رتبه بندی ذره باردار: مقدار شایستگی ذرات باردار بررسی و با یکدیگر مقایسه می‌گردد و به طور افزایشی طبقه‌بندی می‌شوند.

■ گام ۳

ایجاد حافظه بار (CM)، به تعداد اندازه حافظه بار ذرات باردار ابتدایی با مقادیر تابع هدف وابسته به خودشان در حافظه بار ذخیره می‌گردند.

۲-۶-۳-۲ مرحله جستجو

■ گام ۱

تعیین نیروی جاذبه: تعیین احتمال حرکت هر ذره باردار به سوی دیگر ذرات (قانون ۳) و محاسبه بردار نیروی جاذبه هر ذره باردار (قانون ۴).

■ گام ۲

ساختار حل: حرکت هر ذره باردار به موقعیت جدید و پیدا کردن سرعت آنها (قانون ۵)

■ گام ۳

اصلاح موقعیت ذرات باردار: اگر هر ذره باردار از فضای جستجو خارج شود، موقعیت آن با استفاده از قانون ۷ اصلاح می‌گردد.

■ گام ۴

رتبه‌بندی ذرات: مقادیر تابع هدف برای ذرات باردار جدید را بررسی کرده و با هم مقایسه شوند و آنها به طور افزایشی دسته‌بندی می‌گردند.

■ گام ۵

بهنگام کردن حافظه بار: اگر برخی از بردارهای ذرات باردار از آنهایی که در حافظه بار بدترین می‌باشند بهتر باشند بردارهای بهتر را در حافظه بار قرار می‌دهیم و ذرات باردار بد از حافظه خارج می‌شوند (قانون ۶).

۲-۶-۳-۳ مرحله کنترل معیارهای خاتمه

گام‌های مرحله جستجو را تا زمانی که معیار پایان بخش (خاتمه) مشخص شود (قانون ۸) تکرار می‌شود.

فصل سوم

هیندیالی سطح مقطع سازه ها با

الکوریتم های فرا استگاری

۱-۳ مقدمه

یکی از هنرهای مهندسی بدست آوردن بیشترین بازده، با انتخاب یک سیستم مناسب و صرف کمترین مصالح است. در مهندسی سازه، وظیفه سیستم های سازه تحمل بارهای مؤثر و معمولاً انتقال نیرو به تکیه گاه ها است. این سیستم ها، علاوه بر آن که باید ایمن و کارآ باشند، در عین حال بایستی از کمترین هزینه و به نوعی از کمترین مصرف مصالح برخوردار باشند [۱۱].

در راه نیل به امر بهینه سازی سازه ها چهار دسته بندی عمده وجود دارد [۴]:

- ۱- بهینه یابی مقطع^۱ که شامل انتخاب سطح مقطع بهینه اعضا است.
 - ۲- بهینه یابی شکل یا هندسه^۲ که در آن مختصات بهینه گرهی به ازای نحوه قرارگیری مفروض اعضا تعیین می شود.
 - ۳- بهینه یابی توپولوژی^۳ که مشتمل بر تعیین بهترین آرایش اعضا در فضای مفروض مختصات گرهی است.
 - ۴- بهینه یابی پیکربندی که مشتمل بر سه مورد قبلی است.
- در بهینه یابی مقطع، طرح بهینه به ازای کمترین مقدار سطح مقطع با مختصات ثابت گرهی (هندسه ثابت) و همبندی ثابت سازه (توپولوژی ثابت) انجام می پذیرد [۳۳]. معمولاً در مسائل کاربردی مهندسی از مقاطع استاندارد موجود در بازار استفاده می شود. در این راستا بهینه یابی مقاطع منفصل^۴، با اتخاذ اعضا از لیست پروفیل های موجود، توسعه و گسترش یافت [۱۴]. در این پایان نامه نیز در بهینه یابی مقاطع از مقاطع موجود در بازار استفاده شده است.

^۱ . Sizing Optimization

^۲ . Geometrical (Shape) Optimization

^۳ . Topological Optimization

^۴ . Discrete Variable Optimization

در بهینه یابی شکل یا هندسه، مختصات گرهی سازه تحت یک توپولوژی (همبندی) مشخص جهت طراحی بهینه بررسی می شود.

در این پایان نامه، بهینه یابی سطح مقطع سازه ها با کمک روش های نوین و فراابتکاری، بررسی شده است. در این فصل، در ابتدا به طور اجمالی به شرح مبانی بهینه یابی سطح مقطع سازه توسط الگوریتم های فرا ابتکاری پرداخته شده است.

۲-۳ فرمولبندی مسأله بهینه یابی مقاطع بر اساس الگوریتم های فراابتکاری

در این پایان نامه بهینه یابی مقاطع سازه ها بر اساس الگوریتم های فرا ابتکاری، زمانیکه متغیرهای طراحی منفصل بوده و می بایست از لیست مقاطع موجود^۱ انتخاب شوند، مورد توجه و بررسی قرار گرفته است. فرمولبندی مسأله بشرح زیر تعریف می شود:

یافتن سطح مقطع مناسب برای هر عضو:

$$A = [a_1, a_2, \dots, a_{Nos}]^T \quad ; \quad a_i \in S \quad ; \quad i = 1, \dots, Nos \quad (۱-۳)$$

کمینه سازی تابع هدف $f(A)$ که در معرض قیود زیر می باشد:

$$G1 = \begin{cases} g_{i1}(A) = 0 & \text{if } \frac{\sigma_{iT}}{\sigma_{all(Ten)}} - 1 \leq 0 \quad ; \quad i = 1, \dots, Net \\ g_{i1}(A) = \frac{\sigma_{iT}}{\sigma_{all(Ten)}} - 1 & \text{if } \frac{\sigma_{iT}}{\sigma_{all(Ten)}} - 1 > 0 \quad ; \quad i = 1, \dots, Net \end{cases} \quad (۲-۳)$$

$$G2 = \begin{cases} g_{i2}(A) = 0 & \text{if } \frac{\sigma_{iC}}{\sigma_{all(Com)}} - 1 \leq 0 \quad ; \quad i = 1, \dots, Nec \\ g_{i2}(A) = \frac{\sigma_{iC}}{\sigma_{all(Com)}} - 1 & \text{if } \frac{\sigma_{iC}}{\sigma_{all(Com)}} - 1 > 0 \quad ; \quad i = 1, \dots, Nec \end{cases} \quad (۳-۳)$$

^۱ . Available Section List

$$G3 = \begin{cases} g_{i3}(A) = 0 & \text{if } \frac{\lambda_i}{\lambda_{ten}} - 1 \leq 0 ; i = 1, \dots, Net \\ g_{i3}(A) = \frac{\lambda_i}{\lambda_{ten}} - 1 & \text{if } \frac{\lambda_i}{\lambda_{ten}} - 1 > 0 ; i = 1, \dots, Net \end{cases} \quad (4-3)$$

$$G4 = \begin{cases} g_{i4}(A) = 0 & \text{if } \frac{\lambda_i}{\lambda_{com}} - 1 \leq 0 ; i = 1, \dots, Nec \\ g_{i4}(A) = \frac{\lambda_i}{\lambda_{com}} - 1 & \text{if } \frac{\lambda_i}{\lambda_{com}} - 1 > 0 ; i = 1, \dots, Nec \end{cases} \quad (5-3)$$

$$G5 = \begin{cases} g_{i5}(A) = 0 & \text{if } \left| \frac{\Delta_i}{\Delta_i^{all}} \right| - 1 \leq 0 ; i = 1, \dots, N dof \\ g_{i5}(A) = \left| \frac{\Delta_i}{\Delta_i^{all}} \right| - 1 & \text{if } \left| \frac{\Delta_i}{\Delta_i^{all}} \right| - 1 > 0 ; i = 1, \dots, N dof \end{cases} \quad (6-3)$$

هر یک از موارد فوق بشرح زیر قابل تفسیر است

۱-۲-۳ متغیرهای طراحی

رابطه (۱-۳) بردار متغیرهای سطح مقطع را نشان می دهد. در این رابطه

❖ A : حاوی متغیرهای طراحی سطح مقطع می باشد.

❖ S : مجموعه پروفیل های موجود است، که بر اساس آن طراحی تشکیل می گردد.

❖ Nos : تعداد اعضای مجموعه S است. به عبارت دیگر تعداد مقاطع موجود که برای

طراحی استفاده می شود.

❖ a_i : سطح مقطع هر عضو از سازه است، که می تواند هر مقدار از مجموعه S را اختیار

نماید.

۲-۲-۳ تابع هدف

در طراحی هر نوع سیستم کاربردی علاوه بر مسائل فنی و ضوابط طراحی، پارامترهای اقتصادی

نیز دارای اهمیت خاصی می باشند. محاسبه طرحی که کمترین هزینه اجرایی را داشته باشد، با استفاده از الگوریتم های مختلفی امکان پذیر است. این الگوریتم ها، تابعی از پارامترهای اقتصادی مسئله را به عنوان تابع هدف انتخاب می کنند و با محاسبه حداقل و حداکثر آن (با توجه به شرایط مساله) و ارضا کردن ضوابط طراحی مساله گزینه مناسب را بدست می آورند. در نتیجه سیستم حاصل علاوه بر دارا بودن استانداردهای طراحی کمترین هزینه اجرایی را نیز خواهد داشت.

از آنجایی که هدف کمینه نمودن وزن سازه به ازای بهترین سطح مقطع است؛ بنابراین می توان تابع هدف را بشرح زیر تعریف نمود:

$$f(A) = \sum_{i=1}^{Ne} (\rho_i \ell_i a_i) \quad (7-3)$$

که در آن

❖ Ne : تعداد اعضای سازه است.

❖ ρ_i : وزن مخصوص مصالح¹ هر عضو از سازه است که در اغلب اوقات برای تمامی اعضا یکسان فرض می شود.

❖ ℓ_i : طول هر عضو از سازه است.

❖ a_i : سطح مقطع هر عضو از سازه است.

حال اگر هدف کمینه نمودن حجم سازه مورد نظر باشد؛ آنگاه تابع هدف بشرح زیر تعریف

می شود:

$$f(A) = \sum_{i=1}^{Ne} (\ell_i a_i) \quad (8-3)$$

که در آن تعاریف ℓ_i ، a_i و Ne مشابه رابطه (7-3) تعریف می شود.

۳-۲-۳ قیود G1 و G2

برای هر سازه ای، همواره می بایست تنش ناشی از بارگذاری در تمامی اعضا، در محدوده مجاز و قابل قبول آئین نامه ها قرار داشته باشد. این مقدار مجاز براساس جنس مصالح اعضای به کار رفته در

¹ . Density

سازه تعیین می‌شود. بنابراین در روند بهینه یابی مقاطع از طریق الگوریتم های فراابتکاری، پس از تحلیل سازه، تنش تک تک اعضا محاسبه می شود. در صورت نقض قیود بر اساس روابط (۲-۳) و (۳-۳) جریمه ای به طرح های مربوطه اعمال می شود. باید توجه داشت که رابطه (۲-۳) مقدار نقض قیود ناشی از تنش کششی اعضای سازه و رابطه (۳-۳) مقدار نقض قیود ناشی از مقدار تنش فشاری اعضای سازه را محاسبه می نماید. به عبارت دیگر براساس اینکه عضو فشاری یا کششی است، نقض قیود آن توسط رابطه مربوطه کنترل و در صورت لزوم، مقدار آن محاسبه می شود. پارامترهای روابط (۲-۳) و (۳-۳) به شرح ذیل تعریف می شود:

❖ $g_{i1}(A)$: میزان جریمه نقض قید تنش کششی اعضای سازه

❖ σ_{iT} : تنش کششی عضو i ام

❖ $\sigma_{all(Ten)}$: تنش مجاز و قابل قبول کششی

❖ Net : تعداد اعضای کششی سازه به ازای هندسه و توپولوژی موجود

❖ $g_{i2}(A)$: میزان جریمه نقض قید تنش فشاری اعضای سازه

❖ σ_{iC} : تنش فشاری عضو i ام

❖ $\sigma_{all(Com)}$: تنش مجاز و قابل قبول فشاری

❖ Nec : تعداد اعضای فشاری سازه به ازای هندسه و توپولوژی موجود

تنش کششی مجاز، بر اساس مبحث دهم مقررات ملی ساختمان ایران عبارت است از: [۳۴]

$$\sigma_{all(Ten)} = 0.6F_y \quad (۹-۳)$$

❖ F_y : تنش جاری شدن مصالح می باشد.

تنش فشاری مجاز مطابق با آئین نامه فولاد ایران از روابط زیر به دست می‌آید. [۳۴]

$$C_c = \sqrt{\frac{2\pi^2 E}{F_y}} \quad (۱۰-۳)$$

❖ C_c : لاغری مرزی

❖ E : مدول الاستیسیته^۱ مصالح مصرفی می باشد.

if $\lambda > C_c$

$$F_s = 1.67 + 0.375 \left(\frac{\lambda}{C_c} \right) - 0.125 \left(\frac{\lambda}{C_c} \right)^3 \quad (۱۱-۳)$$

$$\sigma_{all(com)} = \frac{1}{F_s} \left[1 - \frac{1}{2} \left(\frac{\lambda}{C_c} \right)^2 \right] \cdot F_y (13-4)$$

if $\lambda < C_c$

$$\sigma_{all(com)} = \frac{12\pi^2 E}{23 \cdot \lambda^2} \quad (۱۲-۳)$$

❖ λ : لاغری عضو

❖ F_s : ضریب اطمینان

۳-۲-۴ قیود G3 و G4

علاوه بر قید تنش، محدودیت لاغری قید بعدی است که باید در نظر گرفته شود. مطابق با

آئین نامه فولاد ایران برای اعضای کششی این مقدار برابر ۳۰۰ می باشد و برای اعضای تحت فشار

لاغری حداکثر نباید از ۲۰۰ تجاوز کند [۳۴]. در این روابط:

❖ $g_{i3}(A)$: مقدار جریمه نقض قید لاغری اعضای کششی

❖ $g_{i4}(A)$: مقدار جریمه نقض قید لاغری اعضای فشاری

❖ λ_i : لاغری عضو i ام

$\lambda_{ten}=300$

❖ λ_{ten} : لاغری مجاز اعضای کششی

¹ . Elasticity Modulus

❖ λ_{com} : لاغری مجاز اعضای فشاری

$$\lambda_{com}=200$$

۳-۲-۵ قیود G5

همان طور که بیان شد، هدف از این پایان نامه بهینه‌یابی با فرض توپولوژی و هندسه ثابت است. در این راستا همواره کنترل مقدار تغییر مکان گره های سازه، امری حیاتی و ضروری است. برای این منظور رابطه (۳-۶) مقدار تغییر مکان کلیه گره های سازه را کنترل می نماید. به عبارت دیگر پس از تحلیل سازه و محاسبه مقادیر تنش، میزان تغییر مکان گره های سازه محاسبه می شود. در صورتی که تغییر مکان درجه آزادی i ام، در محدوده مجاز باشد، جریمه ای به طرح مورد نظر تعلق نمی گیرد. اما در غیر این صورت، طرح مربوطه متناسب با میزان تخطی جریمه خواهد شد. مقدار این جریمه، بر اساس روابط (۳-۶) بدست می آید. در این روابط:

$$\text{❖ } g_{i5}(A) : \text{مقدار جریمه نقض قید تغییر مکان گره های فعال سازه}$$

$$\text{❖ } \Delta_i : \text{مقدار تغییر مکان درجه آزادی } i \text{ام}$$

$$\text{❖ } NoJ : \text{تعداد گره های سازه}^1$$

$$\text{❖ } \Delta_i^{all} : \text{مقدار تغییر مکان مجاز و قابل قبول درجه آزادی } i \text{ام}$$

باید توجه داشت که قیود G1 تا G5 در روابط (۳-۲)، (۳-۳)، (۳-۴)، (۳-۵) و (۳-۶) بصورت قیود نرمال شده^۲ بیان شده اند. این کار نتایج خوبی را به دنبال خواهد داشت زیرا به لحاظ کمی، میزان نقض تنش بسیار متفاوت با نقض تغییر مکان است و نرمال کردن قیود سبب می شود ارزش کمی یکسانی به آن ها اختصاص یابد [۱۱].

¹ . Number Of Joint

² . Normalized Constraints

فصل چهارم:

روش پیشنهادی و راهنمای برنامه

۴-۱ مقدمه

الگوریتم های فراابتکاری روش های جستجوی الهام گرفته شده از طبیعت می باشند که با کمک نقاط مختلف و ایجاد طرح های گوناگون، فضای طراحی را کاوش می نمایند. بنابراین نیاز به تعداد دفعات تکرار زیاد جهت رسیدن به جواب بهینه، سبب افزایش حجم محاسبات عددی در فرآیند بهینه یابی می شود.

از سوی دیگر عدم شناخت درست از پارامترها و روابط حاکم بر الگوریتم های فرا ابتکاری از قبیل تعداد اعضای جمعیت، تعداد نسل ها^۱ و سایر پارامتر های دخیل در آنها، سبب می شود تا در برخی مواقع این الگوریتم ها، بهینه نسبی را بدست آورند.

امروزه با پیشرفت رایانه ها در زمینه نرم افزار و سخت افزار، مشکل سرعت عملیات بهینه یابی از طریق الگوریتم های فراابتکاری، تا حدود زیادی حل شده است؛ اما تنظیم پارامترها و شناسایی صحیح روابط حاکم بر این الگوریتم ها از جمله مشکلات این الگوریتم ها می باشد که افراد مختلفی با ارائه روش های گوناگون سعی در حل این مشکل نموده اند.

در این پایان نامه با استفاده از الگوریتم های ژنتیک، اجتماع مورچگان، جستجوی هارمونی، جستجوی ذرات باردار و ژنتیک جزیره ای و با استعانت از مطالب مرجع [۱۲] روشی ترکیبی و نوین که «روش جستجوی چند منظوره مبتنی بر الگوریتم های فراابتکاری» نامیده شده است جهت بهینه یابی مقاطع سازه های خرپایی ارائه گردیده است.

^۱ . No Of Generation

در روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری، جمعیت به بخش های مختلف (جزایر متنوع) تقسیم می شود. هر بخش^۱ دارای محیطی^۲ متفاوت و ساختاری منحصر بفرد است و از الگوریتم خاصی در هر جزیره استفاده شده است. ویژگی ها و ساختار هر بخش بر اساس آزمودن عملگر ها و پارامترهای سایر الگوریتم ها و انتخاب بهترین ها، شکل گرفته است. به عبارت دیگر برای تشکیل جزایر^۳، کلیه شیوه ها و عملگرهای مشروح در فصل دوم به کمک زبان برنامه نویسی ویژوال بیسیک^۴ کد شده و بر اساس نتایج بهترین ها انتخاب و به کار گرفته شده است [۳۵]. این طریقه عمل سبب می شود که جواب ها تنوع خوبی داشته باشند. در پایان تعداد نسل سازی های معین براساس وقفه مهاجرت و طبق نرخ مهاجرت بهترین ها در مجموعه جزایر جابه جا می شوند تا در نهایت الگوریتم به جواب های مناسبی برسد.

در مسایلی که به دلیل زیاد بودن متغیرهای طراحی و تعداد قیود به کار گرفته شده و بزرگ بودن فضای کاوش، تأثیر پارامترهای سایر الگوریتم ها نقشی اساسی ایفا می نمایند و استفاده از این الگوریتم ها منتهی به جواب بهینه نمی شود، مدل پیشنهادی به طور مؤثر و هوشمندی فضای طراحی را مورد کاوش و در نهایت بهترین جواب های ممکن را ارائه می دهد.

در ادامه ابتدا روش جستجوی چند منظوره مبتنی بر الگوریتم ژنتیک (ژنتیک جزیره ای) به طور مختصر بیان شده [۱۲] و سپس به طور کامل و دقیق به شرح روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری می پردازیم.

^۱ . Part

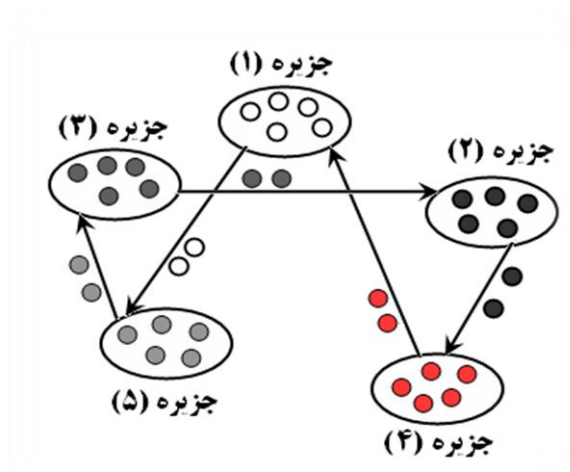
^۲ . Environment

^۳ . Islands

^۴ . Visual Basic

۴-۲ روش جستجوی چند منظوره مبتنی بر الگوریتم ژنتیک (ژنتیک جزیره ای)

در این روش یک جمعیت بزرگ به تعدادی زیر جمعیت^۱ کوچکتر تقسیم می‌شود و الگوریتم ژنتیک مرسوم با عملگرها و پارامترهای متفاوت، برای هر کدام از زیر جمعیت‌ها به طور جداگانه به کار گرفته می‌شود. هر کدام از این جمعیت‌ها به عنوان یک جزیره نامیده می‌شود. تعدادی از بهترین کروموزوم‌های هر جزیره پس از طی چند نسل مشخص، بطور تصادفی به جزایر دیگر منتقل می‌شود. این مطلب در شکل (۴-۱) نشان داده شده است.



شکل (۴-۱) - روش جستجوی چند منظوره

در فرآیند انتقال دو پارامتر بشرح زیر تعریف می‌شود:

۱. وقفه مهاجرت^۲: بیانگر تعداد نسل‌ها بین هر انتقال می‌باشد.
۲. نرخ مهاجرت^۳: بیانگر درصد اعضای منتخب برای انتقال از هر جزیره در زمان انتقال می‌باشد.

^۱ . Subpopulations

^۲ . Migration Interval

^۳ . Migration Rate

نحوه مهاجرت انتخاب شده به صورت یک حلقه با مقصدهای تصادفی^۱ می باشد که هر کدام از زیرجمعیت‌ها دارای یک مقصد تصادفی می‌باشند و این مقصدها به طور تصادفی در هر دوره تناوب مهاجرت^۲ مشخص می‌شوند.

در روش جستجوی چند منظوره مبتنی بر الگوریتم ژنتیک به دلیل خصوصیات مجزای فضای هر جزیره، می‌توان از تمامی روش‌های معتبر و کارآمد معرفی شده توسط محققین بهره جست. به عبارت دیگر هر مسئله بهینه‌یابی در یک لحظه با چندین روش بررسی و فضای طراحی کاوش می‌شود و در ادامه بهترین نتایج، فی مابین دیگر جزایر به اشتراک گذاشته و جمعیت‌های جدیدی در اختیار هر جزیره جهت پرورش قرار می‌گیرد. این خصوصیات بطور دسته جمعی سبب کاهش تأثیر پارامترها و روابط حاکم بر عملیات GA می‌شود و سرعت همگرایی الگوریتم را بطور شگرفی افزایش می‌دهد.

لازم به ذکر است که در الگوریتم فوق فرآیندهای انتخاب در هر جزیره متفاوت با جزیره‌های دیگر می‌باشد. برای این منظور از بهترین فرآیندهای انتخاب الگوریتم ژنتیک که در فصل دوم شرح داده شده است در هر جزیره استفاده شده است. در انتخاب فرآیند ادغام، نیز کلیه عملگرهای GA با زبان برنامه نویسی ویژوال بیسک بررسی و کارآمدترین آن‌ها بر اساس نتایج در جزایر به کار گرفته شده‌اند.

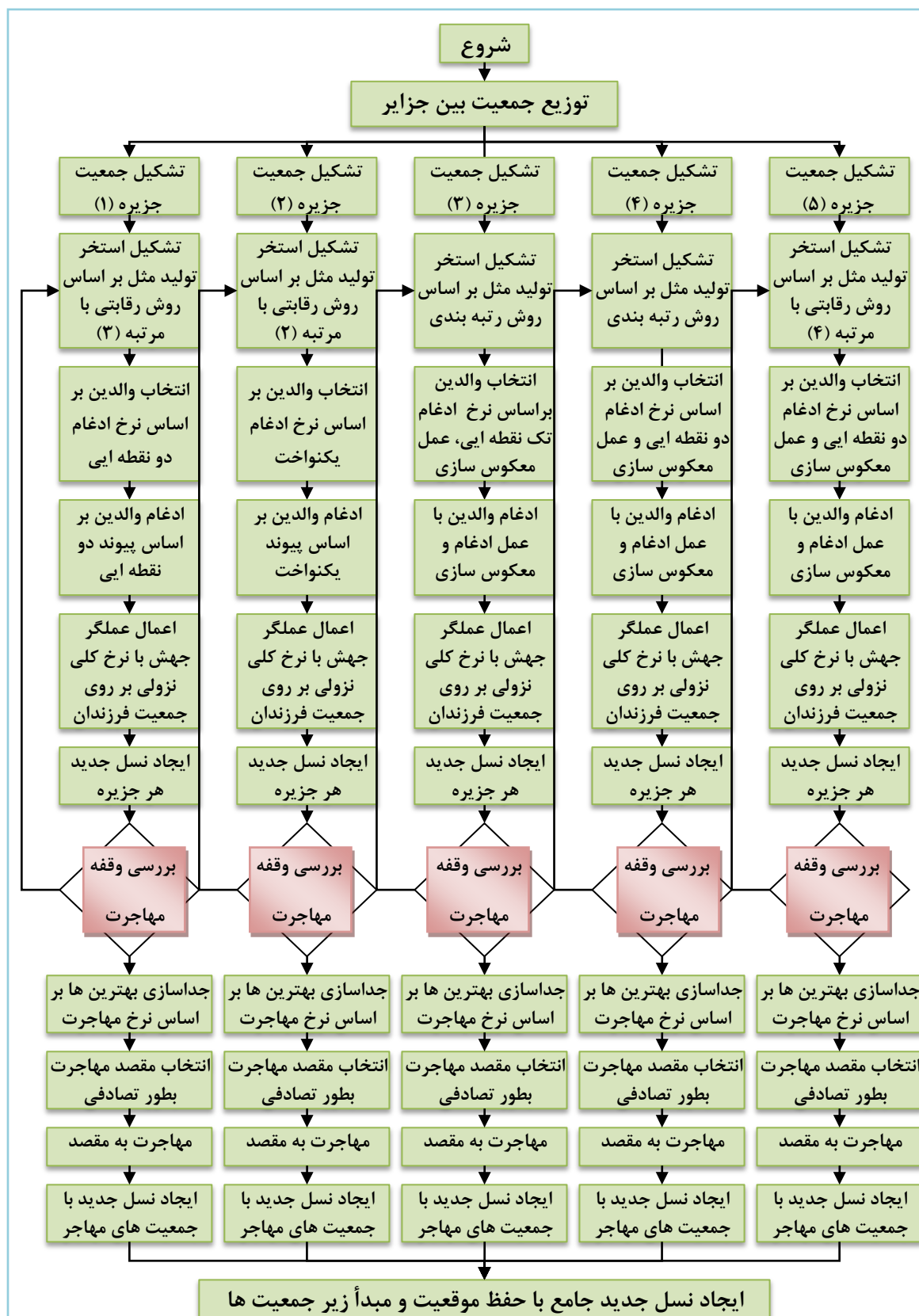
برای این منظور فلوچارت الگوریتم جستجوی چند منظوره مبتنی بر ژنتیک که در این رساله استفاده گردیده است در شکل (۴-۲) نشان داده شده است. کلیه مراحل فلوچارت، در ادامه به تفصیل مورد بحث و بررسی قرار گرفته است.

^۱ . Random Destinations

^۲ . Migration Period



شکل (۲-۴) - فلوچارت الگوریتم جستجوی چند منظوره مبتنی بر ژنتیک برای بهینه یابی مقاطع سازه ها



شکل (۳-۴) - فلوجارت روش جستجوی چند منظوره مبتنی بر ژنتیک

۴-۲-۱ الگوریتم تولید جمعیت اولیه

در این پایان نامه برای نمایش متغیرهای طراحی سطح مقطع، از شیوه کدگذاری دودویی استفاده شده است. بنابراین در زیربرنامه تولید جمعیت اولیه، با تولید تصادفی زیر رشته‌ها به تعداد متغیرهای طراحی سطح مقطع به این مقصود دست می‌یابیم. شیوه‌ی دیگری که در تولید زیر رشته سطح مقطع به کار گرفته می‌شود تولید متغیر سطح مقطع به صورت عدد تصادفی در محدوده صفر تا Nos و تبدیل آن به یک عدد دودویی (رمز کردن^۱) است [۱۲]. در بهینه‌یابی سطح مقطع سازه‌ها، متغیرهای طراحی سطح مقطع عرضی، اعضای گروه‌بندی شده می‌باشند که بایستی از لیست پروفیل‌های موجود یعنی از مجموعه $S = \{A_1, A_2, \dots, A_{Nos}\}$ انتخاب گردند.

۴-۲-۲ رمز گشایی^۲ هر کروموزوم

همان طور که قبلاً بیان شد؛ از جمله ویژگی‌های یک شیوه کدگذاری مناسب، قابلیت تبدیل یک به یک بین عناصر کدشده و کد نشده است. به عبارت دیگر می‌بایست، هر نقطه در فضای جستجو قابل تبدیل به یک رشته باشد. در شیوه نمایش دودویی این ویژگی، بارز و مبرهن است. در این پایان نامه با استعانت از روابط زیر، تبدیل متغیرهای طراحی سطح مقطع به رشته‌های دودویی به راحتی قابل دستیابی است.

هر بیت قابلیت نمایش دو مقدار صفر و یک را دارد. بنابراین در ابتدا طول یک زیررشته بر اساس تعداد اعضای لیست مقاطع بشرح زیر تعیین می‌گردد:

$$2^{NB} \geq Nos \quad (۱-۴)$$

^۱ . Encoding

^۲ . Decoding

در رابطه فوق مجهول NB تعداد بیت ها یا کاراکترهای یک زیررشته می باشد و Nos تعداد اعضای موجود در لیست مقاطع است که مقداری معلوم می باشد.

کوچکترین مقدار NB که به ازای آن رابطه (۴-۱) برقرار باشد، به عنوان طول یک زیررشته منظور می شود. به عبارت دیگر اولین عددی که به ازای آن نامساوی رابطه (۴-۱) برقرار می شود، طول هر زیررشته است.

اما نکته حائز اهمیت، آن است که هر زیررشته سطح مقطع، می بایست بیانگر شماره مقطع ای از لیست پروفیل های موجود باشد. به عبارت دیگر، هر یک از اعضای لیست مقاطع باید هم ارز با یک زیررشته باشد.

با توجه به اینکه یک زیررشته NB بیتی اعداد ۰ تا $2^{NB} - 1$ را کد می کند، مقدار متناظر زیررشته i ام بشرح زیر قابل محاسبه است:

$$IR_i = \sum_{j=0}^{NB-1} 2^j b_j \quad (۴-۲)$$

که هر بیت b می تواند اعداد صفر یا یک را اختیار کند و به صورت رشته زیر نمایش داده می شود:

$$b_{NB-1} \dots b_2 b_1 b_0 \quad (۴-۳)$$

در رابطه (۴-۲) IR_i ، عدد صحیح متناظر با زیررشته i ام می باشد که در محدوده صفر تا $2^{NB} - 1$ قرار دارد.

حال برای آنکه IR_i به شماره مقطع قرار گرفته در لیست پروفیل های S یعنی IS_i ارتباط داده شود، از رابطه کلی زیر استفاده می شود:

$$IS_i = Int \left(IR_i * \frac{(Nos - 1)}{(2^{NB} - 1)} \right) \quad (۴-۴)$$

$$1 \leq IS_i \leq Nos \quad ; \quad 0 \leq IR_i \leq 2^{NB} - 1 \quad \text{که}$$

در رابطه (۴-۴) تابع $Int()$ ، قسمت اعشاری عدد درون پرانتز را حذف و عدد صحیح باقیمانده را ارائه می‌دهد.

براساس رابطه (۴-۴) رشته‌هایی که با هیچکدام از متغیرهای طراحی هم‌ارز نمی‌باشند، ترمیم شده و متناظر با یکی از اعضای مجموعه S قرار می‌گیرند. البته باید توجه داشت که اینکار ممکن است سبب شود برخی از مقاطع لیست پروفیل‌های موجود، با احتمال بیشتری برای متغیر طراحی سطح مقطع انتخاب شوند.

۳-۲-۴ ماتریس سختی هر کروموزوم

طول اعضا متناسب با موقعیت گره‌های ابتدایی و انتهایی توسط رابطه زیر محاسبه می‌گردد:

$$\ell = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (۵-۴)$$

از سوی دیگر در این مرحله، سطح مقطع کلیه اعضای توپولوژی حاضر متناسب با زیررشته مربوطه و بنابر اطلاعات مرحله قبل (رمز گشایی زیررشته‌ها) مشخص می‌گردد. در نهایت کلیه پارامترهای لازم جهت تشکیل ماتریس سختی سازه^۱ یعنی پارامترهای $\frac{EA}{L}$ در این مرحله از الگوریتم محاسبه می‌شوند.

۴-۲-۴ تحلیل سازه مربوط به هر کروموزوم

در این تحقیق، جهت کسب نیروهای داخلی و محاسبه تغییر مکان گره‌های فعال سازه از عملیات تحلیل ماتریسی استفاده شده است. برای این منظور با تشکیل ماتریس سختی برای هر کروموزوم به

^۱ . Structure Stiffness Matrix

آنالیز سازه پرداخته می‌شود. در طی این روند، حل یک دستگاه n معادله با n مجهول نیاز است. برای نیل به این هدف از شیوه حل دستگاه به روش چولسکی^۱ استفاده شده است. استفاده از این شیوه سبب افزایش سرعت عملیات بهینه یابی می‌شود. با حل دستگاه معادلات، مقادیر تغییر مکان گره‌های سازه بدست می‌آید. در ادامه پس از محاسبه تغییر مکان گرهی نوبت به محاسبه مقادیر نیروهای داخلی اعضا می‌رسد تا با استفاده از آن‌ها و سطح مقطع هر عضو، مقدار تنش در اعضای سازه محاسبه شود.

در نهایت امر پس از محاسبه مقادیر تنش‌ها و تغییر مکان گره‌های فعال سازه نوبت به مرحله بعدی الگوریتم، یعنی محاسبه تابع جریمه و تابع هدف می‌رسد.

۴-۲-۵ تابع هدف و تابع جریمه مورد استفاده

پس از تحلیل کروموزوم‌ها به جهت کسب نیروهای داخلی و تغییر مکان‌های گرهی، نوبت به محاسبه میزان جریمه هر کروموزوم متناسب با نوع و میزان تخطی آن می‌رسد. همان‌طور که در فصول پیشین اشاره شد، برای بررسی میزان شایستگی یک کروموزوم، تابعی جهت محاسبه مقدار نقض قیود نیاز است. در راستای نیل به این هدف، در این پایان نامه با استعانت از روابط (۲-۹) و (۲-۱۰) تابع جریمه ای براساس پیشنهاد راجیو و کریشنا مورتی بشرح زیر تعریف می‌گردد:

$$f_i^{Penalty} = f_i(A) * K * C_i \quad (۴-۶)$$

که در آن $f_i^{Penalty}$ مقدار تابع جریمه کروموزوم i ام است و C_i ، بیانگر جمع کل نقض‌های صورت گرفته توسط سازه مرتبط با کروموزوم i ام می‌باشد و بصورت زیر تعریف می‌شود:

$$C_i = \sum_{j=1}^{Ne} \sum_{k=1}^3 g_{jk}(A) + \sum_{n=1}^{NOJ} g_{n4}(A) \quad ; \quad i = 1, \dots, nc \quad , \quad n = 1, \dots, NOJ \quad (۴-۷)$$

^۱ . Cholesky Method

در رابطه فوق nc ، تعداد کروموزوم های موجود در یک جمعیت (pop-size) و NOJ تعداد گره های سازه می باشد.

پس از محاسبه مقدار تابع جریمه متناظر با هر کروموزوم، مقدار تابع هدف مربوط به هر کروموزوم بشرح زیر محاسبه می شود:

$$\left\{ \begin{array}{ll} f_i(A) = \sum_{i=1}^{Ne} (\rho_i \ell_i a_i) & \text{if Object Function Based On Weight Structure} \\ f_i(A) = \sum_{i=1}^{Ne} (\ell_i a_i) & \text{if Object Function Based On Volume Structure} \end{array} \right. \quad (۸-۴)$$

حال با داشتن مقادیر تابع هدف و تابع جریمه متناظر با هر کروموزوم، به راحتی می توان مقدار تابع شایستگی را برای هر رشته حساب نمود .

۶-۲-۴ تابع هدف اصلاح شده و تابع شایستگی مورد استفاده

در این تحقیق جهت محاسبه شایستگی هر کروموزوم با استعانت از روابط (۱۲-۲) و (۱۳-۲) تابع شایستگی بشرح ذیل انتخاب شده است.

$$F_i = [\phi_{Max}(A) + \phi_{Min}(A)] - \phi_i(A) \quad ; \quad i = 1, \dots, nc \quad (۹-۴)$$

$$\phi_i(A) = f_i(A) + f_i^{Penalty} \quad ; \quad i = 1, \dots, nc \quad (۱۰-۴)$$

در روابط فوق همان طور که پیشتر بیان شد؛ $f_i(A)$ ، $f_i^{Penalty}$ ، $\phi_i(A)$ و F_i به ترتیب تابع هدف، تابع جریمه، تابع هدف اصلاح شده و شایستگی کروموزوم i ام در جمعیت جاری است. $\phi_{Min}(A)$ و $\phi_{Max}(A)$ نیز به ترتیب مقادیر حداکثر و حداقل تابع هدف اصلاح شده در جمعیت حاضر می باشد.

براساس روابط (۹-۴) و (۱۰-۴) به طرح هایی با نقض قیود بالا یا $\phi_i(A)$ بالا، میزان شایستگی کم و برعکس به طرح های خوب با مقدار تابع هدف اصلاح شده کم ، شایستگی بالا اختصاص می دهد.

۷-۲-۴ مشخصات جزایر در روش جستجوی چند منظوره مبتنی بر ژنتیک

همان طور که در بخش ۲-۴ شرح داده شد؛ در روش جستجوی چند منظوره مبتنی بر الگوریتم ژنتیک می بایست جمعیت GA، به تعدادی زیر جمعیت تبدیل شود. بنابراین در الگوریتم جستجوی چند منظوره مبتنی بر ژنتیک جهت نیل به این هدف، پنج جزیره برای طی عملیات بهینه یابی سازه های خرابایی در نظر گرفته شده است. برای این منظور، جمعیت GA توسط کاربر برنامه به صورت مضربی از پنج داده می شود.

پس از تشکیل جمعیت در همه جزایر، هر جزیره بر اساس اسلوب پیشنهادی، پرورش کروموزوم ها را سرلوحه کار خود قرار می دهد. شیوه هر جزیره با جزایر دیگر اندکی متفاوت است. انتخاب هر یک از متدهای موجود در جزایر، بر اساس آزمودن کلیه روش ها و عملگرهای دنیای GA صورت گرفته است. این روش ها و عملگرها در فصل دوم به تفصیل مورد بحث قرار گرفته است و در جزایر الگوریتم جستجوی چند منظوره به کار گرفته شده است. شرح تفصیلی روش و عملگرهای هر یک از جزایر در ذیل آمده است.

۷-۲-۴-۱ جزیره (۱)

با توجه به متدها و عملگرهای به کار برده شده در جزیره (۱) مطابق شکل (۴-۳)، برای وارد نمودن اطلاعات این جزیره از پنجره Island (1) استفاده می شود. شکل (۴-۴) پنجره Island (1) را نشان می دهد.

Island (1)

Data Of Island (1)

The Number Population Selection For Tournament: 3

Data Of Parameters GA

Maximum Of Rate Cross Over (RC_Max): 0.85

Minimum Of Rate Cross Over (RC_Min): 0.7

Maximum Of Rate Mutation (PM_Max): 0.008

Minimum Of Rate Mutation (PM_Min): 0.003

Ok Cancel

شکل (۴-۴) - پنجره ورودی اطلاعات جزیره (۱)

پنجره (1) Island با علامت دار نمودن گزینه^۱ مربوطه در پنجره Data Parameters Of GA ظاهر می شود. همان طور که در شکل (۴-۴) مشاهده می شود، براساس این پنجره اطلاعاتی همچون نرخ پیوند و نرخ جهش به همراه مرتبه (سایز رقابت) روش رقابتی توسط کاربر^۲ به برنامه داده می شود. سایز رقابت در جزیره (۱) همواره برابر سه در نظر گرفته می شود؛ اما این موضوع قابل تغییر است و می توان متناسب با شرایط مسئله آن را تغییر داد. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok، اطلاعات در متغیرهای جزیره (۱) ذخیره می شوند.

^۱ . Option

^۲ . User

۲-۷-۲-۴ جزیره (۲)

با توجه به متد ها و عملگرهای به کار برده شده در جزیره (۲) مطابق شکل (۳-۴)، برای وارد نمودن اطلاعات این جزیره از پنجره Island (2) استفاده می شود. شکل (۵-۴) پنجره Island (2) را نشان می دهد.

The screenshot shows a software window titled "Island2". Inside, there is a section titled "Data Of Island (2)". Below this, there is a field for "The Number Population Selection For Tournament:" with a value of 3. Another section is titled "Data Of Parameters GA". It contains four fields: "Maximum Of Rate Cross Over (RC_Max):" with value 0.9, "Minimum Of Rate Cross Over (RC_Min):" with value 0.8, "Maximum Of Rate Mutation (PM_Max):" with value 0.01, and "Minimum Of Rate Mutation (PM_Min):" with value 0.005. At the bottom of the dialog are "Ok" and "Cancel" buttons.

Parameter	Value
The Number Population Selection For Tournament:	3
Maximum Of Rate Cross Over (RC_Max):	0.9
Minimum Of Rate Cross Over (RC_Min):	0.8
Maximum Of Rate Mutation (PM_Max):	0.01
Minimum Of Rate Mutation (PM_Min):	0.005

شکل (۵-۴) - پنجره ورودی اطلاعات جزیره (۲)

پنجره Island (2) با علامت دار نمودن گزینه مربوطه در پنجره Data Parameters Of GA ظاهر می شود. همان طور که در شکل (۵-۴) مشاهده می شود، براساس این پنجره اطلاعاتی همچون نرخ پیوند و مرتبه (سایز رقابت) روش رقابتی به همراه مقادیر حداکثر و حداقل نرخ جهش توسط کاربر به برنامه داده می شود. سایز رقابت در جزیره (۲) همواره برابر دو در نظر گرفته می شود؛ اما این موضوع قابل تغییر است و می توان متناسب با شرایط مسئله آن را تغییر داد. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok، اطلاعات در متغیرهای جزیره (۲) ذخیره می شوند.

۳-۷-۲-۴ جزیره (۳)

با توجه به متد ها و عملگرهای به کار برده شده در جزیره (۳) در شکل (۳-۴)، برای وارد نمودن اطلاعات این جزیره (۳) از پنجره Island (3) استفاده می شود. شکل (۴-۶) پنجره Island (3) را نشان می دهد.

Island3

Data Of Island (3)

Data Of Parameters GA

Maximum Of Rate Cross Over (RC_Max): 0.85

Minimum Of Rate Cross Over (RC_Min): 0.8

Maximum Of Rate Mutation (PM_Max): 0.009

Minimum Of Rate Mutation (PM_Min): 0.005

Ok Cancel

شکل (۴-۶) - پنجره ورودی اطلاعات جزیره (۳)

پنجره Island (3) با علامت دار نمودن گزینه مربوطه در پنجره Data Parameters Of GA ظاهر می شود. همان طور که در شکل (۴-۶) مشاهده می شود، براساس این پنجره اطلاعاتی همچون نرخ پیوند و معکوس سازی به همراه مقادیر حداکثر و حداقل نرخ جهش توسط کاربر به برنامه داده می شود. لازم به ذکر است که عمل معکوس سازی در جزیره (۳)، براساس معکوس سازی بیت ها پی ریزی شده است. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok، اطلاعات در متغیرهای جزیره (۳) ذخیره می شوند.

۴-۲-۷-۴ جزیره (۴)

با توجه به متد ها و عملگرهای به کار برده شده در جزیره (۴) در شکل (۴-۳)، برای وارد نمودن اطلاعات این جزیره از پنجره Island (4) استفاده می شود. شکل (۴-۷) پنجره Island (4) را نشان می دهد.

Island4

Data Of Island (4)

Data Of Parameters GA

Maximum Of Rate Cross Over (RC_Max): 0.85

Minimum Of Rate Cross Over (RC_Min): 0.75

Maximum Of Rate Mutation (PM_Max): 0.005

Minimum Of Rate Mutation (PM_Min): 0.001

Ok Cancel

شکل (۴-۷) - پنجره ورودی اطلاعات جزیره (۴)

پنجره Island (4) با علامت دار نمودن گزینه مربوطه در پنجره Data Parameters Of GA ظاهر می شود. همان طور که در شکل (۴-۷) مشاهده می شود، براساس این پنجره اطلاعاتی همچون نرخ پیوند و معکوس سازی به همراه مقادیر حداکثر و حداقل نرخ جهش توسط کاربر به برنامه داده می شود. لازم به ذکر است که عمل معکوس سازی در جزیره (۴)، براساس معکوس سازی بیت ها پی ریزی شده است. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok، اطلاعات در متغیرهای جزیره (۴) ذخیره می شوند.

۴-۲-۵ جزیره (۵)

با توجه به متد ها و عملگرهای به کار برده شده در جزیره (۵) در شکل (۴-۳)، برای وارد نمودن اطلاعات این جزیره از پنجره Island (5) استفاده می شود. شکل (۴-۸) پنجره Island (5) را نشان می دهد.

The screenshot shows a dialog box titled "Island5". It contains two main sections: "Data Of Island (5)" and "Data Of Parameters GA".

Data Of Island (5):

- The Number Population Selection For Tournament: 4

Data Of Parameters GA:

- Maximum Of Rate Cross Over (RC_Max): 0.85
- Minimum Of Rate Cross Over (RC_Min): 0.7
- Maximum Of Rate Mutation (PM_Max): 0.005
- Minimum Of Rate Mutation (PM_Min): 0.001

At the bottom, there are "Ok" and "Cancel" buttons.

شکل (۴-۸) - پنجره ورودی اطلاعات جزیره (۵)

پنجره Island (5) با علامت دار نمودن گزینه مربوطه در پنجره Data Parameters Of GA ظاهر می شود. همان طور که در شکل (۴-۸) مشاهده می شود، براساس این پنجره اطلاعاتی همچون، نرخ پیوند و مرتبه (سایز رقابت) روش رقابتی به همراه مقادیر حداکثر و حداقل نرخ جهش توسط کاربر به برنامه داده می شود. سایز رقابت در جزیره (۵) همواره برابر چهار در نظر گرفته می شود؛ اما این موضوع قابل تغییر است و می توان متناسب با شرایط مسئله آن را تغییر داد. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok، اطلاعات در متغیرهای جزیره (۵) ذخیره می شوند.

در تمامی جزایر مطروحه پنجره Data Parameters Of GA، به عنوان پنجره مبنا، جهت ورود اطلاعات GA و فراخوانی جزایر معرفی شده است. این پنجره در شکل (۹-۴) نشان داده شده است. براساس این پنجره اطلاعاتی همچون میزان تکرار^۱ و نسل سازی به عنوان ورودی‌های عمومی^۲ GA توسط کاربر اعلام می گردند. وقفه مهاجرت نیز از جمله اطلاعاتی می باشد که می‌بایست توسط کاربر اعلام گردد. این اطلاعات مختص الگوریتم جستجوی چند منظوره مبتنی بر ژنتیک می باشند.

شکل (۹-۴) - پنجره ورودی اطلاعات GA و فراخوانی جزایر

در سوی دیگر در پنجره GA، همان طور که بارز است با علامت‌دار نمودن هر یک از گزینه‌های مربوط به جزایر، جزیره مربوطه فراخوانی شده و ظاهر می گردد. در نهایت امر، با فشردن دکمه فرمان OK اطلاعات ورودی در حافظه برنامه ذخیره می شوند.

^۱ . Iteration

^۲ . Public

۳-۴ روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری

درسازه هایی که فضای طراحی بزرگ و تعداد قیود طراحی زیاد باشد تعادل مناسب بین مولفه های تنوع و تشدید (کاوش و تبعیت) در حصول جواب بهینه از اهمیت بالایی برخوردار می باشد، بدین منظور در روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری از الگوریتم های ژنتیک، اجتماع مورچگان، جستجوی هارمونی و جستجوی ذرات باردار در ۴ جزیره مستقل و مجزا استفاده گردیده تا با استفاده از این روش ها فضای طراحی به طور مناسب و هوشمند مورد کاوش قرار گرفته و تعادل مناسب بین مولفه های تنوع و تشدید (کاوش و تبعیت) ایجاد شده و جواب های نزدیک به جواب بهینه مطلق بدست آید. در این روش هر یک از جزایر ۴ گانه با توجه به الگوریتم های به کار گرفته شده در آن ها در طی چند نسل متوالی به کاوش فضای طراحی می پردازند و طرح های مناسب و مختلفی ایجاد می نمایند. بعد از طی چند نسل و در هنگام وقفه مهاجرت، بر اساس نرخ مهاجرت تعدادی از بهترین طرح های هر جزیره برای عمل مهاجرت انتخاب می شوند. در این روش عمل مهاجرت بر اساس رعایت قوانین زیر صورت می گیرد:

(۱) هر یک از جزایر دارای مقصد تصادفی می باشند و در هنگام وقفه مهاجرت مشخص می شوند.

(۲) معیار مقایسه برای طرح های هر جزیره با سایر جزایر باید یکسان باشد تا بر اساس این معیار عمل مهاجرت صورت پذیرد.

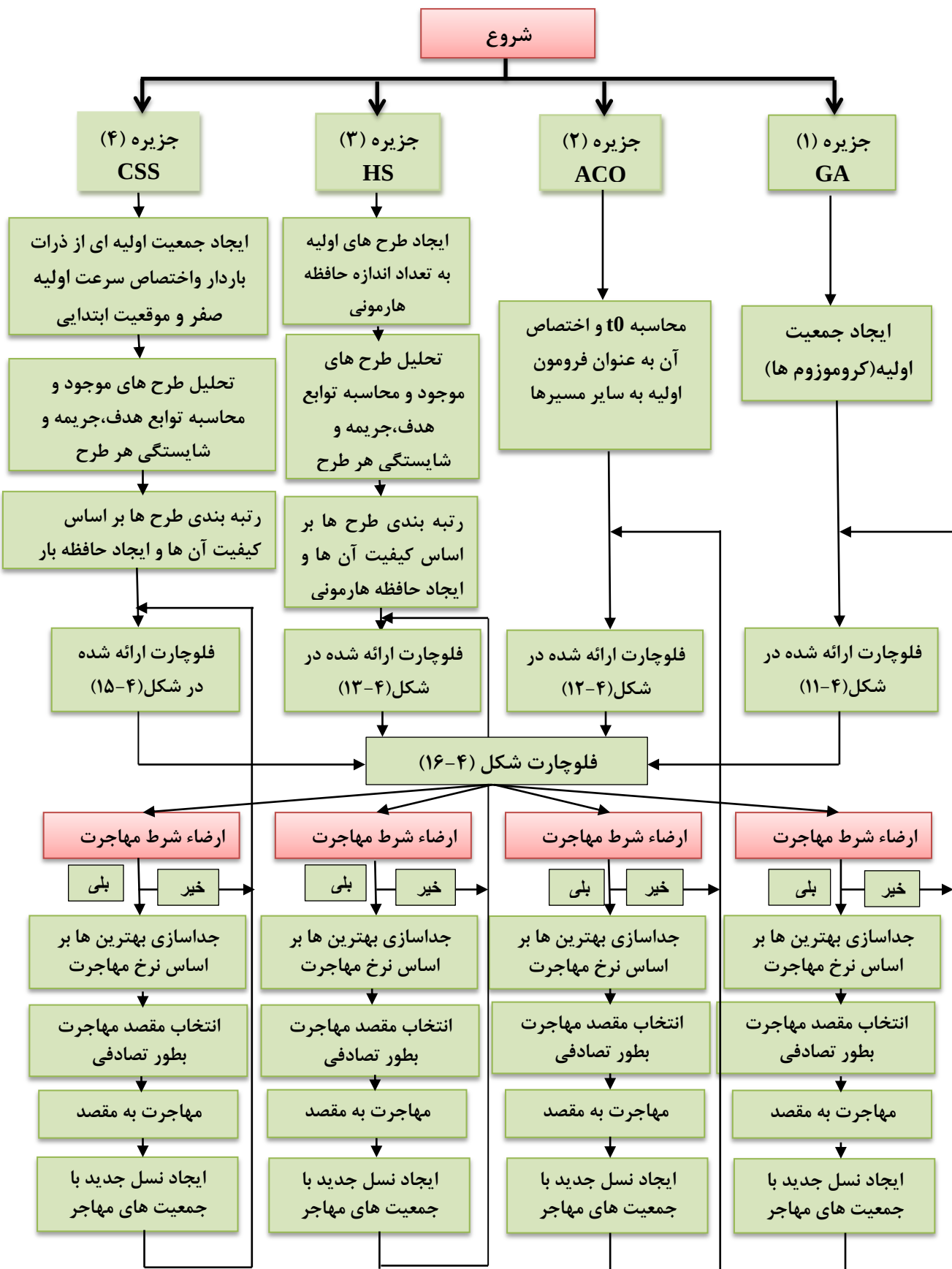
(۳) طرح های مهاجر باید با طرح های جزیره مقصد سازگاری داشته باشد و به صورت الگوی به

کار گرفته شده برای طرح های جزیره مقصد درآیند تا جمعیت نهایی هر جزیره بعد از عمل

مهاجرت بر اساس متد آن جزیره شکل گیرد.

فلوچارت مربوط به روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری در شکل

(۴-۱۰) ارائه شده است.



شکل (۴-۱۰) - فلو چارت روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری



شکل (۴-۱۱) - فلو چارت مربوط به جزیره ۱



شکل (۴-۱۲) - فلوچارت مربوط به جزیره ۲

در تابع احتمال گذر رابطه (۲-۳۰) پارامتر اطلاعات کاوشی η_{ij} به صورت زیر تعریف می گردد:

$$\eta_{ij} = \frac{1}{A_j L_i} \quad (۴-۱۱)$$

که A_j سطح مقطع j ام مربوط به گروه i ام و L_i طول مربوط به اعضای گروه i می باشد.

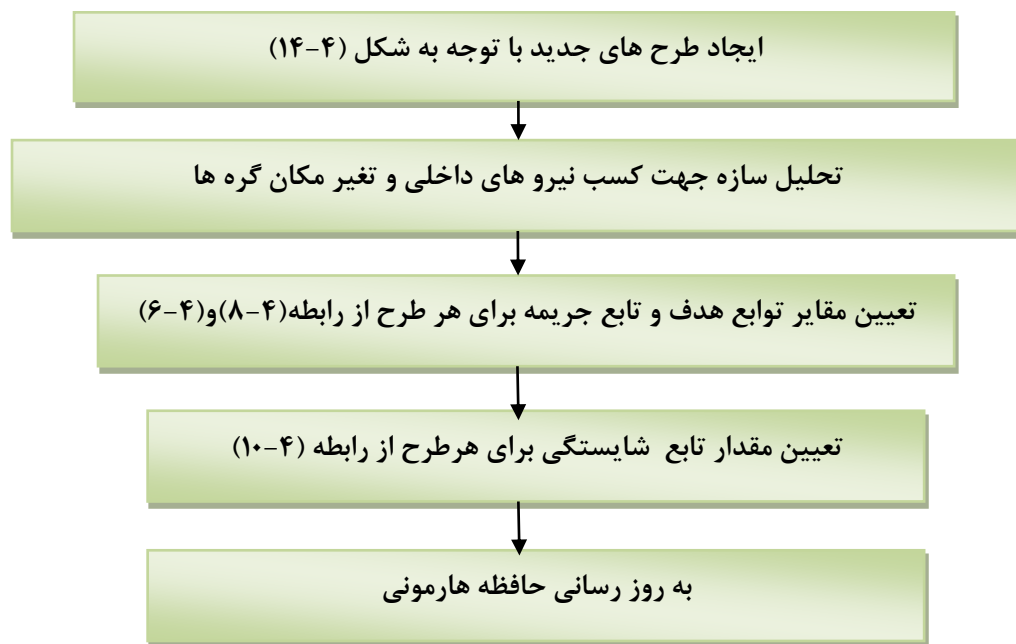
همچنین به روز رسانی محلی فرومون مسیرهای انتخاب شده توسط رابطه زیر انجام می گردد:

$$\tau_{ij}(t+1) = \xi \cdot \tau_{ij} \quad (۴-۱۲)$$

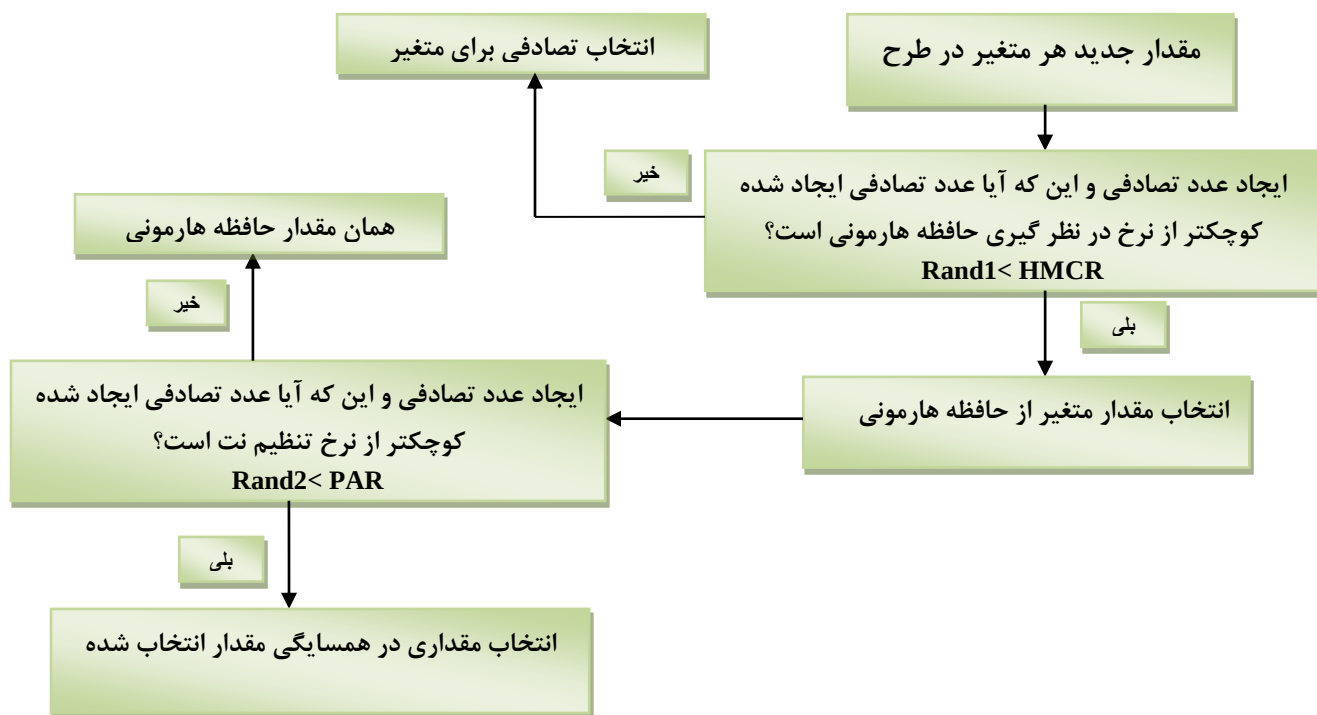
که در آن ξ پارامتر مربوط به تبخیر محلی می باشد.

در برجای گذاری فرومون با استفاده از روابط (۲-۳۵) تا (۲-۳۷)، c^{bs} و c^u وزن (تابع هدف) مربوط

به طرح های ایجاد شده توسط مورچه کلی و مورچه با رتبه μ می باشد.



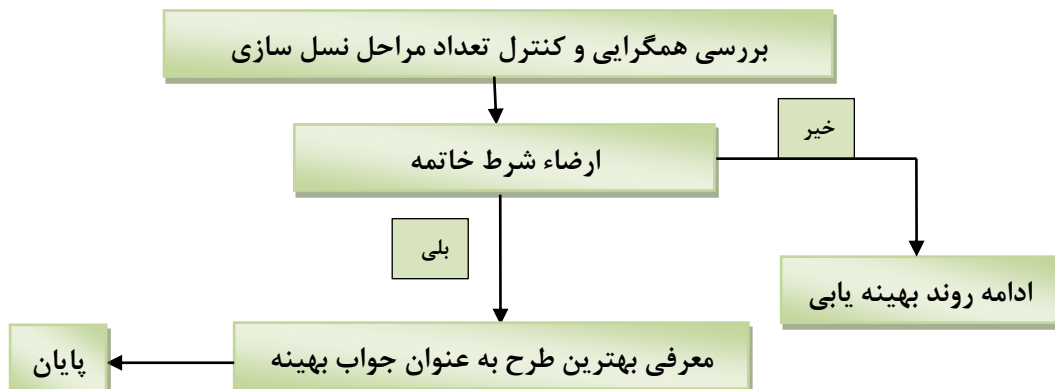
شکل (۴-۱۳) - فلوچارت مربوط به جزیره ۳



شکل (۴-۱۴) - انتخاب مقدار برای هر متغیر تصمیم در الگوریتم جستجو هارمونی



شکل (۴-۱۵) - فلوچارت مربوط به جزیره ۴



شکل (۴-۱۶) - بررسی معیار خاتمه

۳-۱-۴ معیار مقایسه طرح ها در هنگام مهاجرت

زمانی که هر یک از جزیره ها بعد از طی چند نسل به وقفه مهاجرت می رسند بهترین طرح های این جزایر با انتخاب مقصد خود به صورت تصادفی باید با طرح های نامناسب جزیره مقصد مقایسه گردند و زمانی که از لحاظ کیفیت بهتر از آن طرح ها بودند جایگزین آن ها شوند. چون از چند الگوریتم مختلف در جزایر استفاده شده است باید معیاری یکسان برای مقایسه طرح های این جزایر اعمال گردد تا بر این اساس اقدام به مهاجرت نمایند. در اینجا ما می دانیم طرحی که دارای کم ترین میزان نقض و یا به عبارتی دارای تابع هدف اصلاح شده کم تری می باشد از لحاظ کیفی طرح مناسب تری است. پس معیار مقایسه کیفیت طرح ها در روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری بر اساس رابطه زیر تعریف می گردد:

$$\varphi_i(A) = f_i(A) * (1 + K * C_i) \quad i = 1, \dots, nk \quad (13-4)$$

که nk تعداد طرح های هر جزیره و C_i میزان نقض قیود طرح می باشد.

۲-۳-۴ سازگار شدن طرح های مهاجر با طرح های جزایر مقصد

در هنگام وقفه مهاجرت بر اساس مقصد انتخاب شده خواهیم داشت:

(۱) اگر مقصد جزیره (۱) باشد چون در این جزیره الگوریتم ژنتیک به پرورش طرح ها (کروموزوم ها) می پردازد باید طرح های مهاجر به صورت کروموزوم هایی هم ارز معادل گردند.

(۲) اگر مقصد جزیره (۲) باشد چون در این جزیره الگوریتم اجتماع مورچگان بر اساس فرومون های موجود در مسیر به ایجاد طرح های مناسب می پردازد. ابتدا باید مسیرهای مربوط (سطح مقطع هر گروه) در طرح های مهاجر مشخص گردیده و بر مبنای میزان کیفیت این طرح ها فرومون های مسیر های یکسان در جزیره مقصد افزایش یابد. برای افزایش فرومون های مسیر از فرمول پیشنهادی زیر استفاده گردیده است.

$$\tau_{ij}(t+1) = \tau_{ij}(t) + (1-\rho) \cdot \frac{\tau_{ij}(t)}{2\tau_0} \cdot \frac{\varphi_i(\text{best-Aco})}{\varphi_i(A)} \cdot (\text{iter})^{0.05} \quad (14-4)$$

که در آن $\tau_{ij}(t)$ فرومون موجود روی مسیر ij قبل از افزایش فرومون، ρ نرخ تبخیر فرومون، τ_0 فرومون اولیه اختصاص داده شده به همه مسیرها می باشند که در فصل دوم تشریح شده اند. $\varphi_i(\text{best-Aco})$ میزان تابع هدف اصلاح شده بهترین طرح در تکرار جاری جزیره ACO و $\varphi_i(A)$ میزان تابع هدف اصلاح شده طرح مهاجر می باشد.

همچنین می دانیم در تکرار های بالاتر طرح های مناسب دارای فرومون بیشتری می گردند و این طرح ها دارای کیفیت مناسبی می باشند. بدین منظور در معادله (4-11) اثر تکرار نیز لحاظ شده است.

۳) اگر مقصد جزایر (۳) و (۴) باشند آنگاه طرح های مهاجر جایگزین طرح های نامناسب حافظه هارمونی و حافظه بار می گردند.

۳-۳-۴ بررسی همگرایی و کنترل تعداد مراحل نسل سازی

روش های متنوعی جهت بررسی شرط خاتمه در الگوریتم های فرا ابتکاری وجود دارد. در این پایان نامه، شرط خاتمه الگوریتم با کنترل تعداد مراحل نسل سازی اقناع می شود. به عبارت دیگر شرط همگرایی، سپری شدن تعداد مشخصی از نسل ها که توسط کاربر مشخص می گردد؛ تعیین شده است. تعیین تعداد تکرار (تعداد نسل های تولید شده) جهت خاتمه الگوریتم مقداری اختیاری بوده و به تعداد اعضای سازه موردنظر جهت بهینه یابی بستگی دارد. زمانی که شرط خاتمه (تکمیل نسل سازی) اقناع شود، الگوریتم جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری بهترین طرح نسل حاضر (آخرین نسل) را به عنوان طرح بهینه معرفی می نماید و جهت اعلام مقاطع بهینه به زیربرنامه چاپ خروجی^۱ می فرستد.

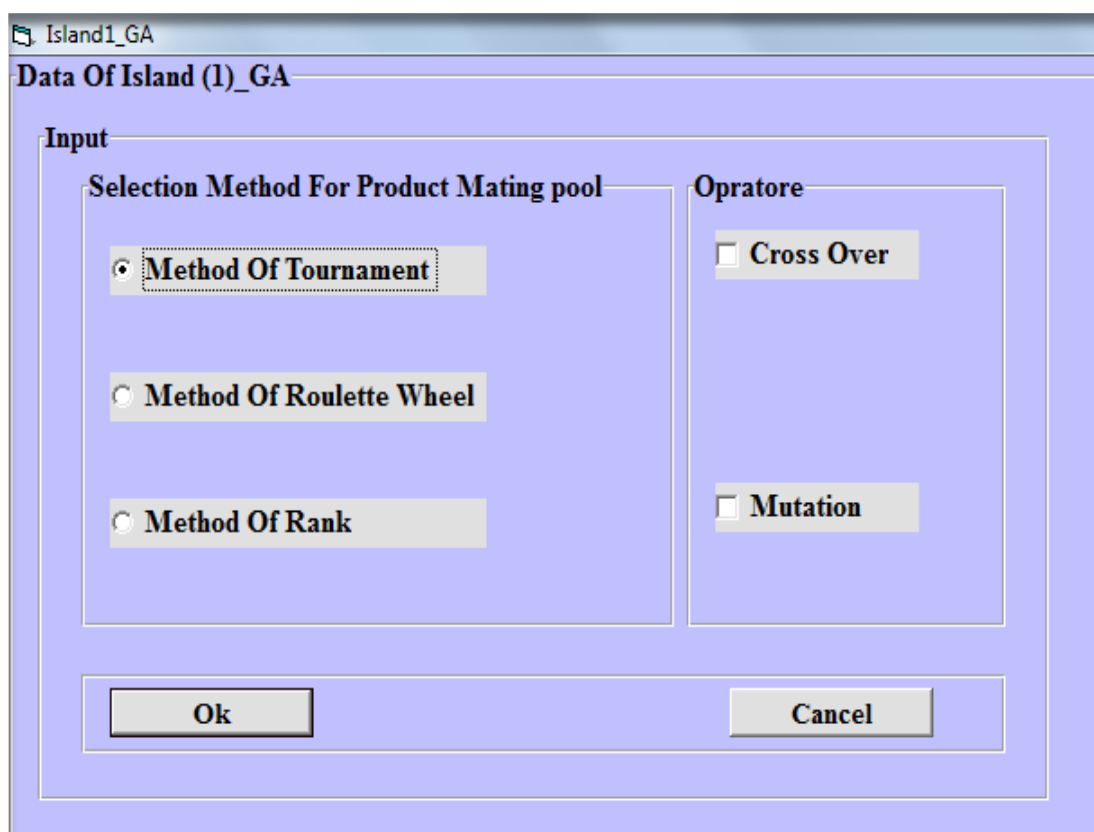
¹ . Print Output

۴-۳-۴ مشخصات جزایر در روش جستجوی چند منظوره مبتنی بر الگوریتم های فراابتکاری

در این بخش به مشخصات جزایر ۴ گانه در بهینه یابی سازه های خریایی پرداخته می شود. هر جزیره با توجه به پارامتر ها و عملگر های به کار گرفته شده به طور مجزا به بهینه یابی سازه پرداخته و در هنگام وقفه مهاجرت تبادل اطلاعات و طرح ها بین جزایر صورت می گیرد.

۴-۳-۴-۱ جزیره (۱)

برای وارد نمودن اطلاعات این جزیره از پنجره Island1_GA استفاده می شود. شکل (۴-۱۷) پنجره Island1_GA را نشان می دهد.



شکل (۴-۱۷) - اطلاعات جزیره (۱)

پنجره Island1_GA با علامت دار نمودن گزینه مربوطه در پنجره Data parameters of MSM ظاهری شود. این پنجره روش های مختلف انتخاب برای تشکیل استخر تولید مثل و همچنین

انتخاب عملگرهای جهش و پیوند را بر عهده کاربر گذاشته شده است. با علامت دار کردن سایر گزینه های موجود در این پنجره، پنجره های شکل (۴-۱۸)، (۴-۱۹) و (۴-۲۰) ظاهر می گردند.

شکل (۴-۱۸) - اندازه سایز رقابت در روش تورنمنت

در این پنجره سایز رقابت مربوط به روش انتخاب تورنمنت توسط کاربر وارد می گردد.

شکل (۴-۱۹) - مشخصات پیوند در جزیره (۱)

در این پنجره مشخصات مربوط به عمل ادغام (پیوند)، از قبیل نوع پیوند، خطی یا غیرخطی بودن نرخ پیوند وارد می گردد.

شکل (۴-۲۰) - مشخصات جهش در جزیره (۱)

در این پنجره مشخصات مربوط به عملکرد جهش از قبیل خطی یا غیر خطی بودن نرخ جهش وارد می گردد. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok در هر پنجره، اطلاعات در متغیرهای جزیره (۱) ذخیره می شوند.

۲-۴-۳-۴ جزیره (۲)

برای وارد نمودن اطلاعات این جزیره از پنجره Island2_ACO استفاده می شود. شکل

(۲۱-۴) پنجره Island2_ACO را نشان می دهد.

شکل (۲۱-۴) - اطلاعات جزیره (۲)

پنجره Island2_ACO با علامت دار نمودن گزینه مربوطه در پنجره Data parameters of

MSM ظاهر می شود. بر اساس این پنجره اطلاعاتی همچون تعداد مورچه های ممتاز، ثابت های α

و β ، نرخ تبخیر فرومون، نرخ پارامترتبخیر فرومون محلی و همچنین نوع بر جای گذاری فرومون

مشخص می گردد.

پس از وارد نمودن اطلاعات ، با فشردن دکمه Ok در هر پنجره، اطلاعات در متغیرهای جزیره

(۲) ذخیره می شوند.

۳-۴-۳-۴ جزیره (۳)

برای وارد نمودن اطلاعات این جزیره از پنجره Island3_HS استفاده می شود. شکل (۴-۲۲) پنجره Island3_HS را نشان می دهد.

Data Of Island (3)_HS	
Input	
Harmony Memory Considering Rate(HMCR):	0.95
Maximum Pitch Adjusting Rate(MAX_PAR):	0.85
Minimum Pitch Adjusting Rate(MIN_PAR):	0.55
Harmony Memory Size(HMS):	70
Ok Cancel	

شکل (۴-۲۲) - اطلاعات جزیره (۳)

پنجره Island3_HS با علامت دار نمودن گزینه مربوطه در پنجره MSM Data parameters of MSM ظاهر می شود. بر اساس این پنجره اطلاعاتی همچون نرخ در نظر گیری حافظه هارمونی، نرخ تنظیم نت ماکزیمم و مینیمم و اندازه حافظه هارمونی توسط کاربر داده می شود. پس از وارد نمودن اطلاعات، با فشردن دکمه Ok در هر پنجره، اطلاعات در متغیرهای جزیره (۳) ذخیره می شوند.

۴-۳-۴ جزیره (۴)

برای وارد نمودن اطلاعات این جزیره از پنجره Island4_CSS استفاده می شود. شکل (۴-۲۳) پنجره Island4_CSS را نشان می دهد.

Data Of Island (4)_CSS	
Input	
Epsilon:	0.0001
Radius Of Particle:	0.95
Constant Kt:	0.9
Constant T:	10
Charge Memory Considering Rate(HMCR):	0.95
Max_Pitch Adjusting Rate(MAX_PAR_Css):	0.85
Min_Pitch Adjusting Rate(MIN_PAR_Css):	0.55
Charge Memory Size(CMS):	20
Ok Cancel	

شکل (۴-۲۳) - اطلاعات جزیره (۴)

پنجره Island4_CSS با علامت دار نمودن گزینه مربوطه در پنجره Data parameters of MSM ظاهر می شود. بر اساس این پنجره اطلاعاتی همچون شعاع ذرات باردار، پارامتر نیرو، پارامتر زمان، نرخ در نظر گیری حافظه بار، نرخ تنظیم نت و اندازه حافظه بار توسط کاربر داده میشود.

پس از وارد نمودن اطلاعات، با فشردن دکمه Ok در هر پنجره، اطلاعات در متغیرهای جزیره (۴) ذخیره می شوند.

در تمامی جزایر مطروحه پنجره Data Parameters Of MSM، به عنوان پنجره مبنا، جهت ورود اطلاعات MSM(GA-ACO-HS-CSS) و فراخوانی جزایر معرفی شده است. این پنجره در شکل (۴-۲۴) نشان داده شده است. براساس این پنجره اطلاعاتی همچون میزان تکرار و نسل سازی به عنوان ورودی‌های عمومی MSM توسط کاربر اعلام می گردند. وقفه مهاجرت نیز از جمله اطلاعاتی می باشد که می‌بایست توسط کاربر اعلام گردد. این اطلاعات مختص الگوریتم جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری می باشند. در این پنجره Migration Rate همان نرخ مهاجرت یا در صد اعضای منتخب برای انتقال از هر جزیره در زمان انتقال و Migration Interval همان وقفه مهاجرت یا بیانگر تعداد نسل ها بین هر انتقال می باشد.

شکل (۴-۲۴) - پنجره ورودی اطلاعات MSM و فراخوانی جزایر

همان طور که بارز است با علامت‌دار نمودن هر یک از گزینه‌های مربوط به جزایر، جزیره مربوطه فراخوانی شده و ظاهر می گردد. در نهایت امر، با فشردن دکمه فرمان OK اطلاعات ورودی در حافظه برنامه ذخیره می شوند.

فصل پنجم:

مسائل های عددی

۵-۱ مقدمه

در این فصل با ارائه چند مثال از سازه های خرپایی دو بعدی و فضایی، به بررسی عملکرد الگوریتم های ژنتیک، اجتماع مورچگان، جستجوی هارمونی، جستجوی ذرات باردار، ژنتیک جزیره ای و روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری از لحاظ سرعت همگرایی و دستیابی به جواب های بهینه می پردازیم. در بررسی های به عمل آمده سعی شده است تا تمامی شرایط یکسان جهت مثال های متداول برقرار شود.

۵-۲ مثال (۱)

در این مثال به ارزیابی خرپای ۴۷ عضوی که از مرجع [۳۶] انتخاب شده است توسط روش پیشنهادی و سایر الگوریتم های ارائه شده می پردازیم. این خرپا در شکل (۵-۱) نشان داده شده است.

این خرپا ۲۲ گره دارد و حول محور y متقارن می باشد. اعضای آن همگی از جنس فولاد بوده و E و ρ برای سایر این اعضا به شرح زیر تعریف می گردند:

$$E = 100000 \text{ Ksi} \quad \& \quad \rho = 0.3 \frac{\text{lb}}{\text{in}^3}$$

در این مثال قید های تنش کششی و فشاری برای کلیه اعضای خرپا به شرح زیر می باشند:

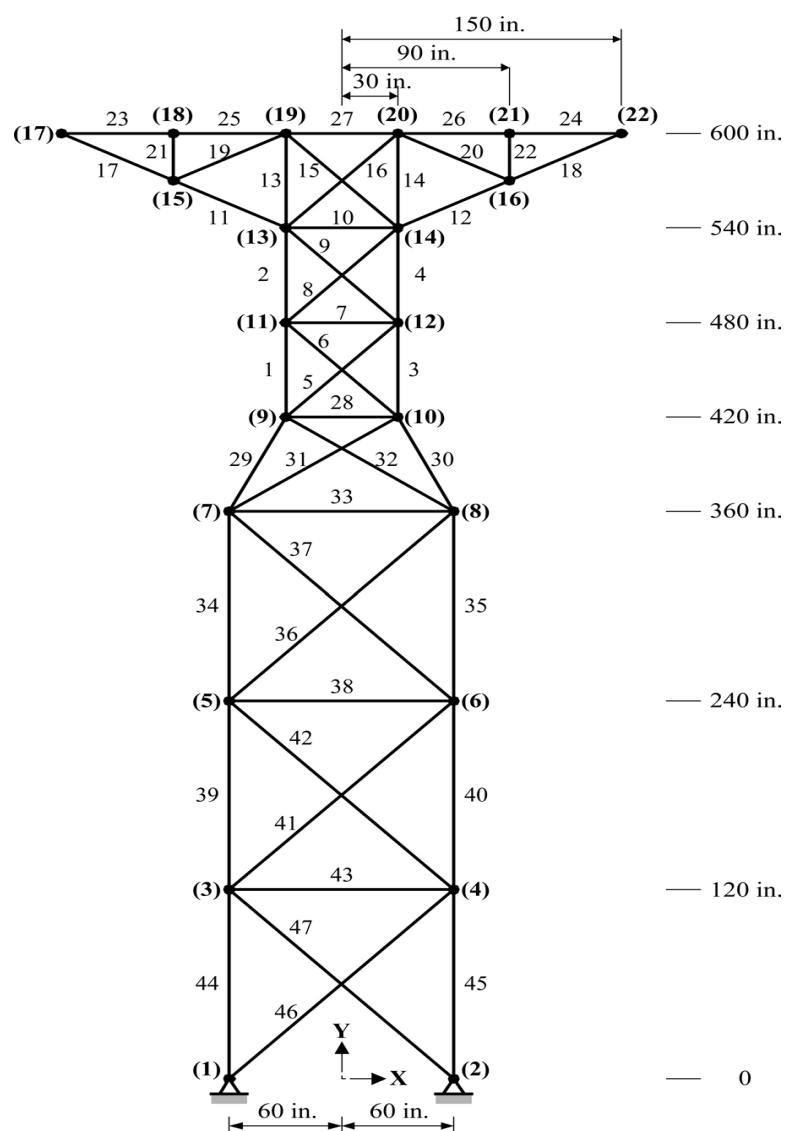
$$\sigma_{all(Ten)} = 20 \text{ Ksi} \quad \& \quad \sigma_{all(Com)} = 15 \text{ Ksi}$$

همچنین تنش مجاز کمانشی اوایلر در نظر گرفته شده برای سایر اعضای فشاری عبارت است از:

$$\sigma_i^{cr} = \frac{-KEA_i}{L_i^2} \quad i = 1, 2, 3, \dots, 47 \quad (5-1)$$

که در این رابطه E مدول الاستیسیته، L طول اعضا، A سطح مقطع اعضا و K ثابت کمانش است که برابر ۳.۹۶ در نظر گرفته شده است. این سازه برای سه حالت بار مجزا طراحی می گردد که

این بارهای موثر در جدول (۱-۵) نشان داده شده است. برای این مثال سطح مقطع های عرضی از مقادیر گسسته جدول (۲-۵) انتخاب می گردند. اعضای این خرپا نیز در ۲۷ گروه به شرح جدول (۳-۵) دسته بندی می شوند. برای این مثال پارامترهای روش پیشنهادی به شرح جدول (۴-۵) در نظر گرفته می شود. همچنین برای این مثال و سایر مثال های ارائه شده در این فصل مشخصات مربوط به روش ژنتیک جزیره ای مطابق شکل های (۴-۴) تا (۹-۴) می باشد.



شکل (۱-۵) خرپای دو بعدی ۴۷ عضوی

جدول (۱-۵) بارهای موثر وارد بر خرپای ۴۷ عضوی

بارگذاری		حالت (۱)		حالت (۲)		حالت (۳)	
گره		17	22	17	22	17	22
P_x (Kips)	6	6	6	6	-----	-----	6
P_y (Kips)	-14	-14	-14	-14	-----	-----	-14

جدول (۲-۵) لیست مقاطع خرپای ۴۷ عضوی جدول (۳-۵) گروه بندی اعضای خرپای ۴۷ عضوی

مقاطع (in^2)		شماره اعضا		گروه	
0.111	3.84	1,3	1	1	1
0.141	3.87	2,4	2	2	2
0.196	3.88	5,6	3	3	3
0.25	4.18	7	4	4	4
0.307	4.22	8,9	5	5	5
0.391	4.49	10	6	6	6
0.442	4.59	11,12	7	7	7
0.563	4.8	13,14	8	8	8
0.602	4.97	15,16	9	9	9
0.766	5.12	17,18	10	10	10
0.785	5.74	19,20	11	11	11
0.994	7.22	21,22	12	12	12
1	7.97	23,24	13	13	13
1.13	8.53	25,26	14	14	14
1.228	9.3	27	15	15	15
1.266	10.85	28	16	16	16
1.457	11.5	29,30	17	17	17
1.563	13.5	31,32	18	18	18
1.62	13.9	33	19	19	19
1.8	14.2	34,35	20	20	20
1.99	15.5	36,37	21	21	21
2.13	16	38	22	22	22
2.38	16.9	39,40	23	23	23
2.62	18.8	41,42	24	24	24
2.63	19.9	43	25	25	25
2.88	22	44,45	26	26	26
2.93	22.9	46,47	27	27	27
3.09	24.5				
3.38	26.5				
3.47	28				
3.55	30				
3.63	33.5				

جدول (۴-۵) پارامتر های مرتبط با روش پیشنهادی در خریای ۴۷ عضوی

جزیره (۴) CSS	جزیره (۳) HS	جزیره (۲) ACO	جزیره (۱) GA
پارامتر a (شعاع): 0.95	پارامتر $HMCR$: 0.93	پارامتر α : 0.55	حداکثر نرخ پیوند: 0.85
پارامتر $CMCR$: 0.94	پارامتر HMS : 40	پارامتر β : 0.15	حداقل نرخ پیوند: 0.8
پارامتر CMS : 20	حداکثر پارامتر PAR : 0.85	تعداد مورچه ممتاز: 15	حداکثر نرخ جهش: 0.005
پارامتر ΔT : 10 پارامتر K_t : 0.9	حداقل پارامتر PAR : 0.55	نرخ تبخیر سراسری: 0.1	حداقل نرخ جهش: 0.001
حداکثر پارامتر PAR : 0.85	-----	پارامتر تبخیر محلی: 0.6	روش انتخاب: تورنمنت
حداقل پارامتر PAR : 0.55	-----	پارامتر τ_0 : 100	سایز انتخاب: 3
نرخ مهاجرت: 40		وقفه مهاجرت: 15	
تعداد تکرار: 300	ثابت K : 4		تعداد اعضای جمعیت: 200

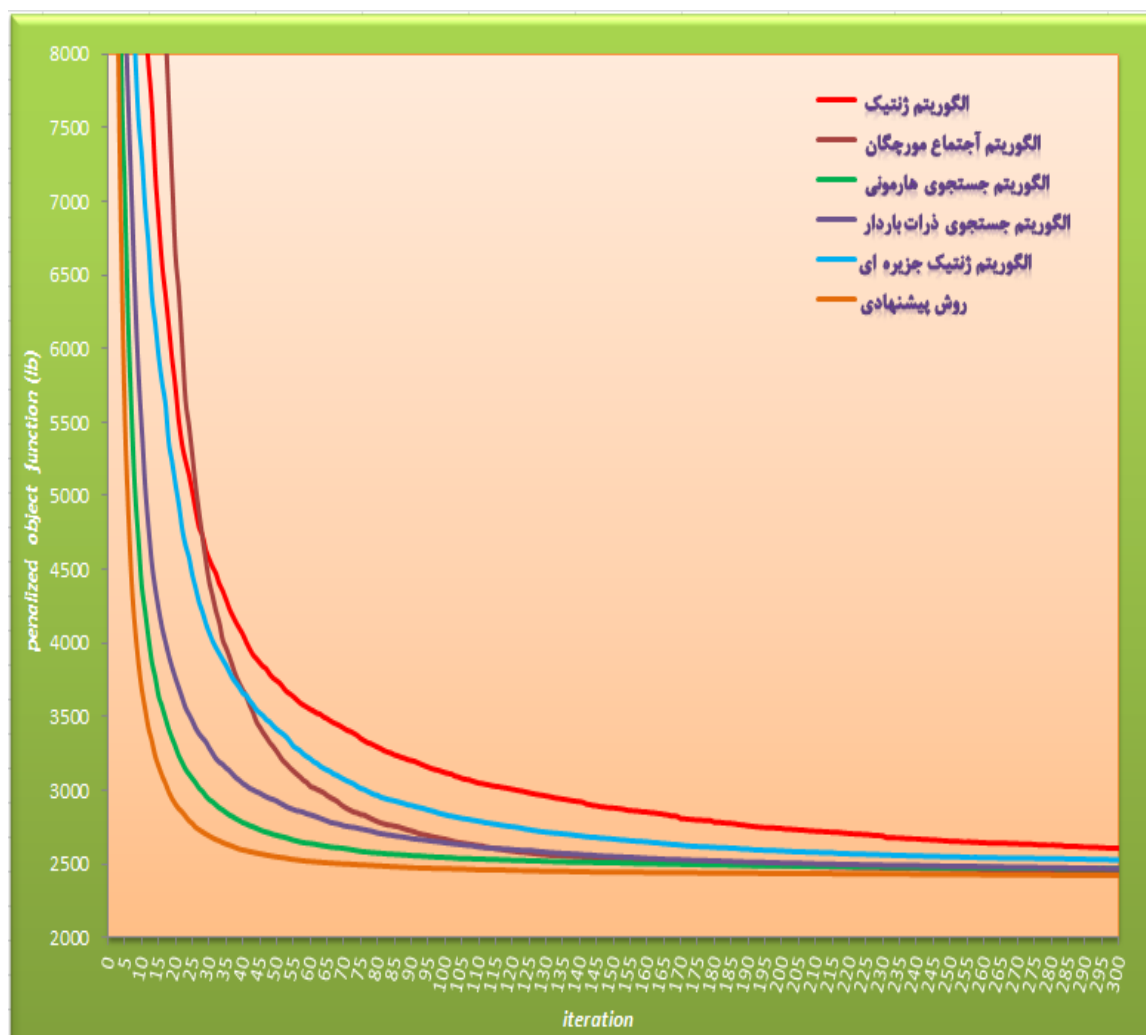
نتایج حاصل از طرح بهینه ی هریک از الگوریتم ها به همراه مقادیر توابع هدف و حجم مربوط به این طرح ها و نتایج طرح بهینه مرجع [۳۶] در جدول (۵-۵) نشان داده شده است. همچنین مقدار میانگین تابع هدف حاصل از ۵۰ بار اجرای مجزای برنامه بهینه سازی سازه برای هر الگوریتم نیز در این جدول قرار داده شده است.

جدول (۵-۵) نتایج طرح بهینه خریای ۴۷ عضوی

شماره گروه	GA	ACO	HS	CSS	MSM GA	LEE et.al[36]	روش پیشنهادی
1	3.88	3.87	3.84	3.84	3.84	3.84	3.84
2	3.38	3.38	3.38	3.38	3.38	3.38	3.38
3	0.766	0.785	0.785	0.785	0.785	0.766	0.766
4	0.196	0.307	0.111	0.196	0.141	0.141	0.111
5	0.994	1	0.785	0.785	0.766	0.785	0.785
6	1.99	1.8	1.99	2.38	3.63	1.99	1.99
7	2.13	2.13	2.13	2.13	2.13	2.13	2.13
8	1.457	1.228	1.228	1.228	1.228	1.228	1.228
9	1.563	1.62	1.563	1.563	1.563	1.563	1.563
10	2.38	2.13	2.13	2.13	2.13	2.13	2.13
11	0.111	0.111	0.111	0.25	0.111	0.111	0.111
12	0.111	0.141	0.111	0.141	0.111	0.111	0.111
13	1.99	1.8	1.8	1.8	1.8	1.8	1.8
14	1.8	1.8	1.8	1.8	1.99	1.8	1.8
15	1.457	1.457	1.457	1.457	1.457	1.457	1.457
16	0.766	0.563	0.602	0.602	0.391	0.442	0.563
17	3.63	3.63	3.63	3.63	3.63	3.63	3.63
18	1.457	1.457	1.563	1.457	1.563	1.457	1.457
19	0.442	0.25	0.25	0.307	0.442	0.391	0.25
20	3.09	3.09	3.09	3.09	3.09	3.09	3.09
21	1.457	1.266	1.266	1.266	1.457	1.457	1.228
22	0.111	0.391	0.307	0.391	0.196	0.196	0.307
23	3.84	3.84	3.84	3.84	3.84	3.84	3.84
24	1.62	1.563	1.563	1.563	1.563	1.563	1.563
25	0.196	0.111	0.111	0.111	0.111	0.196	0.141
26	4.59	4.59	4.59	4.59	4.59	4.59	4.59
27	1.563	1.457	1.563	1.457	1.457	1.457	1.457
Best Weight(lb)	2456.31	2394.51	2394.3	2396.4	2437.99	2396.88	2372.15
Average(lb)	2605.5	2446.3	2467.3	2468.2	2526.9	----	2421.1
Volume(in ³)	8187.71	7981.7	7980.9	7987.9	8126.6	7989.6	7907.17

همان طور که در جدول (۵-۵) مشاهده می شود؛ روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری وزن کم تری را نسبت به سایر روش ها و مراجع می دهد.

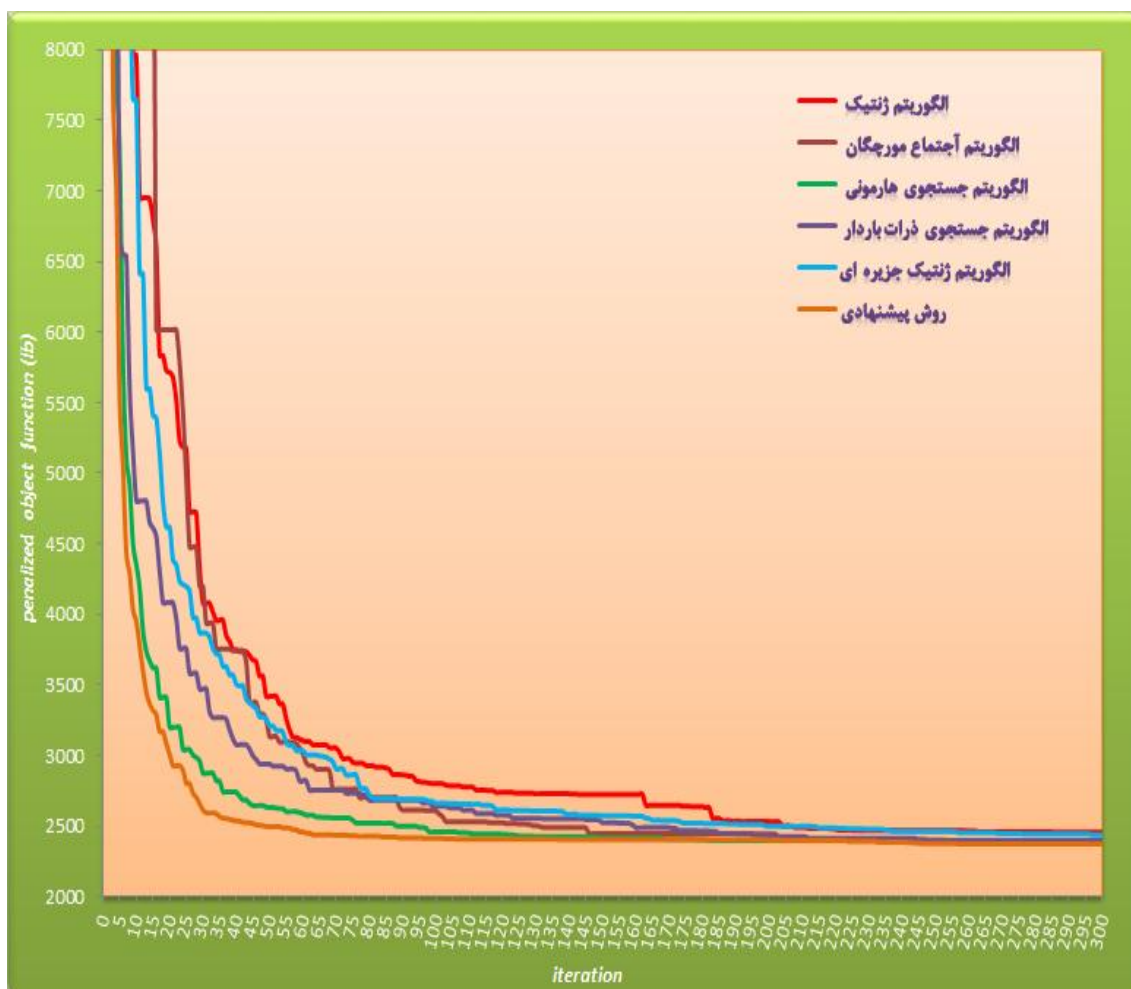
در شکل (۵-۲) نمودارهای همگرایی حاصل از میانگین ۵۰ بار اجرای مجزای برنامه بهینه سازی سازه با استفاده از الگوریتم ژنتیک جزیره ای، روش پیشنهادی و روش های به کار گرفته شده در سایر جزایر، به طور مجزا نشان داده شده است.



شکل (۵-۲) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خریای ۴۷ عضوی

همان طور که مشاهده می شود؛ روش پیشنهادی که ترکیبی از سایر الگوریتم ها می باشد، روند همگرایی مناسبی دارد. در میان جزایر، جزیره (۱) سرعت همگرایی ضعیف تری نسبت به سایر جزایر در دستیابی به جواب های بهینه دارد. جزیره (۲) در ابتدا با سرعت همگرایی کم تری به جستجوی فضای طراحی می پردازد و پس از انجام جستجوی کلی، روند مناسبی جهت یافتن جواب های بهینه در پیش می گیرد. جزایر (۳) و (۴) در انجام جستجوی کلی موفق عمل می نمایند و سرعت همگرایی اولیه مناسبی دارند و قادرند جواب های بهینه مناسبی را بدست آورند.

در شکل (۳-۵) نمودار های همگرایی برای بهترین جواب های حاصل از اجرای برنامه توسط سایر روش ها برای خرپای ۴۷ عضوی ارائه شده است.



شکل (۳-۵) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خرپای ۴۷ عضوی

۵-۳ مثال (۲)

در این مثال به ارزیابی خرابی فضایی ۷۲ عضوی توسط روش پیشنهادی و سایر الگوریتم های ارائه شده می پردازیم. این مثال الگوی بسیاری از الگوریتم های مطرح در دنیای بهینه سازی است و از آن برای نشان دادن کارایی این الگوریتم ها استفاده می گردد. این خرابا در شکل (۵-۴) نشان داده شده است.

برای کلیه اعضاء این خرابا E و ρ به شرح زیر تعریف می گردند:

$$E = 10000Ksi \quad \& \quad \rho = 0.1 \frac{lb}{in^3}$$

در این مثال سازه در معرض قیود تنش و تغییر مکان قرار گرفته است. تنش های کششی و فشاری مجاز برای کلیه اعضای خرابا به شرح زیر می باشند:

$$\sigma_{all(Ten)} = +25Ksi \quad \& \quad \sigma_{all(Com)} = -25Ksi$$

همچنین تغییر مکان (جابه جایی) مجاز سایر گره های سازه در تمام جهات برابر است با:

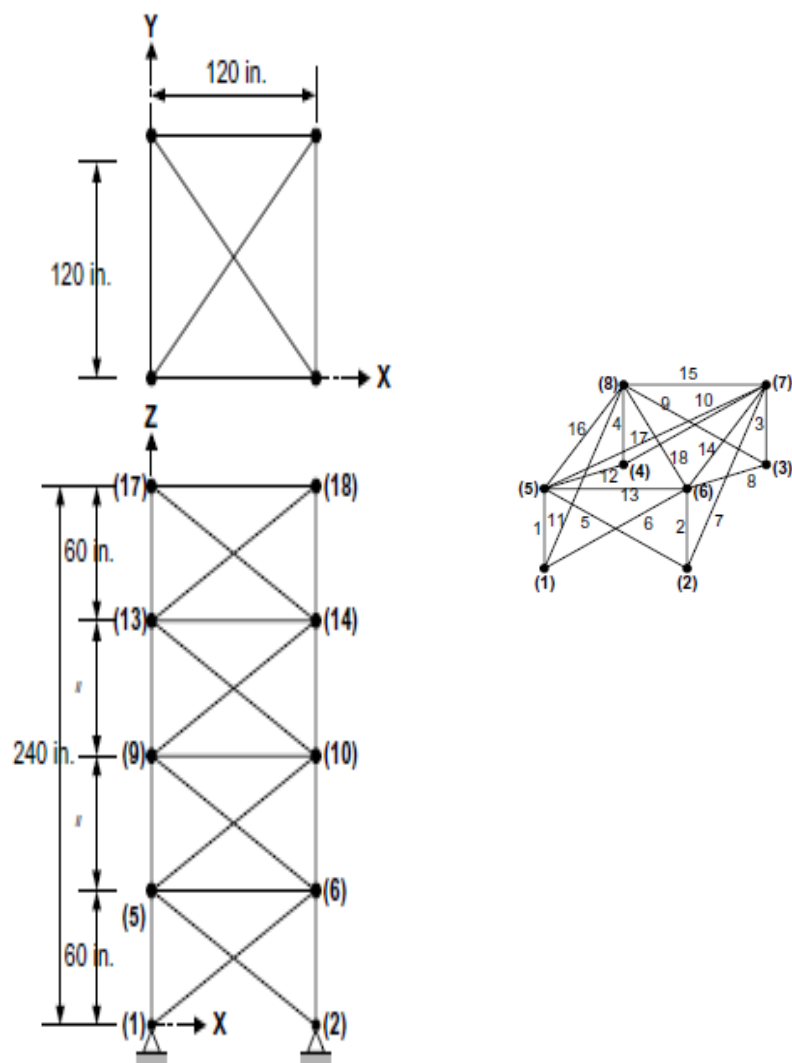
$$\delta_{all}^i = 0.25in \quad i = 1, 2, 3, \dots, 20$$

بارهای موثر وارد بر سازه در ۲ حالت در جدول (۵-۶) نشان داده شده است.

برای این مثال سطح مقطع های عرضی اعضاء از مقادیر گسسته جدول (۵-۷) انتخاب می گردند.

این خرابا دارای ۲۰ گره می باشد و در دو راستای x و y متقارن است. این تقارن سبب می گردد اعضای خرابا در ۱۶ گروه به شرح جدول (۵-۸) دسته بندی شوند.

برای این مثال پارامترهای روش پیشنهادی جهت کسب جواب های بهینه به شرح جدول (۵-۹) در نظر گرفته می شود.



شکل (۴-۵) خرپای فضایی ۷۲ عضوی

جدول (۶-۵) بارهای موثر وارد بر خرپای فضایی ۷۲ عضوی

بارگذاری				حالت (۱)				حالت (۲)			
گره				17	18	19	20	17	18	19	20
P_x (Kips)				5	-----	-----	-----	-----	-----	-----	-----
P_y (Kips)				5	-----	-----	-----	-----	-----	-----	-----
P_z (Kips)				5	-----	-----	-----	-5	-5	-5	-5

جدول (۵-۸) گروه بندی اعضای خرپای ۷۲ عضوی

گروه	شماره اعضا
1	1,2,3,4
2	5,6,7,8,9,10,11,12
3	13,14,15,16
4	17,18
5	19,20,21,22
6	23,24,25,26,27,28,29,30
7	31,32,33,34
8	35,36
9	37,38,39,40
10	41,42,43,44,45,46,47,48
11	49,50,51,52
12	53,54
13	55,56,57,58
14	59,60,61,62,63,64,65,66
15	67,68,69,70
16	71,72

جدول (۵-۷) لیست مقاطع خرپای ۷۲ عضوی

مقاطع (in ²)			
0.111	1.457	3.84	11.5
0.141	1.563	3.87	13.5
0.196	1.62	3.88	13.9
0.25	1.8	4.18	14.2
0.307	1.99	4.22	15.5
0.391	2.13	4.49	16
0.442	2.38	4.59	16.9
0.563	2.62	4.8	18.8
0.602	2.63	4.97	19.9
0.766	2.88	5.12	22
0.785	2.93	5.74	22.9
0.994	3.09	7.22	24.5
1	3.38	7.97	26.5
1.13	3.47	8.53	28
1.228	3.55	9.3	30
1.266	3.63	10.85	33.5

جدول (۵-۹) پارامتر های مرتبط با روش پیشنهادی در خریای ۷۲ عضوی

جزیره (۱)	جزیره (۲)	جزیره (۳)	جزیره (۴)
GA	ACO	HS	CSS
حداکثر نرخ پیوند:	پارامتر α :	پارامتر $HMCR$:	پارامتر a (شعاع):
0.85	0.55	0.93	0.95
حداقل نرخ پیوند:	پارامتر β :	پارامتر HMS :	پارامتر $CMCR$:
0.8	0.15	70	0.99
حداکثر نرخ جهش:	تعداد مورچه ممتاز:	حداکثر پارامتر PAR :	پارامتر CMS :
0.005	15	0.75	20
حداقل نرخ جهش:	نرخ تبخیر سراسری:	حداقل پارامتر PAR :	پارامتر ΔT : پارامتر K_t : 0.9
0.005	0.1	0.75	
روش انتخاب: تورنمنت	پارامتر تبخیر محلی:	-----	حداکثر پارامتر PAR :
	0.6		0.9
سایز انتخاب:	پارامتر τ_0 :	-----	حداقل پارامتر PAR :
3	50		0.9
وقفه مهاجرت: 15		نرخ مهاجرت: 40	
تعداد اعضای جمعیت: 200		ثابت K : 4	
		تعداد تکرار: 300	

نتایج حاصل از طرح بهینه ی هریک از الگوریتم ها به همراه مقادیر توابع هدف و حجم مربوط

به این طرح ها و نتایج طرح بهینه مراجع مختلف در جدول (۵-۱۰) نشان داده شده است.

همچنین مقدار میانگین تابع هدف حاصل از ۵۰ بار اجرای مجزای برنامه بهینه سازی سازه

برای هر الگوریتم نیز در جدول (۵-۱۰) قرار داده شده است.

جدول (۵-۱۰) نتایج طرح بهینه خریای ۷۲ عضوی

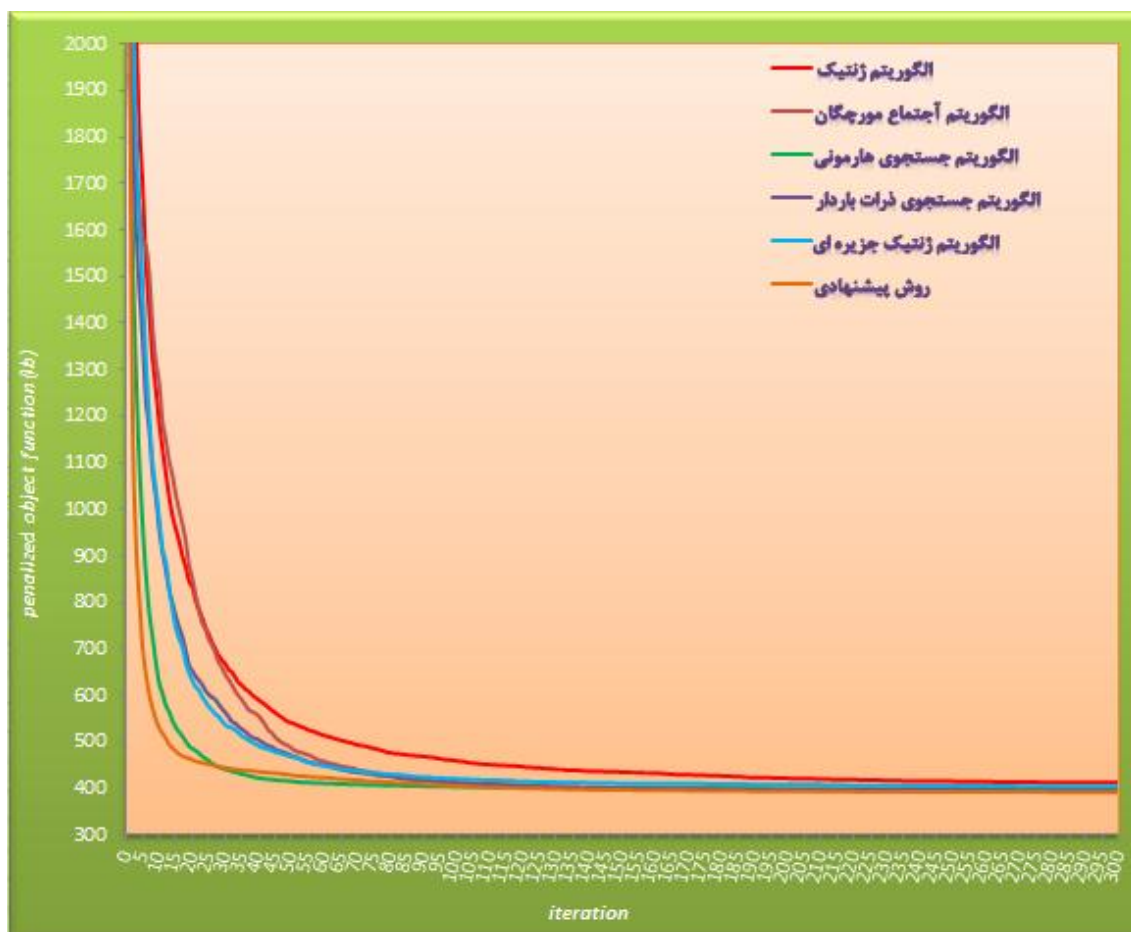
شماره گروه	GA	ACO	HS	CSS	MSM GA	Wu [37]	Li [38]	kaveh [32]	kaveh [39]	روش پیشنهادی
1	1.8	1.99	1.8	1.8	1.99	0.196	4.49	1.8	1.99	1.99
2	0.563	0.442	0.563	0.442	0.442	0.602	1.457	0.442	0.442	0.442
3	0.111	0.111	0.111	0.111	0.111	0.307	0.111	0.141	0.111	0.111
4	0.111	0.111	0.111	0.111	0.111	0.766	0.111	0.111	0.111	0.111
5	1.62	1.266	1.266	1.228	1.266	0.391	2.62	1.228	0.994	1.228
6	0.563	0.563	0.563	0.563	0.563	0.391	1.13	0.563	0.563	0.563
7	0.111	0.111	0.111	0.111	0.111	0.141	0.196	0.111	0.111	0.111
8	0.111	0.111	0.111	0.111	0.111	0.111	0.111	0.111	0.111	0.111
9	0.602	0.442	0.563	0.442	0.602	1.8	1.266	0.563	0.563	0.563
10	0.442	0.563	0.442	0.563	0.602	0.602	1.457	0.563	0.563	0.563
11	0.111	0.111	0.111	0.111	0.111	0.141	0.111	0.111	0.111	0.111
12	0.111	0.111	0.111	0.111	0.111	0.307	0.111	0.25	0.111	0.111
13	0.196	0.196	0.196	0.196	0.196	1.563	0.442	0.196	0.196	0.196
14	0.563	0.563	0.602	0.602	0.563	0.766	1.457	0.563	0.563	0.563
15	0.307	0.442	0.391	0.442	0.391	0.141	1.228	0.442	0.442	0.391
16	0.602	0.563	0.563	0.602	0.442	0.111	1.457	0.563	0.766	0.563
Best Weight(lb)	392.4	389.8	389.9	389.8	391.3	427.2	941.8	393.4	393.05	389.3
Average(lb)	412.2 651	393.9	398.0 908	395.3	403.6	-----	-----	-----	-----	391.64 42
Volume (in ³)	3920. 4	3890. 8	3890. 9	3890. 8	3910. 3	4270.2	9410. 8	3930. 4	3930.0 5	3890.3

همان طور که در جدول (۵-۱۰) مشاهده می شود بهترین طرح موجود در مراجع مختلف توسط کاوه

و همکارانش در سال (۲۰۱۰) ارائه شده است که وزن کمینه این طرح ۳۹۳.۰۵ پوند می باشد. توجه

به نتایج طرح های حاصل از این پایان نامه برای خرپای ۷۲ عضوی که در جدول (۵-۱۰) ارائه شده است نشان می دهد که سایر روش ها دارای جواب های مناسب تری نسبت به جواب ارائه شده توسط کاوه می باشند. همچنین روش پیشنهادی با کاوش مناسب فضای جستجو نیل به سوی جواب بهینه مطلق داشته و طرح بهینه مناسب تری را بدست آورده است.

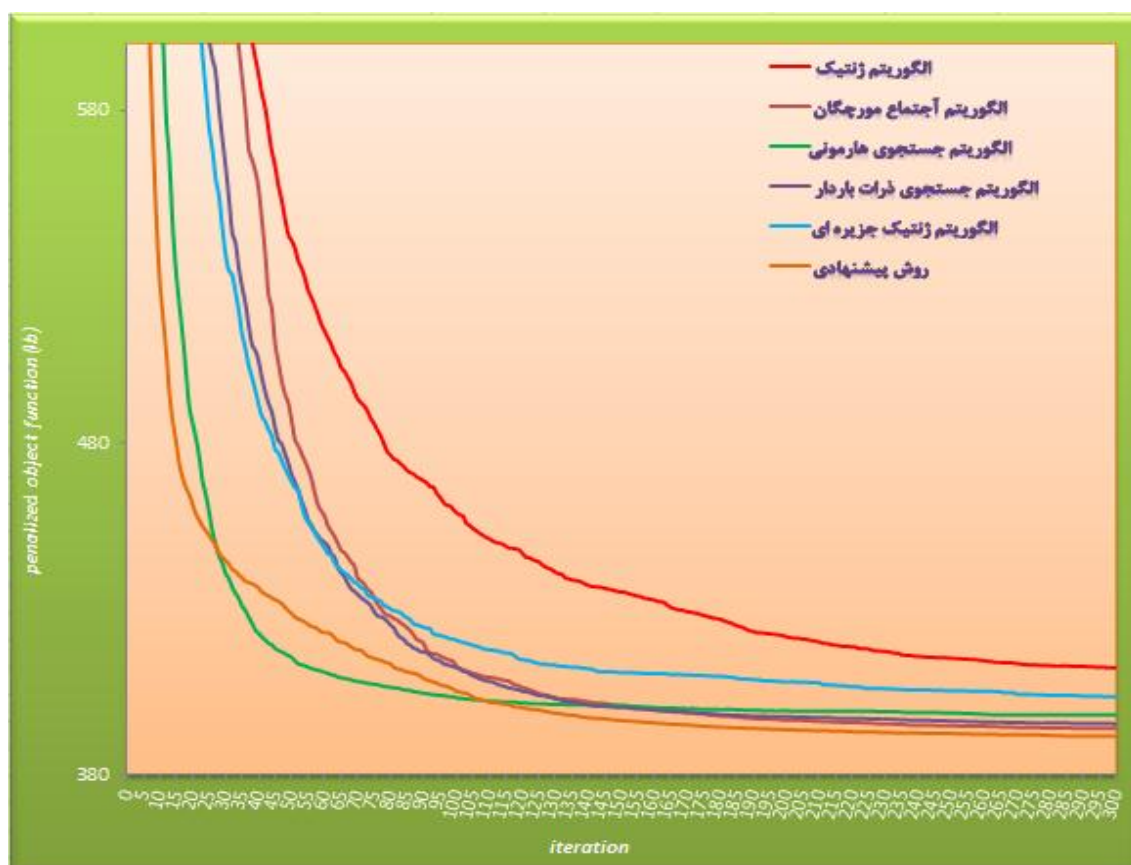
در شکل (۵-۵) نمودار های همگرایی حاصل از میانگین ۵۰ بار اجرای مجزای برنامه بهینه سازی سازه با استفاده از الگوریتم ژنتیک جزیره ای، روش پیشنهادی و روش های به کار گرفته شده در سایر جزایر به طور مجزا برای خرپای ۷۲ عضوی نشان داده شده است.



شکل (۵-۵) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر

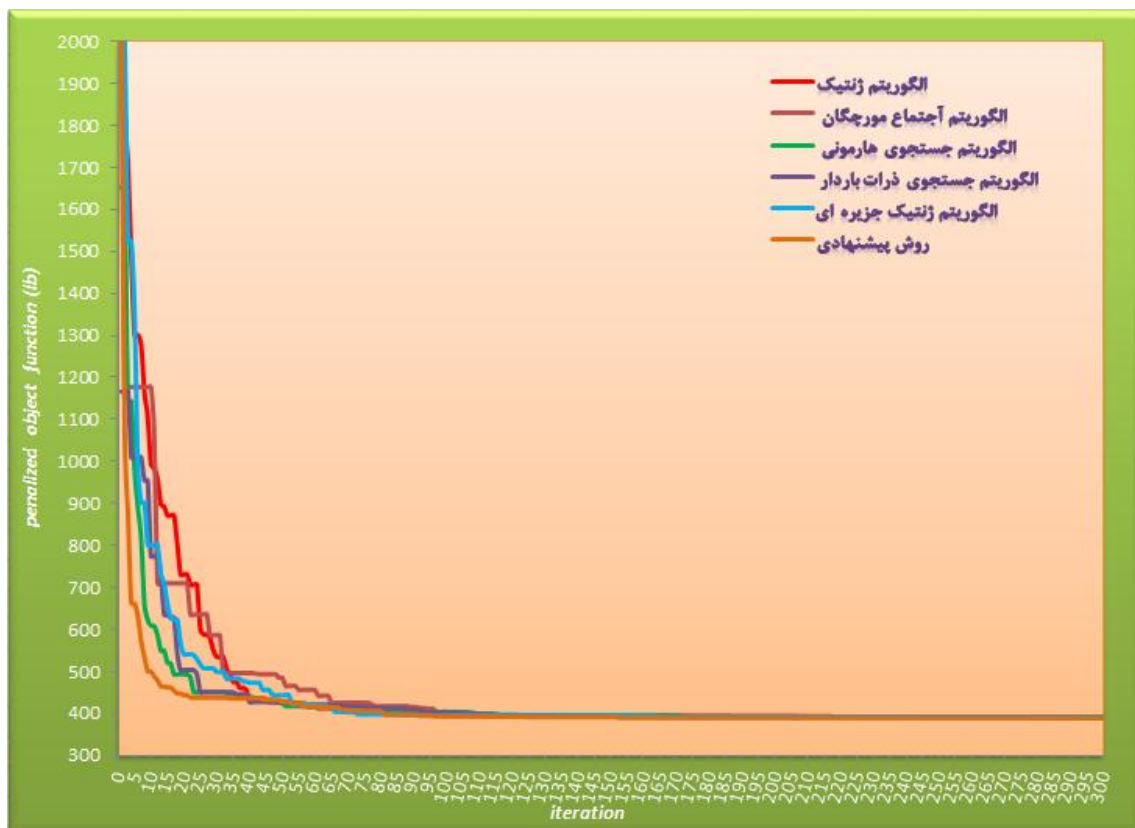
الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۷۲ عضوی

همان طور که مشاهده می شود؛ در این مثال سایر روش ها و جزایر به طور نسبی در دستیابی به جواب بهینه موفق عمل می کنند. روش پیشنهادی که ترکیبی از سایر الگوریتم ها می باشد با سرعت همگرایی مناسب و متعادل به سمت نقطه کمینه حرکت می نماید. در میان جزایر، جزیره (۱) سرعت همگرایی ضعیف تری نسبت به سایر جزایر در دستیابی به جواب های بهینه دارد. جزیره (۲) در ابتدا با سرعت همگرایی کم تری به جستجوی فضای طراحی می پردازد و پس از انجام جستجوی کلی، روند مناسبی جهت یافتن جواب های بهینه در پیش می گیرد و به صورت مماس بر جزیره (۴) حرکت می نماید. جزیره (۳) در انجام جستجوی کلی موفق عمل می نماید و سرعت همگرایی اولیه بسیار مناسبی دارد. جزیره (۴) در ابتدا به صورت مماس بر الگوریتم ژنتیک جزیره ای حرکت می نماید و در ادامه مطابق شکل (۵-۶) که شکل بزرگ نمایی شده ی شکل (۵-۵) می باشد روند همگرایی مناسب تری می گیرد.



شکل (۵-۶) بزرگ نمایی شکل (۵-۵)

در شکل (۵-۷) نمودار های همگرایی برای بهترین جواب های حاصل از اجرای برنامه توسط سایر روش ها برای خرپای ۷۲ عضوی ارائه شده است.



شکل (۵-۷) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خرپای ۷۲ عضوی

۴-۵ مثال (۳)

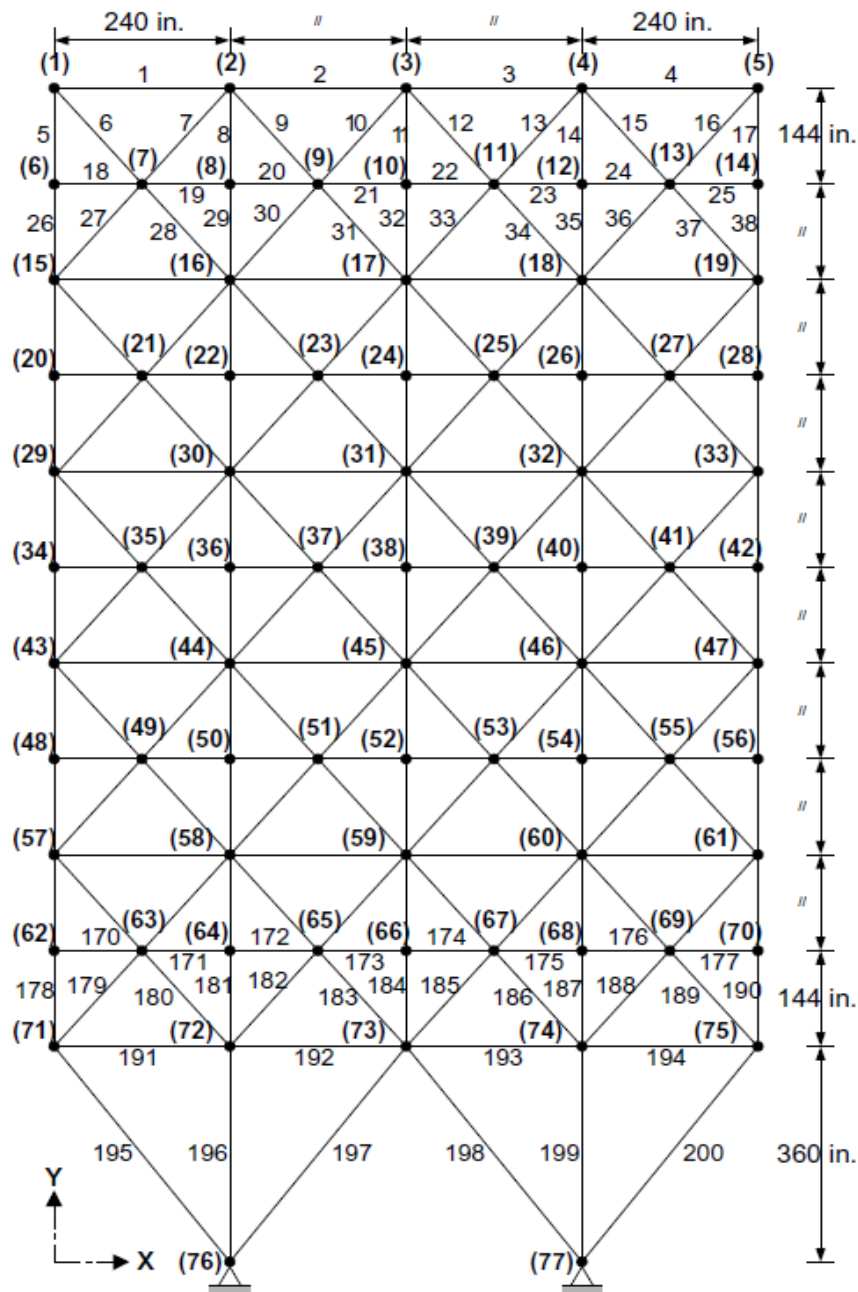
در سومین مثال ارائه شده به بررسی عملکرد روش پیشنهادی و سایر الگوریتم ها در بهینه یابی خرپای ۲۰۰ عضوی با ۲۹ گروه که خرپای نسبتاً بزرگ و مطرحی جهت اثبات کارایی الگوریتم های مختلف می باشد، می پردازیم. این خرپا در شکل (۵-۸) نشان داده شده است.

برای کلیه اعضا خرپا E و ρ به شرح زیر تعریف می گردند:

$$E = 30000 \text{ Ksi} \quad \& \quad \rho = 0.283 \frac{\text{lb}}{\text{in}^3}$$

در این مثال فقط قید های تنش کششی و فشاری برای کلیه اعضای خرپا اعمال شده که به شرح زیر می باشند:

$$\sigma_{all(Ten)} = +10Ksi \quad \& \quad \sigma_{all(Com)} = -10Ksi$$



شکل (۵-۸) خرپای دو بعدی ۲۰۰ عضوی با ۲۹ گروه

اعضای این خرپا در ۲۹ گروه به شرح جدول (۵-۱۱) دسته بندی می شوند.

جدول (۵-۱۱) گروه بندی اعضاي خرياي ۲۰۰ ميله اي (۲۹گروه)

گروه	شماره اعضا
1	1 , 2 , 3 , 4
2	5 , 8 , 11 , 14 , 17
3	19 , 20 , 21 , 22 , 23 , 24
4	18 , 25 , 56 , 63 , 94 , 101 , 132 , 139 , 170 , 177
5	26 , 29 , 32 , 35 , 38
6	6 , 7 , 9 , 10 , 12 , 13 , 15 , 16 , 27 , 28 , 30 , 31 , 33 , 34 , 36 , 37
7	39 , 40 , 41 , 42
8	43 , 46 , 49 , 52 , 55
9	57 , 58 , 59 , 60 , 61 , 62
10	64 , 67 , 70 , 73 , 76
11	44 , 45 , 47 , 48 , 50 , 51 , 53 , 54 , 65 , 66 , 68 , 69 , 71 , 72 , 74 , 75
12	77 , 78 , 79 , 80
13	81 , 84 , 87 , 90 , 93
14	95 , 96 , 97 , 98 , 99 , 100
15	102 , 105 , 108 , 111 , 114
16	82 , 83 , 85 , 86 , 88 , 89 , 91 , 92 , 103 , 104 , 106 , 107 , 109 , 110 , 112 , 113
17	115 , 116 , 117 , 118
18	119 , 122 , 125 , 128 , 131
19	133 , 134 , 135 , 136 , 137 , 138
20	140 , 143 , 146 , 149 , 152
21	120 , 121 , 123 , 124 , 129 , 127 , 129 , 130 , 141 , 142 , 144 , 145 , 147 , 148 , 150 , 151
22	153 , 154 , 155 , 156
23	157 , 160 , 163 , 166 , 169
24	171 , 172 , 173 , 174 , 175 , 176
25	178 , 181 , 184 , 187 , 190
26	158 , 159 , 161 , 162 , 164 , 165 , 167 , 168 , 179 , 180 , 182 , 183 , 185 , 186 , 188 , 189
27	191 , 192 , 193 , 194
28	195 , 197 , 198 , 200
29	196 , 199

بارهای موثر در ۳ حالت بر سازه اعمال می گردد که در جداول (۵-۱۲) تا (۵-۱۴) نشان داده شده

است. برای این مثال سطح مقطع های عرضی از مقادیر گسسته جدول (۵-۱۵) انتخاب می گردند.

جدول (۵-۱۲) بارگذاری مؤثر بر خرابی ۲۰۰ میله ای (حالت ۱)

شماره گره	نیرو در جهت X P_X (kip)	نیرو در جهت Y P_Y (kip)
1	1	0
6	1	0
15	1	0
20	1	0
29	1	0
34	1	0
43	1	0
48	1	0
57	1	0
62	1	0
71	1	0

جدول (۵-۱۳) بارگذاری مؤثر بر خرابی ۲۰۰ میله ای (حالت ۲)

شماره گره	نیرو در جهت X P_X (kip)	نیرو در جهت Y P_Y (kip)	شماره گره	نیرو در جهت X P_X (kip)	نیرو در جهت Y P_Y (kip)
1	0	-10	40	0	-10
2	0	-10	42	0	-10
3	0	-10	43	0	-10
4	0	-10	44	0	-10
5	0	-10	45	0	-10
6	0	-10	46	0	-10
8	0	-10	47	0	-10
10	0	-10	48	0	-10
12	0	-10	50	0	-10
14	0	-10	52	0	-10
15	0	-10	54	0	-10
16	0	-10	56	0	-10
17	0	-10	57	0	-10
18	0	-10	58	0	-10
19	0	-10	59	0	-10
20	0	-10	60	0	-10
22	0	-10	61	0	-10
24	0	-10	62	0	-10
26	0	-10	64	0	-10
28	0	-10	66	0	-10
29	0	-10	68	0	-10
30	0	-10	70	0	-10
31	0	-10	71	0	-10
32	0	-10	72	0	-10
33	0	-10	73	0	-10
34	0	-10	74	0	-10
36	0	-10	75	0	-10
38	0	-10			

جدول (۵-۱۴) بارگذاری مؤثر بر خرابای ۲۰۰ میله ای (حالت ۳)

شماره گره	نیرو در جهت X P_X (kip)	نیرو در جهت Y P_Y (kip)	شماره گره	نیرو در جهت X P_X (kip)	نیرو در جهت Y P_Y (kip)
1	1	-10	39	0	0
2	0	-10	40	0	-10
3	0	-10	41	0	0
4	0	-10	42	0	-10
5	0	-10	43	1	-10
6	1	-10	44	0	-10
7	0	0	45	0	-10
8	0	-10	46	0	-10
9	0	0	47	0	-10
10	0	-10	48	1	-10
11	0	0	49	0	0
12	0	-10	50	0	-10
13	0	0	51	0	0
14	0	-10	52	0	-10
15	1	-10	53	0	0
16	0	-10	54	0	-10
17	0	-10	55	0	0
18	0	-10	56	0	-10
19	0	-10	57	1	-10
20	1	-10	58	0	-10
21	0	0	59	0	-10
22	0	-10	60	0	-10
23	0	0	61	0	-10
24	0	-10	62	1	-10
25	0	0	63	0	0
26	0	-10	64	0	-10
27	0	0	65	0	0
28	0	-10	66	0	-10
29	1	-10	67	0	0
30	0	-10	68	0	-10
31	0	-10	69	0	0
32	0	-10	70	0	-10
33	0	-10	71	0	-10
34	1	-10	72	0	-10
35	0	0	73	0	-10
36	0	-10	74	0	-10
37	0	0	75	0	-10
38	0	-10			

جدول (۵-۱۵) لیست مقاطع خرابای ۲۰۰ عضوی

مقاطع (in^2)					
0.1	1.081	2.142	3.813	8.525	17.17
0.347	1.174	2.697	4.805	9.3	19.18
0.44	1.333	2.8	5.952	10.85	23.68
0.539	1.488	3.131	6.572	13.33	28.08
0.954	1.764	3.565	7.192	14.29	33.7

برای این مثال پارامترهای روش پیشنهادی به شرح جدول (۵-۱۶) در نظر گرفته می شود.

جدول (۵-۱۶) پارامتر های مرتبط با روش پیشنهادی در خریای ۲۰۰ عضوی

جزیره (۱) GA	جزیره (۲) ACO	جزیره (۳) HS	جزیره (۴) CSS
حداکثر نرخ پیوند:	پارامتر α :	پارامتر $HMCR$:	پارامتر a (شعاع):
0.85	0.55	0.93	0.9
حداقل نرخ پیوند:	پارامتر β :	پارامتر HMS :	پارامتر $CMCR$:
0.8	0.15	40	0.92
حداکثر نرخ جهش:	تعداد مورچه ممتاز:	حداکثر پارامتر PAR :	پارامتر CMS :
0.005	15	0.85	25
حداقل نرخ جهش:	نرخ تبخیر سراسری:	حداقل پارامتر PAR :	پارامتر ΔT : 10
0.001	0.1	0.55	پارامتر K_t : 0.9
روش انتخاب:	پارامتر تبخیر محلی:	-----	حداکثر پارامتر PAR :
تورنمنت	0.6		0.85
سایز انتخاب:	پارامتر τ_0 :	-----	حداقل پارامتر PAR :
3	100		0.55
وقفه مهاجرت: 15		نرخ مهاجرت: 40	
تعداد اعضای جمعیت: 200		ثابت K : 4	
		تعداد تکرار: 300	

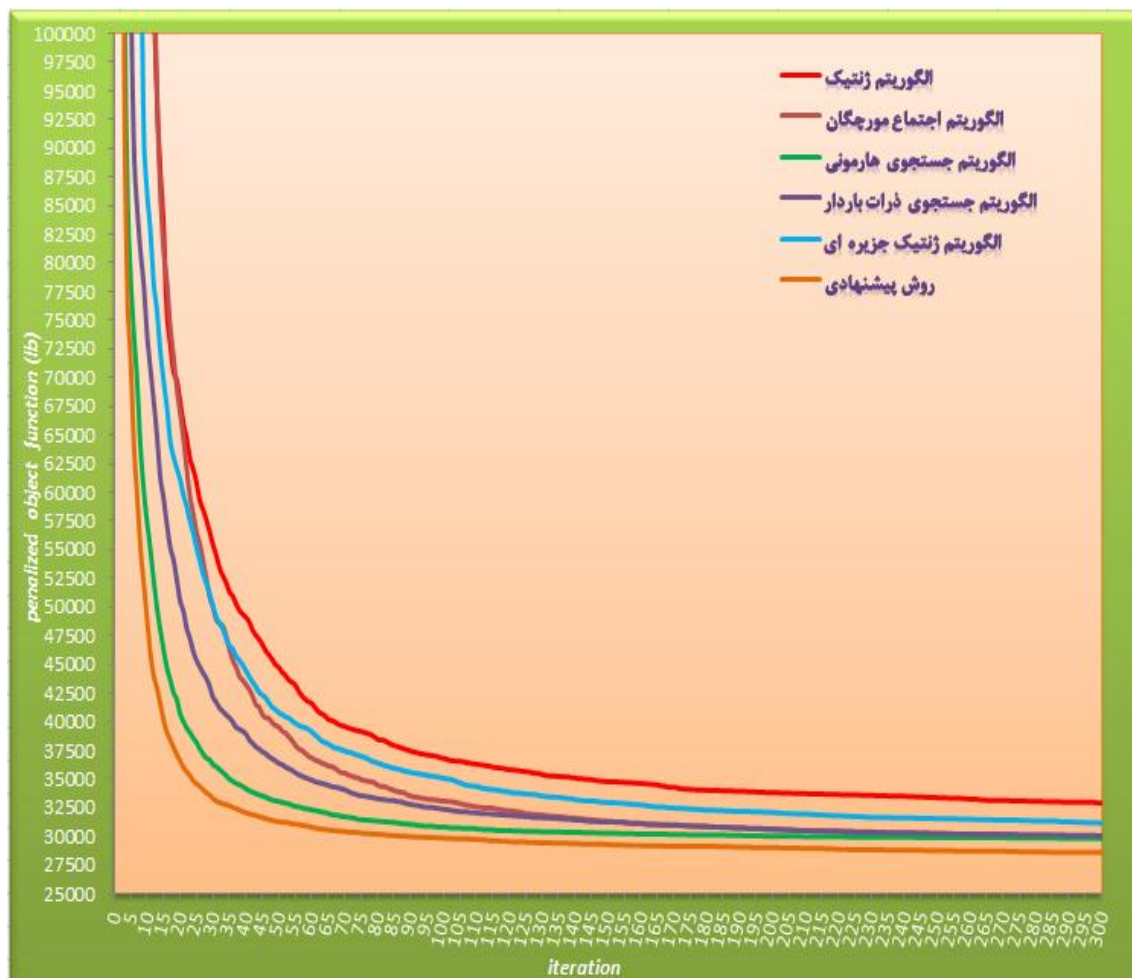
نتایج حاصل از طرح بهینه ی هریک از الگوریتم ها به همراه مقادیر توابع هدف و حجم مربوط به این طرح ها و نتایج طرح بهینه مرجع [۴۰] در جدول (۵-۱۷) نشان داده شده است. همچنین مقدار میانگین تابع هدف حاصل از ۳۰ بار اجرای مجزای برنامه بهینه سازی سازه برای هر الگوریتم نیز در این جدول قرار داده شده است.

جدول (۵-۱۷) نتایج طرح بهینه خرپای ۲۰۰ عضوی با ۲۹ گروه

شماره گروه	GA	ACO	HS	CSS	MSM GA	Toğan et.al[40]	روش پیشنهادی
1	0.1	0.1	0.347	0.1	0.1	0.347	0.1
2	0.954	0.954	0.954	0.954	0.954	1.081	1.081
3	0.1	0.1	0.1	0.347	0.347	0.1	0.347
4	0.1	0.1	0.1	0.1	0.1	0.1	0.1
5	3.565	2.142	2.142	2.142	2.697	2.142	2.142
6	0.347	0.347	0.347	0.347	0.347	0.347	0.347
7	0.1	0.347	0.347	0.1	0.1	0.1	0.1
8	3.565	3.131	3.813	3.131	3.131	3.565	3.131
9	0.1	0.347	0.1	0.1	0.1	0.347	0.1
10	4.805	4.805	4.805	4.805	4.805	4.805	4.805
11	0.347	0.539	0.44	0.539	0.539	0.44	0.44
12	1.333	0.1	0.1	0.347	0.1	0.44	0.1
13	4.805	5.952	5.952	5.952	5.952	5.952	5.952
14	1.081	0.347	0.44	0.1	0.1	0.347	0.1
15	6.572	6.572	7.192	6.572	6.572	6.572	6.572
16	1.333	0.954	0.954	0.954	0.44	0.954	0.539
17	0.347	0.347	0.44	0.539	0.954	0.347	0.44
18	8.525	8.525	8.525	8.525	8.525	8.525	8.525
19	0.1	0.1	0.1	0.1	0.1	0.1	0.1
20	10.85	9.3	9.3	9.3	8.525	9.3	9.3
21	0.954	1.174	1.081	1.174	0.954	0.954	0.954
22	0.347	1.081	0.954	0.954	0.347	1.764	0.539
23	10.85	13.33	13.33	13.33	10.85	13.33	10.85
24	0.954	0.954	0.539	0.539	0.1	0.347	0.44
25	13.33	13.33	13.33	13.33	14.29	13.33	13.33
26	1.333	2.142	2.142	2.142	0.954	2.142	1.333
27	8.525	3.565	3.813	3.813	7.192	4.805	7.192
28	10.85	8.525	8.525	8.525	13.33	9.3	10.85
29	14.29	17.17	17.17	17.17	13.33	17.17	14.29
Best Weight(lb)	29137.4	27829.8	27941.4	27779.9	28074.3	28544.1	27233.8
Average(lb)	31828.7	29559.9	29751.7	29644.1	30515	----	28388.3
Volume(in ³)	102959	98338.3	98732.8	98162.2	99202.4	100862	96232.6

همچنین مرجع [۴۱] نیز وزن بهینه ۳۶۱۶۷.۷۳ پوند را برای این خرپا ارائه می دهد. همان طور که در جدول (۵-۱۷) مشاهده می شود بهترین طرح موجود در این جدول توسط روش پیشنهادی بدست آمده است. سایر روش ها نیز در قیاس با مراجع ذکر شده طرح بهینه مناسبی را ارائه می دهند.

در شکل (۵-۹) نمودار های همگرایی حاصل از میانگین ۳۰ بار اجرای مجزای برنامه بهینه سازی سازه با استفاده از الگوریتم ژنتیک جزیره ای، روش پیشنهادی و روش های به کار گرفته شده در سایر جزایر به طور مجزا برای خرپای ۲۰۰ عضوی با ۲۹ گروه نشان داده شده است.

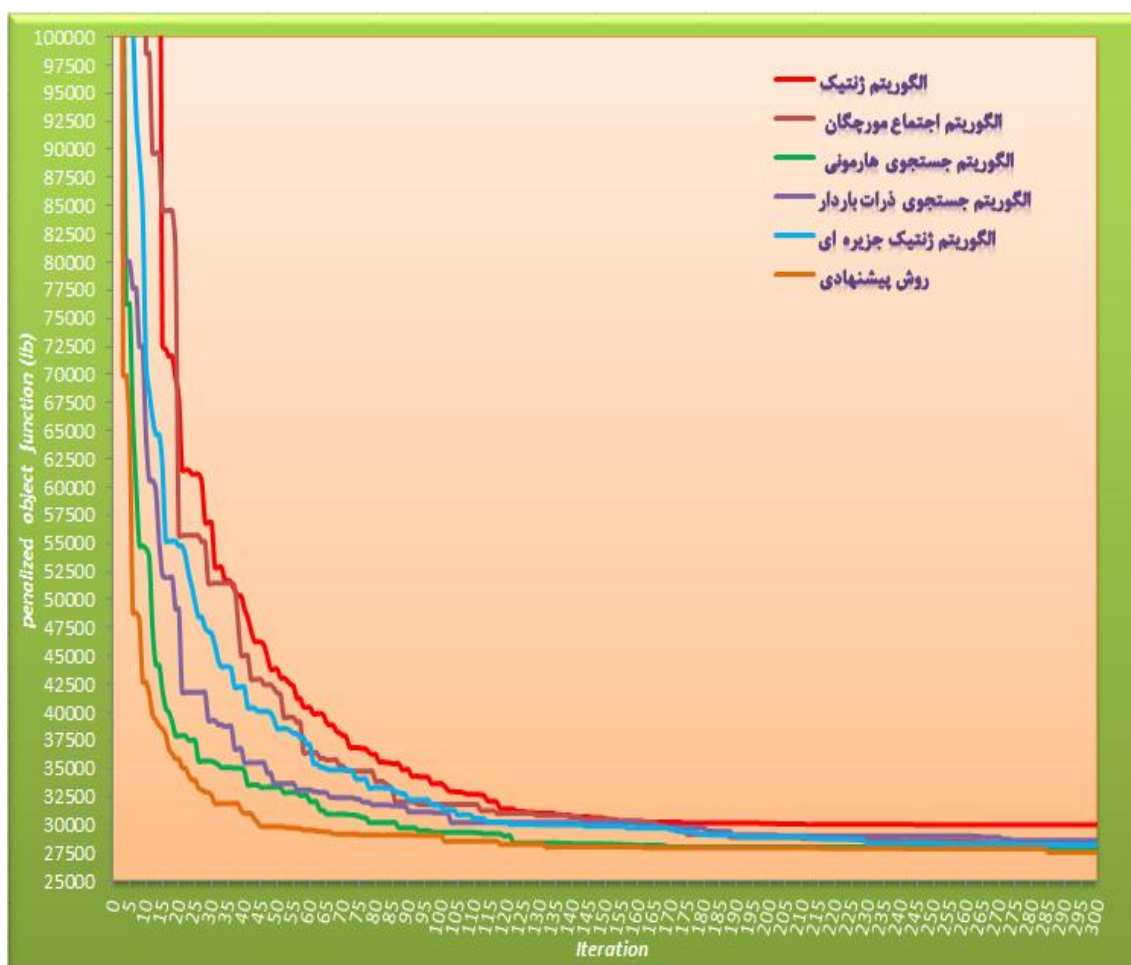


شکل (۵-۹) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر

الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۲۰۰ عضوی ۲۹ گروهی

همان طور که مشاهده می شود؛ روش پیشنهادی که ترکیبی از سایر الگوریتم ها می باشد با سرعت همگرایی مناسب به سمت نقطه کمینه حرکت می نماید و با کاوش هدفمند فضای طراحی جواب های نزدیک به جواب بهینه مطلق را بدست می آورد.

در شکل (۵-۱۰) نمودار های همگرایی برای بهترین جواب های حاصل از اجرای برنامه توسط سایر روش ها برای خریای ۲۰۰ عضوی با ۲۹ گروه ارائه شده است.



شکل (۵-۱۰) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خریای ۲۰۰ عضوی با ۲۹ گروهی

۵-۴ مثال (۴)

در این مثال به بررسی نحوه عملکرد روش پیشنهادی و سایر الگوریتم ها در بهینه یابی خرپای ۲۰۰ عضوی با ۹۶ گروه، که خرپایی با فضای طراحی بزرگ می باشد، می پردازیم. این خرپا نیز در شکل (۵-۸) نشان داده شده است.

برای کلیه اعضاء خرپا E و ρ به شرح زیر تعریف می گردند:

$$E = 30000Ksi \quad \& \quad \rho = 0.283(lb/in^3)$$

در این مثال سازه در معرض قیود تنش و تغییر مکان قرار گرفته است. تنش های کششی و فشاری مجاز برای کلیه اعضاء خرپا به شرح زیر می باشند:

$$\sigma_{all(Ten)} = +30Ksi \quad \& \quad \sigma_{all(Com)} = -30Ksi$$

همچنین تغییر مکان (جابه جایی) مجاز سایر گر ه های سازه در تمام جهات برابر است با:

$$\delta_{all}^i = 0.5in \quad i = 1, 2, 3, \dots, 77$$

بارهای اعمال شده بر سازه همانند بارگذاری وارد بر خرپای ۲۰۰ عضوی با ۲۹ گروه می باشد که در جداول (۵-۱۲) تا (۵-۱۴) نشان داده شده است.

برای این مثال سطح مقطع های عرضی نیز از مقادیر گسسته جدول (۵-۱۵) انتخاب می گردند.

با توجه به شکل (۵-۸) اعضاء این خرپا در ۹۶ گروه همان طور که در جدول (۵-۱۸) نشان داده شده است دسته بندی می شوند.

پارامترهای مربوط به روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری مطابق جدول (۵-۱۹) در نظر گرفته می شوند.

جدول (۵-۱۸) گروه بندی اعضا ی خریای ۲۰۰ میله ای (۹۶گروه)

شماره اعضا	گروه	شماره اعضا	گروه	شماره اعضا	گروه	شماره اعضا	گروه
157 , 169	76	104 , 112	51	47 , 51	26	1 , 4	1
158 , 168	77	105 , 111	52	48 , 50	27	2 , 3	2
159 , 167	78	106 , 110	53	49	28	5 , 17	3
160 , 166	79	107 , 109	54	57 , 58 , 61 , 62	29	6 , 16	4
161 , 165	80	108	55	59 , 60	30	7 , 15	5
162 , 164	81	115 , 118	56	64 , 76	31	8 , 14	6
163	82	116 , 117	57	65 , 75	32	9 , 13	7
171 , 172 , 175 , 176	83	119 , 131	58	66 , 74	33	10 , 12	8
173 , 174	84	120 , 130	59	67 , 73	34	11	9
178 , 190	85	121 , 129	60	68 , 72	35	18 , 25 , 56 , 63 , 94 , 101 , 132 , 139 , 170 , 177	10
179 , 189	86	122 , 128	61	69 , 71	36	19 , 20 , 23 , 24	11
180 , 188	87	123 , 127	62	70	37	21 , 22	12
181 , 187	88	124 , 126	63	77 , 80	38	26 , 38	13
182 , 186	89	125	64	78 , 79	39	27 , 37	14
183 , 185	90	133 , 134 , 137 , 138	65	81 , 93	40	28 , 36	15
184	91	135 , 136	66	82 , 92	41	29 , 35	16
191 , 194	92	140 , 152	67	83 , 91	42	30 , 34	17
192 , 193	93	141 , 151	68	84 , 90	43	31 , 33	18
195 , 200	94	142 , 150	69	85 , 89	44	32	19
196 , 199	95	143 , 149	70	86 , 88	45	39 , 42	20
197 , 198	96	144 , 148	71	87	46	40 , 41	21
		145 , 147	72	95 , 96 , 99 , 100 ,	47	43 , 55	22
		146	73	97 , 98	48	44 , 54	23
		153 , 156	74	102 , 114	49	45 , 53	24
		154 , 155	75	103 , 113	50	46 , 52	25

جدول (۵-۱۹) پارامتر های مرتبط با روش پیشنهادی در خرپای ۲۰۰ عضوی ۹۶ گروهی

جزیره (۴) CSS	جزیره (۳) HS	جزیره (۲) ACO	جزیره (۱) GA
پارامتر a (شعاع): 0.95	پارامتر $HMCR$: 0.94	پارامتر α : 0.55	حداکثر نرخ پیوند: 0.85
پارامتر $CMCR$: 0.93	پارامتر HMS : 40	پارامتر β : 0.15	حداقل نرخ پیوند: 0.8
پارامتر CMS : 25	حداکثر پارامتر PAR : 0.85	تعداد مورچه ممتاز: 10	حداکثر نرخ جهش: 0.005
پارامتر ΔT : 10 پارامتر K_t : 0.9	حداقل پارامتر PAR : 0.55	نرخ تبخیر سراسری: 0.1	حداقل نرخ جهش: 0.005
حداکثر پارامتر PAR : 0.85	-----	پارامتر تبخیر محلی: 0.6	روش انتخاب: تورنمنت
حداقل پارامتر PAR : 0.85	-----	پارامتر τ_0 : 100	سایز انتخاب: 3
نرخ مهاجرت: 40		وقفه مهاجرت: 25	
تعداد تکرار: 500	ثابت K : 4		تعداد اعضای جمعیت: 200

نتایج حاصل از طرح بهینه ی هریک از الگوریتم ها به همراه مقادیر توابع هدف و حجم مربوط به این طرح ها و نتایج طرح بهینه مرجع [۴۲] در جدول (۵-۲۰) نشان داده شده است. همچنین مقدار میانگین تابع هدف حاصل از ۳۰ بار اجرای مجزای برنامه بهینه سازی سازه برای هر الگوریتم نیز در این جدول قرار داده شده است.

جدول (۵-۲۰) نتایج طرح بهینه خربای ۲۰۰ عضوی با ۹۶ گروه

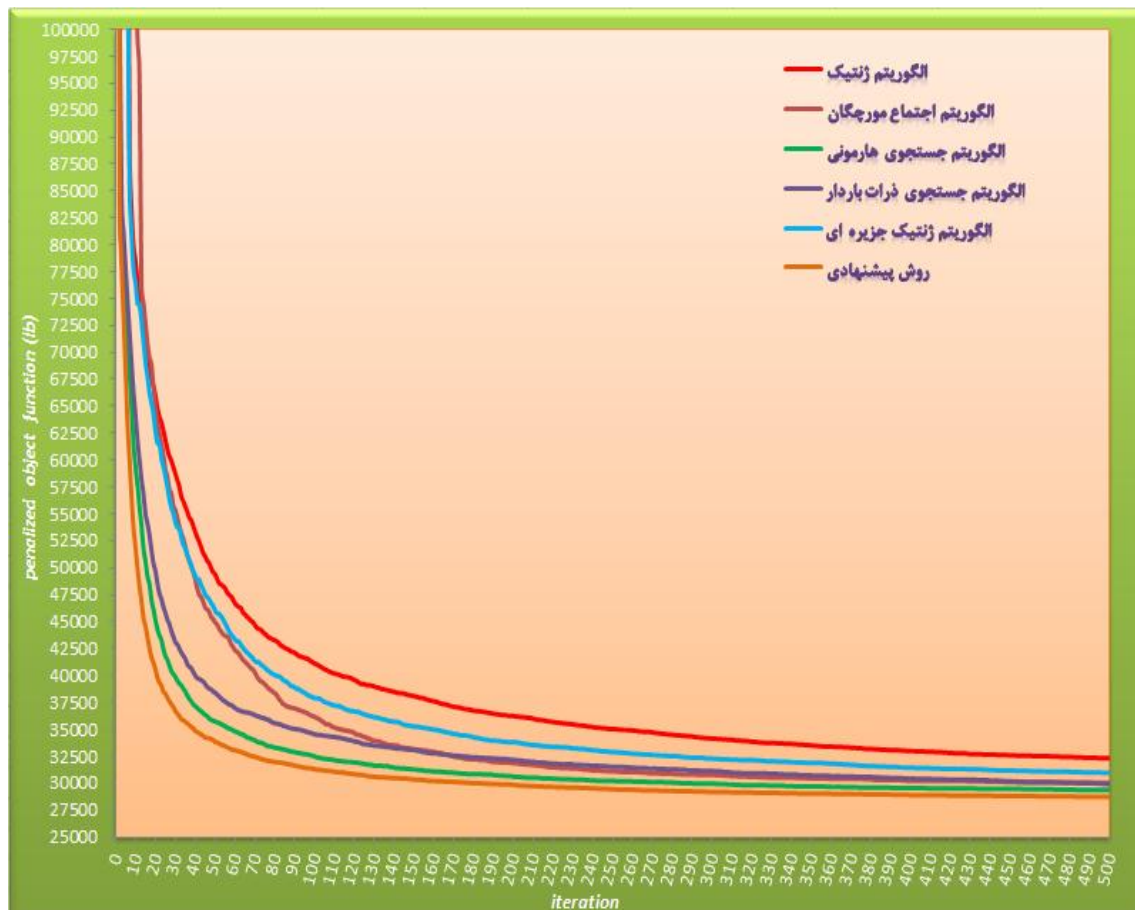
شماره گروه	GA	ACO	HS	CSS	MSM GA	Thierauf et.al[42]	روش پیشنهادی
1	3.565	0.954	1.488	1.764	1.764	-----	1.174
2	1.764	1.174	2.142	1.081	2.8	-----	2.8
3	3.131	2.697	1.764	3.131	3.565	-----	1.764
4	1.488	2.8	3.813	3.131	3.813	-----	3.131
5	0.1	1.174	0.44	0.539	0.44	-----	0.44
6	1.488	1.333	1.764	2.8	2.697	-----	1.333
7	1.333	0.539	0.1	1.174	0.1	-----	0.44
8	1.081	0.1	2.142	1.488	0.44	-----	0.954
9	2.697	1.764	1.333	0.954	3.565	-----	2.142
10	0.1	0.539	0.1	0.1	0.347	-----	0.1
11	0.539	0.539	0.44	0.954	0.1	-----	0.539
12	0.1	1.174	0.539	1.333	0.539	-----	0.44
13	3.565	5.952	4.805	3.813	3.131	-----	4.805
14	0.44	0.539	0.539	1.081	0.44	-----	0.539
15	2.697	3.813	3.131	3.813	2.8	-----	3.565
16	2.697	1.764	2.142	2.142	3.813	-----	1.764
17	1.488	0.347	1.174	0.954	0.1	-----	1.081
18	1.174	1.488	0.347	1.081	0.44	-----	0.44
19	4.805	5.952	3.131	2.8	3.565	-----	3.813
20	1.333	1.764	1.488	1.764	3.565	-----	2.142
21	0.1	0.1	0.539	0.954	0.1	-----	0.539
22	1.764	1.174	1.174	1.174	0.954	-----	1.333
23	2.8	3.565	3.565	4.805	3.131	-----	3.131
24	0.1	1.081	0.1	0.954	1.488	-----	0.44
25	9.3	9.3	8.525	8.525	4.805	-----	6.572
26	0.1	1.174	0.954	0.539	0.347	-----	0.44
27	1.764	0.347	1.488	0.954	3.565	-----	3.565
28	1.488	4.805	1.174	2.142	2.142	-----	0.954
29	0.1	0.539	0.347	0.539	0.44	-----	0.1
30	0.1	1.488	0.954	0.347	0.539	-----	1.333
31	3.813	2.697	3.131	2.8	4.805	-----	3.131
32	0.347	1.174	0.954	1.764	1.764	-----	0.347
33	5.952	6.572	3.813	3.131	3.565	-----	3.813

34	7.192	6.572	5.952	7.192	4.805	-----	8.525
35	3.565	1.333	1.488	1.081	3.565	-----	2.142
36	0.539	1.081	0.347	1.333	0.44	-----	0.347
37	2.697	3.813	2.697	2.697	4.805	-----	1.174
38	1.764	1.174	1.333	1.174	2.8	-----	1.174
39	1.488	0.1	0.347	1.488	0.954	-----	0.539
40	3.131	1.764	1.488	1.488	0.954	-----	2.142
41	2.697	3.565	2.142	3.813	2.697	-----	2.8
42	0.1	1.333	0.44	0.347	0.1	-----	0.1
43	7.192	13.33	14.29	14.29	9.3	-----	10.85
44	0.1	0.44	0.347	0.539	0.347	-----	0.954
45	1.333	2.142	0.347	2.142	3.565	-----	1.081
46	1.081	3.131	2.142	1.488	0.954	-----	1.764
47	0.539	0.1	0.347	0.1	0.347	-----	0.347
48	0.44	0.347	0.539	0.44	0.1	-----	0.1
49	6.572	3.565	3.565	2.142	3.565	-----	3.131
50	0.1	1.174	0.44	0.539	1.174	-----	0.539
51	3.565	3.565	3.131	3.813	6.572	-----	2.8
52	14.29	13.33	14.29	10.85	19.18	-----	14.29
53	3.813	1.764	1.174	1.488	1.488	-----	0.954
54	0.1	0.44	0.347	1.174	0.539	-----	0.44
55	3.565	6.572	2.142	2.142	2.142	-----	3.565
56	1.174	0.44	1.488	1.174	1.333	-----	1.081
57	1.488	0.1	0.539	0.1	0.1	-----	0.539
58	2.697	2.697	1.081	1.174	1.488	-----	1.081
59	2.8	1.488	2.697	1.764	1.174	-----	2.142
60	1.174	1.488	0.347	0.1	0.1	-----	0.1
61	13.33	17.17	19.18	23.68	17.17	-----	14.29
62	1.333	0.539	0.347	1.174	0.44	-----	0.539
63	2.142	1.081	1.488	0.539	0.954	-----	1.174
64	1.488	2.142	2.697	2.697	0.347	-----	2.142
65	0.347	0.1	0.44	0.347	0.44	-----	0.347
66	0.1	1.081	1.488	0.539	0.347	-----	0.347
67	3.565	3.131	3.565	2.142	3.565	-----	2.142
68	0.347	0.954	0.539	0.1	0.1	-----	0.347
69	3.813	1.333	1.764	2.142	3.813	-----	3.131
70	19.18	19.18	17.17	19.18	19.18	-----	17.17
71	1.174	1.081	2.142	0.44	1.333	-----	1.333

72	0.1	1.488	0.1	0.954	0.539	-----	0.44
73	3.813	4.805	3.565	1.764	1.081	-----	2.697
74	3.813	0.44	1.488	2.142	1.174	-----	1.174
75	0.1	0.347	0.44	1.174	0.1	-----	0.44
76	0.44	1.488	1.081	1.333	1.081	-----	1.174
77	2.697	1.488	2.697	2.697	0.954	-----	2.8
78	0.1	0.44	1.174	0.954	0.44	-----	0.347
79	17.17	19.18	19.18	28.08	23.68	-----	19.18
80	1.764	0.539	1.333	0.539	1.333	-----	0.954
81	0.44	0.954	2.142	1.333	1.764	-----	1.174
82	5.952	6.572	1.764	0.44	1.174	-----	2.142
83	0.1	1.174	0.347	0.1	0.539	-----	0.44
84	0.1	0.347	0.954	1.488	1.333	-----	0.44
85	2.697	1.764	1.764	1.174	3.565	-----	1.488
86	0.1	1.174	0.347	0.44	0.347	-----	0.44
87	4.805	1.333	2.142	2.8	1.081	-----	2.697
88	23.68	17.17	19.18	23.68	23.68	-----	23.68
89	0.44	0.44	2.697	1.764	0.954	-----	1.333
90	2.697	0.347	1.081	1.488	1.081	-----	1.488
91	1.333	7.192	1.488	2.8	1.488	-----	2.8
92	1.174	2.8	1.174	1.488	1.764	-----	1.174
93	0.1	1.764	0.44	0.44	0.44	-----	0.954
94	3.131	3.131	2.8	2.142	1.333	-----	2.697
95	28.08	23.68	28.08	23.68	28.08	-----	28.08
96	3.565	5.952	3.565	2.697	2.8	-----	3.565
Best Weight(lb)	29902.9	29250.1	28279.8	28984.1	29386.5	29641	27745.6
Average(lb)	32294.6	30024.56	29351.42	29907.65	30955.39	----	28701.18
Volume(in³)	105664	103357.1	99928.5	102417.4	103839.3	104730	98041.1

همان طور که از این جدول مشاهده می شود طرح بهینه حاصل از روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری، سبک تر از طرح های بدست آمده توسط الگوریتم های به کار گرفته شده در سایر جزایر و همچنین روش ژنتیک جزیره ای و مرجع ذکر شده می باشد.

در شکل (۵-۱۱) نمودار های همگرایی حاصل از میانگین ۳۰ بار اجرای مجزای برنامه بهینه سازی سازه با استفاده از الگوریتم ژنتیک جزیره ای، روش پیشنهادی و روش های به کار گرفته شده در سایر جزایر به طور مجزا برای خرپای ۲۰۰ عضوی با ۹۶ گروه نشان داده شده است.



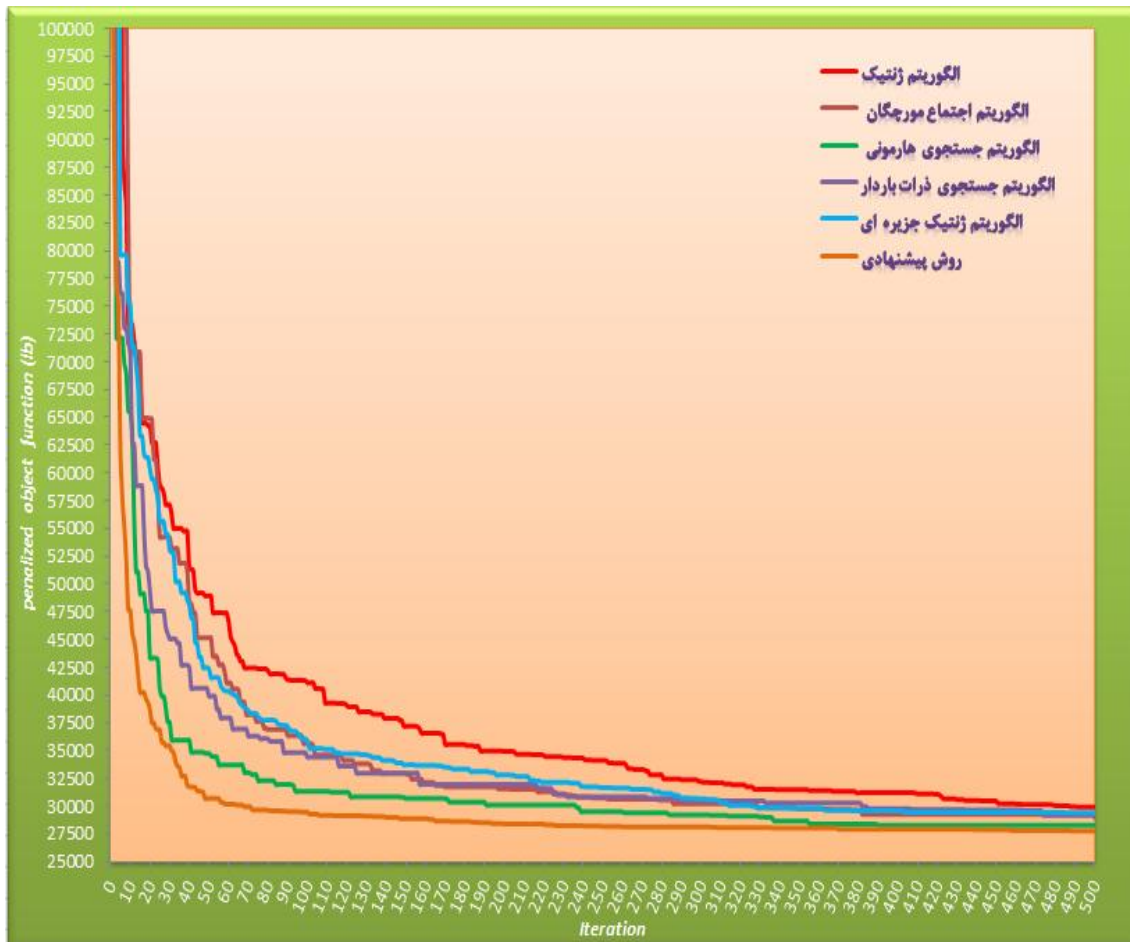
شکل (۵-۱۱) روند بهینه یابی توسط الگوریتم ژنتیک جزیره ای، روش جستجوی چند منظوره مبتنی بر

الگوریتم های فرا ابتکاری و هر یک از جزایر به طور مجزا برای خرپای ۲۰۰ عضوی ۹۶ گروهی

با توجه به شکل (۵-۱۱) در می یابیم که جزایر (۲) و (۳) و (۴) عملکرد مناسبی در نیل به سوی نقطه کمینه دارند و جزیره (۱) روندی آهسته و پیوسته را برای رسیدن به نقطه کمینه در پیش می گیرد. روش ژنتیک جزیره ای نیز عملکرد بهتری نسبت به جزیره (۱) دارد و همچنین روش پیشنهادی که ترکیبی از جزایر به کار گرفته شده می باشد روند همگرایی مناسب تری را نتیجه می دهد.

در شکل (۵-۱۲) نمودار های همگرایی برای بهترین جواب های حاصل از اجرای برنامه توسط

سایر روش ها برای خرپای ۲۰۰ عضوی با ۹۶ گروه ارائه شده است.



شکل (۵-۱۲) روند بهینه یابی توسط سایر روش ها برای بهترین جواب های حاصل برای خرپای ۲۰۰ عضوی ۹۶ گروهی

۵-۵ مثال (۵)

در این مثال به بهینه یابی وزن خرپای فضایی ۹۴۲ عضوی (برج ۲۶ طبقه) به عنوان یک

خرپای بسیار بزرگ با استفاده از روش های ارائه شده در این پایان نامه می پردازیم. این خرپا و

نماهای مختلف آن در شکل های (۵-۱۳) و (۵-۱۴) نشان داده شده است.

برای کلیه اعضا خرپا E و ρ به شرح زیر تعریف می گردند:

$$E = 29000 \text{ Ksi } (2038936 \text{ Kg} / \text{Cm}^2) \quad \& \quad \rho = 0.283 \left(\frac{\text{lb}}{\text{in}^3} \right)$$

همچنین تغییر مکان (جابه جایی) مجاز سایر گره های سازه در تمام جهات برابر است با:

$$\delta_{all}^i = 15 \text{ in} \quad i = 1, 2, 3, \dots, 232$$

محدودیت های تنش و کمانش بر اساس مباحث فصل سوم و مرجع [۴۳] در نظر گرفته

می شود.

تنش تسلیم سایر اعضا برابر است با:

$$F_Y = 36 \text{ Ksi} (2531 \text{ Kg / Cm}^2)$$

در این مثال رفتار گسسته مسئله مورد بررسی قرار می گیرد. بدین منظور اعضای خرپا از

لیست مقاطع (W_SHAPE) مرجع [۴۳] که شامل ۲۹۵ مقطع گسسته می باشد انتخاب می گردند.

این خرپا در ۵۹ گروه دسته بندی می گردد که این که گروه بندی بر اساس شماره گذاری

مشخص شده در شکل (۵-۱۳) صورت می گیرد.

این سازه در معرض بارهای قائم و افقی به شرح ذیل قرار می گیرد:

(۱) بار قائم اعمال شده بر سایر گره های مقاطع ۱ و ۲ و ۳ به ترتیب برابر است با:

$$P_z (\text{sec } 1) = -3 \text{ Kips} (-13.344 \text{ KN})$$

$$P_z (\text{sec } 2) = -6 \text{ Kips} (-26.688 \text{ KN})$$

$$P_z (\text{sec } 3) = -3 \text{ Kips} (-13.344 \text{ KN})$$

(۲) بار جانبی وارد شده بر سایر گره های این خرپا در راستای y برابر است با:

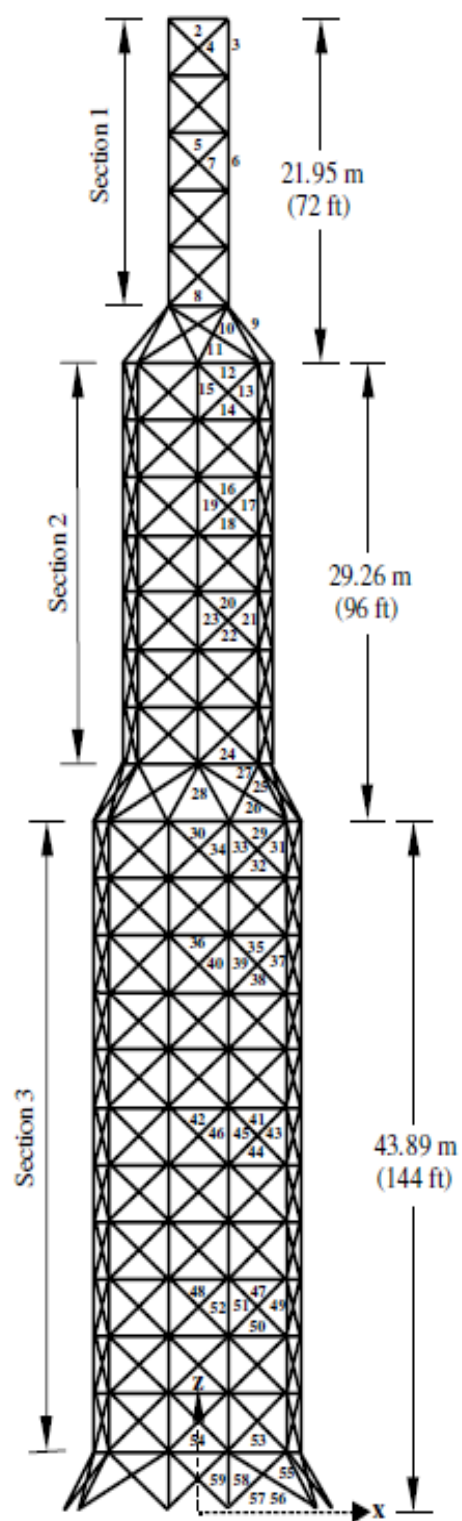
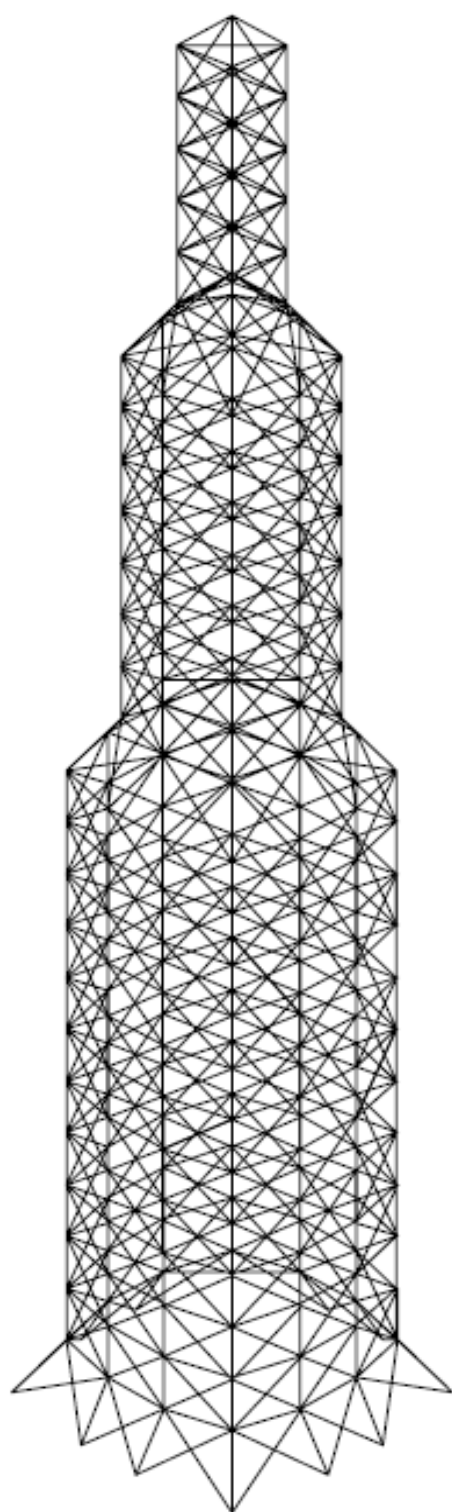
$$P_y = 1 \text{ Kips} (4.448 \text{ KN})$$

(۳) بار جانبی وارد شده بر سایر گره های سمت چپ سازه در راستای x برابر است با:

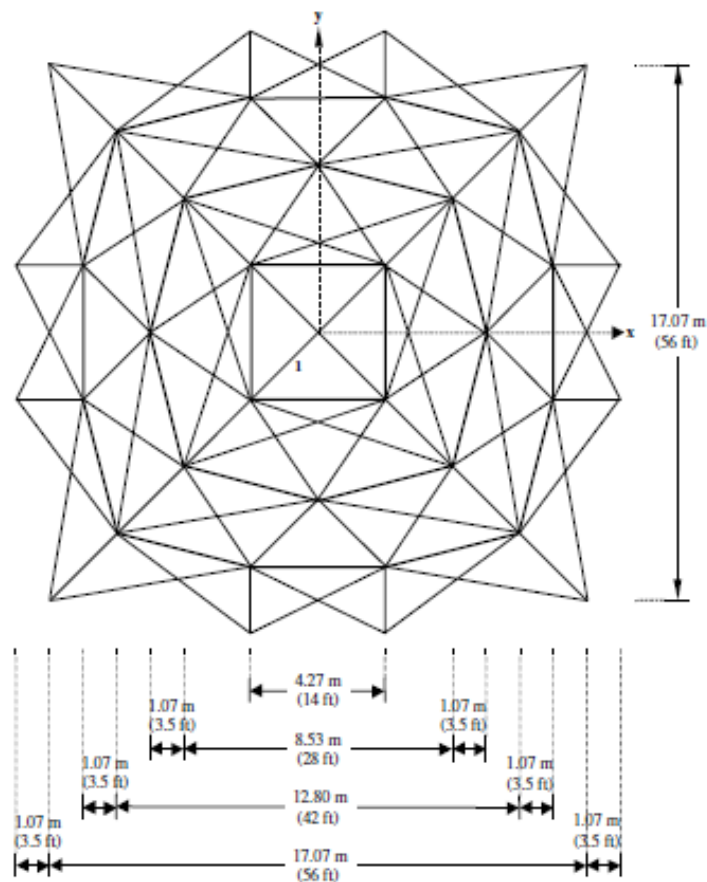
$$P_{x(\text{left})} = 1.5 \text{ Kips} (6.672 \text{ KN})$$

(۴) بار جانبی وارد شده بر سایر گره های سمت راست سازه در راستای x برابر است با:

$$P_{x(\text{right})} = 1 \text{ Kips} (4.448 \text{ KN})$$



شکل (۵-۱۳) نمای سه بعدی و جانبی خرابای فضایی ۹۴۲ عضوی



شکل (۵-۱۴) نمای از بالای خرپای فضایی ۹۴۲ عضوی

مشخصات و پارامترهای جزایر مرتبط با روش پیشنهادی برای خرپای ۹۴۲ عضوی مطابق شکل‌های (۴-۱۷) تا (۴-۲۴) می‌باشد.

در این خرپا با توجه به زیاد بودن تعداد قیود تنش، کمانش و جابه‌جایی و همچنین لیست مقاع‌اعضاء فضای طراحی بسیار بزرگ و پیچیده می‌باشد. در این صورت برای دستیابی به جواب‌های بهینه مناسب باید فضای طراحی به طور مناسب مورد کاوش و بررسی قرار گیرد.

نتایج حاصل از طرح بهینه‌ی هریک از الگوریتم‌ها به همراه مقادیر توابع هدف و حجم مربوط به این طرح‌ها و نتایج طرح بهینه مرجع [۴۴] در جدول (۵-۲۱) نشان داده شده است.

جدول (۵-۲۱) نتایج طرح بهینه خرپای ۹۴۲ عضوی

شماره گروه	GA	ACO	HS	CSS	MSM GA	Erbatur et.al[44]	Hasac et.al[45]	روش پیشنهادی
1	2.68	7.68	4.41	2.68	4.41	6.49	2.68	2.68
2	2.68	4.68	2.96	2.68	2.96	2.68	2.68	2.68
3	2.96	4.44	4.74	4.74	3.55	2.68	2.68	2.68
4	4.43	5.26	5.26	4.43	4.43	4.43	4.43	4.43
5	2.68	7.61	3.55	4.44	4.44	2.68	2.68	2.68
6	4.43	6.49	5.54	4.43	5.26	5.54	4.43	4.43
7	6.49	6.16	5.87	4.43	5.87	4.68	4.43	4.43
8	6.16	8.85	5.26	3.55	4.43	6.49	2.68	2.68
9	7.08	7.34	5.87	5.87	11.5	14.7	5.87	7.08
10	10.6	7.08	8.25	10.3	7.34	7.08	7.08	7.08
11	10.6	5.26	4.43	7.08	10.6	4.43	4.43	4.43
12	2.68	3.83	2.68	4.44	3.55	2.68	2.68	2.68
13	6.16	10	6.49	7.34	24.7	6.49	5.87	7.08
14	10.6	5.87	5.26	5.54	5.87	4.43	4.43	4.43
15	4.68	3.83	4.99	7.61	4.44	4.68	3.83	4.43
16	2.68	4.44	2.68	2.68	3.55	2.68	2.68	2.68
17	28.2	12.6	9.13	9.13	10.6	8.25	8.25	8.25
18	5.26	5.54	5.87	4.68	4.43	4.43	4.43	4.43
19	10.6	7.08	5.26	7.68	10.6	4.43	4.43	4.43
20	10.6	5.26	3.55	7.65	2.68	2.68	2.68	2.68
21	17.1	13.3	14.7	9.71	17.1	10.3	10.3	10.3
22	5.87	7.34	5.87	7.34	7.08	5.87	5.87	5.87
23	10.6	8.84	8.25	9.12	7.34	9.13	7.34	7.08
24	20.8	11.7	10.3	10.3	17.9	11.8	10.3	11.2
25	25.3	24.1	17.1	17.9	20.1	17.1	14.4	14.4
26	7.65	15.8	10.3	10.3	10.6	9.71	9.13	9.13
27	9.13	4.43	7.61	10	10	4.43	4.43	5.87
28	11.5	8.25	7.08	9.71	8.79	7.65	7.08	7.08
29	5.87	7.68	7.08	9.12	6.16	7.08	7.69	7.08
30	5.87	8.79	6.49	7.08	6.16	6.49	6.16	6.49
31	22.4	17.9	29.4	17.6	25.3	20	25.6	17.6
32	10.6	7.34	5.87	7.65	10.6	7.08	5.87	7.08

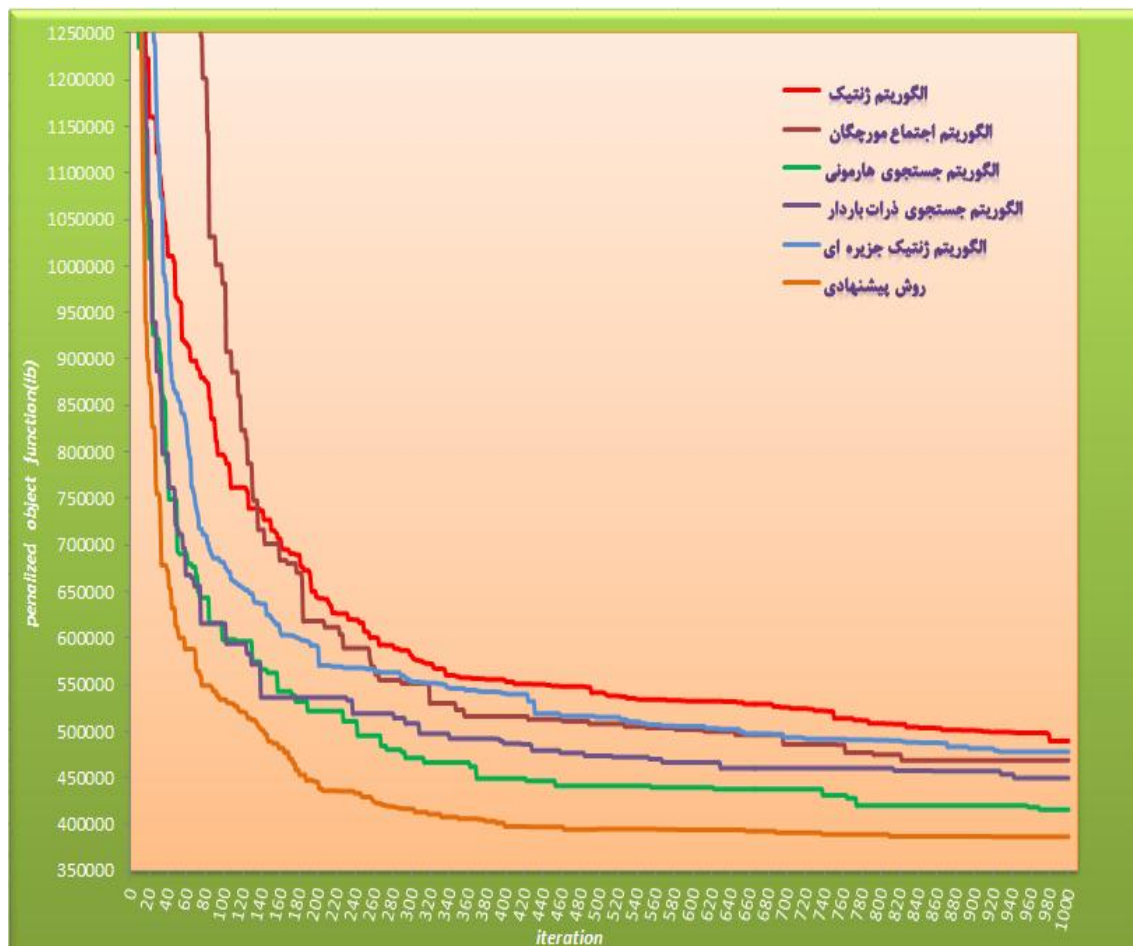
33	12.6	5.26	6.49	4.43	13.5	4.43	5.87	5.54
34	5.87	4.43	5.26	4.43	10.6	4.43	4.43	4.43
35	3.55	3.83	2.68	4.74	2.68	2.68	2.68	3.55
36	2.68	3.83	3.84	3.55	3.55	2.68	2.68	2.68
37	31.1	26.5	32	29.4	31.2	30.6	29.1	29.1
38	7.08	7.65	7.34	7.08	7.34	7.08	7.08	7.08
39	5.26	7.34	5.26	4.43	4.43	4.43	4.43	4.68
40	5.87	9.12	5.87	5.87	5.87	5.87	5.87	5.87
41	2.68	2.68	3.55	4.74	2.68	2.68	2.68	3.55
42	2.96	3.83	3.84	3.55	4.43	3.83	2.68	2.68
43	39.9	35.3	41.6	35.9	38.2	39.9	38.5	38.5
44	10.6	10.6	9.13	10.6	10.6	9.13	9.13	9.13
45	6.16	7.65	5.87	9.13	5.87	4.43	4.43	5.54
46	10.6	9.13	7.34	9.13	8.84	7.08	7.08	7.08
47	4.68	5.87	3.83	5.87	10.6	5.26	3.83	3.83
48	3.83	7.34	4.74	4.99	2.68	5.87	2.68	2.96
49	42.7	43.2	46.7	42.9	42.1	42.7	42.7	42.7
50	11.5	9.13	9.13	9.71	11.5	9.13	9.13	9.13
51	10.6	10	7.65	11.5	8.25	5.87	8.25	8.25
52	10.6	11.7	9.13	9.13	10	9.13	7.08	8.25
53	21.1	28.2	22.4	21.1	21.1	17.9	17.6	19.1
54	31.2	22.4	19.1	32.9	25.9	14.1	20.1	18.3
55	48.8	53.7	39.9	47.7	49.1	35.3	38.8	42.7
56	17.1	20.1	9.13	19.1	11.5	9.13	10.3	9.13
57	23.2	17.9	23.2	17.6	19.1	29.4	23.2	19.1
58	7.08	9.13	8.85	12.6	5.87	9.71	7.08	7.08
59	9.13	15.6	9.13	11.8	24.8	9.71	10.3	15.6
Weight (lb)	489598.1	468856.7	415833.7	449635.1	477923.9	394321	379660	386712.9
Volume (in ³)	1731054.2	1657719.7	1470247.9	1589758.3	1689778.4	1393595.5	1342350.1	1367286.9

همان طور که مشاهده می گردد طرح بهینه حاصل از اجرای روش پیشنهادی، از طرح های ایجاد

شده توسط هر یک از جزایر، روش ژنتیک جزیره ای و مرجع [۴۴] مناسب تر می باشد.

در شکل (۵-۱۵) نمودار همگرایی حاصل از اجرای برنامه بهینه سازی برای خرپای ۹۴۲ عضوی با

استفاده از الگوریتم ژنتیک جزیره ای، روش پیشنهادی و روش های به کار گرفته شده در سایر جزایر به طور مجزا نشان داده شده است.



شکل (۵-۱۵) روند بهینه یابی توسط سایر روش ها برای خرابای ۹۴۲ عضوی

همان طور که این نمودارها نشان می دهد روش پیشنهادی در مثال های بزرگ فضای طراحی را به طور مناسبی مورد کاوش و جستجو قرار داده و با سرعت همگرایی قابل توجه ای به سمت نقطه بهینه حرکت می نماید. جزیره (۱) سرعت همگرایی ضعیف تری نسبت به سایر جزایر دارد. جزیره (۲) در ابتدا با سرعت همگرایی کم تری به جستجوی کلی می پردازد و در ادامه روند متعادلی می گیرد. جزایر (۳) و (۴) نیز جستجوی کلی را به نحو مطلوبی انجام می دهند. الگوریتم ژنتیک جزیره ای نیز با سرعت همگرایی بهتری نسبت به جزیره (۱) درصد یافتن جواب های بهینه می باشد.

فصل ششم:

نتیجه گیری و پیشنهادات

۶-۱ مقدمه

در این فصل به بیان خلاصه ای از نتایج و دستاوردهای حاصل از این پایان نامه پرداخته می شود. همچنین با ارائه پیشنهاداتی، راه های آتی جهت ادامه پژوهش در این زمینه مطرح می گردد. هدف از این پایان نامه ارائه الگوریتم جستجوی چند منظوره مبتنی بر الگوریتم های فراابتکاری جهت بهینه یابی سازه های خرابایی با استعانت از روش های نوین الگوریتم ژنتیک، اجتماع مورچگان، جستجوی هارمونی و جستجوی ذرات باردار می باشد. در این راستا نتایجی بشرح زیر حاصل شده است.

۶-۲ نتایج

۱. روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری در نیل به نقطه کمینه دارای سرعت همگرایی قابل ملاحظه ای می باشد. بدین صورت در مسائل با فضای طراحی بزرگ مشکل سرعت بهینه یابی توسط سایر الگوریتم ها تا حدودی حل می شود.

۲. با بکارگیری الگوریتم های مختلف در چند جزیره، تعادل بین کاوش و تبعیت در دسترسی به جواب های بهینه فراهم شده است. به عبارت دیگر فضای طراحی با روش های مختلف بطور هوشمند و موثری مورد جستجو قرار می گیرد و این مسئله سبب می گردد تا حتی الا مکان روش جستجوی چند منظوره مبتنی بر الگوریتم های فراابتکاری در دام بهینه محلی نیفتد و به سمت بهینه مطلق حرکت نماید.

۳. در این پایان نامه از روش نخبه گرایی استفاده شده است بدین صورت که در جزیره (۱) بهترین کروموزوم در هر نسل برای نسل بعدی باقی می ماند. در جزیره (۲) بهترین طرح حاصل از حرکت مورچه در هر تکرار ذخیره می گردد. در جزایر (۳) و (۴) بهترین طرح های حاصل در حافظه هارمونی و حافظه بار ذخیره می گردند. این حالت سبب می گردد که طرح های مناسب به دست آمده از تکرارهای قبل دستخوش تغییرات نشده و سرعت بهینه یابی زیاد گردد.

۴. انتقال بهترین طرح های حاصل از هر جزیره در هنگام وقفه مهاجرت به جزایر دیگر و جایگزینی این طرح ها با طرح های نامناسب جزیره مقصد سبب بهبود عملکرد جزایر مقصد گردیده و همچنین مفهوم نخبه گرایی را نیز به طور غیر مستقیم اعمال می نماید.

۵. در هنگام وقفه مهاجرت و انتقال طرح های مهاجر به سایر جزایر، جستجوی محلی حول محدوده همگرایی هر یک از جزایر بخوبی انجام می شود و طرح های مناسب تری ایجاد می گردد.

۶. انتخاب الگوریتم های متنوع در جزایر مختلف و ترکیب آنها سبب افزایش قدرت کاربر در کسب نتیجه بهینه مطلوب می گردد.

۷. با به کارگیری روش پیشنهادی جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری از ایجاد جمعیت های یکنواخت با شایستگی های نزدیک که سبب قرار گیری الگوریتم در بهینه محلی می شود جلوگیری می گردد.

۳-۶ پیشنهادات

۱. بهینه یابی توپولوژی و هندسه خرپا بر اساس روش جستجوی چند منظوره مبتنی بر الگوریتم های

فرا ابتکاری

۲. استفاده از روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری در بهینه یابی دیگر

سازه ها همچون پوسته ها، سقف های مرکب، تیر ورق ها، قاب ها و سازه های فضاکار و.....

۳. ترکیب هر یک از الگوریتم های به کار گرفته شده در جزایر با روش های ریاضی و بررسی طرح

های حاصله از آن ها.

۴. بررسی تاثیر وقفه مهاجرت در دستیابی روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا

ابتکاری به طرح های بهینه.

۵. به کار گیری شکل بهبود یافته هر یک از الگوریتم ها در جزایر جهت نیل به نقطه کمینه سراسری.

۶. بررسی پارامتر های k_a و k_v در روش جستجوی ذرات باردار و تاثیر این پارامتر ها در روند

همگرایی و کیفیت طرح های بهینه.

۷. حل مسائل توابع چند هدفه با روش جستجوی چند منظوره مبتنی بر الگوریتم های فرا ابتکاری.

۸. استفاده از راه کارهایی جهت کاهش فضای جستجو جهت دستیابی سریع تر به جواب های بهینه.

۹. اضافه کردن فرآیندهای جستجوی محلی در روش جستجوی هارمونی و اجتماع مورچگان.

فہرست منابع

[۱] هفته‌آر. تی و گوردال زد. (۱۳۸۲). "مبانی بهینه‌سازی سازه‌ها". ابوالبشری م. ح. چاپ اول

، دانشگاه فردوسی مشهد ، مشهد.

[2] Vanderplaats, G.N.(1984) "Numerical Optimization Techniques For Engineering Design With Applications", McGraw-Hill. New York.

[3] Yang, X.S. (2008) "Nature-Inspired Metaheuristic Algorithms". Luniver Press

[4] Christensen, P. W and Klarbring. A.(2009)" An Introduction to Structural Optimization" Springer, pp 5

[5] Khorsand, A, R., Akbarzadeh-T, M, R.(2007)" Multi-Objective Meta Level Soft Computing-Based Evolutionary Structural Design" , Journal Of The Franklin Institute, pp 595–612

[6] Hassani B. And Hinton E.(1999)"Homogenization And Structural Topology Optimization". Springer.

[7] F. Glover, M. Laguna. (1997) "Tabu Search", Kluwer Academic Publishers, Dordrecht

[8] Blum, C., Roli, A. (2003) "Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Comput. Surv. 35,pp 268–308

[۹] علیرضا م. (۱۳۸۶). "مقدمه‌ای بر الگوریتم‌های ژنتیک و کاربردهای آن". چاپ دوم ،

شرکت ناقوس اندیشه ، تهران.

[۱۰] رضایی ع. و رنجبران س. (۱۳۸۶). "آموزش کاربردی الگوریتم ژنتیک در نرم افزار

Matlab". چاپ اول ، کتاب پدیده ، تهران.

[۱۱] کلات جاری و. ر. (۱۳۸۰)، رساله دکتری. "بهینه‌یابی خرپاها با استفاده از الگوریتم

ژنتیک و روش جبری نیروها". دانشکده عمران ، دانشگاه علم و صنعت ایران.

[۱۲] طالبپور. ح، ۱۳۸۷، پایان نامه کارشناسی ارشد: "بهینه‌یابی سطح مقطع و توپولوژی سازه‌های

خرپایی با استفاده از روش‌های نوین الگوریتم ژنتیک"، دانشکده عمران و معماری دانشگاه صنعتی

شاهرود.

[۱۳] باوی، ا.، صالحی، م. (۱۳۸۷). "الگوریتم ژنتیک و بهینه سازی سازه های مرکب". چاپ اول، انتشارات عابد، تهران.

[14] Rajeev S. And Krishnamoorthy C. S.(1992) "Discrete Optimization Of Structures Using Genetic Algorithms", ASCE, Journal Of Structural Engineering. **118**, 5, Pp. **1233-1250**.

[15] Goldberg D. E. .(1989) "**Genetic Algorithm in Search, Optimization and Machine Learning**". Addison-Wesley Publishing Co.

[16] Kaveh A. and Kalatjari V.(2002) "Genetic Algorithm for Discrete-Sizing Optimal Design of Trusses Using the Force Method", International Journal of Numerical Methods in Engineering. **55**, pp **55-72**.

[17] Haupt R. L. And Haupt E.(2004) "**Practical Genetic Algorithms**". Second Edition, John Wiley And Sons, New York.

[۱۸] فتاحی پ. (۱۳۸۸). " الگوریتم های فرا ابتکاری ". چاپ اول، انتشارات دانشگاه بوعلی سینا، همدان.

[19] Dorigo, M., Stutzle, T. (2004). "Ant Colony Optimization". Mit Press, Cambridge

[20] Dorigo, M., Blum, C. (2005) "Ant Colony Optimization Theory: A Survey". Theor. Comput. Sci. **344**,Pp **243–278**

[21] Cordon,O. Herrera,F. Stützle,T.(2002) "A Review On The Ant Colony Optimization Metaheuristic: Basis, Models And New Trends". Journal Of Mathware And Soft Computing.**9**, pp. **141-175**.

[22] Dorigo, M. Gambardella, L.M. "Ant Colony System: A Cooperative Learning Approach To The Traveling Salesman Problem".IEEE Transactions On Evolutionary Computation.**1**,Pp **53-66**

[23] Bonabeau, E., Dorigo, M., Theraulaz, G. (1999) "Swarm Intelligence: From Natural To Artificial Systems". Oxford University Press, Oxford

[24] Lee, K.S., Geem, Z.W., (2005) "A New Meta-Heuristic Algorithm For Continuous Engineering Optimization: Harmony Search Theory And Practice". Comput. Methods Appl. Mech. Engrg, **194**, Pp 3902–3933.

[25] Z.W. Geem, J.H. Kim, G.V. Loganathan, (2001) "A New Heuristic Optimization Algorithm: Harmony Search", Simulation, **76**, pp 60-68.

[۲۶] آهنگران. م، ۱۳۸۹، پایان نامه کارشناسی ارشد: "ترتیب و طراحی با استفاده از بهینه یابی هارمونی"، دانشکده عمران دانشگاه علم و صنعت.

[27] Halliday, D., Resnick, R., Walker, J. (2008) Fundamentals of Physics, 8th edn. Wiley, New York

[28] Kaveh A, Talatahari S. (2010) "A Novel Heuristic Optimization Method: Charged System Search, Acta Mechanica" DOI: 10.1007/S00707-009-0270-4.

[29] Kaveh A, Talatahari S. (2010) "Charged System Search For Optimum Grillage System Design Using The LRFD-AISC Code". Journal Of Constructional Steel Research **66**, pp 767_771

[30] Coello, C.A.C. (2002) "Theoretical And Numerical Constraint-Handling Techniques Used With Evolutionary Algorithms: A Survey Of The State Of The Art". Comput. Methods Appl. Mech. Eng. **191**(11–12), pp 1245–1287

[31] Kaveh, A., Talatahari, S.: (2009) "A Particle Swarm Ant Colony Optimization For Truss Structures With Discrete Variables". J. Constr. Steel Res. 65(8–9), 1558–1568

[32] Kaveh, A., Talatahari, S. (2009) "Particle Swarm Optimizer, Ant Colony Strategy And Harmony Search Scheme Hybridized For Optimization of Truss Structures". Comput. Struct. **87**(5–6), pp. 267–283

[33] Goldberg D. E. And Samtani M. P. (1986). "Engineering Optimization Via Genetic Algorithm". ASCE, Proceedings Of The Ninth Conference On Electronic Computations, pp 471-482

[۳۴] مقررات ملی ساختمان مبحث دهم (۱۳۸۹). "طرح و اجرای ساختمان های فولادی". چاپ

چهارم

[۳۵] پری گ. و هتیوا س. (۱۳۸۰). "آموزش ویژوال بیسیک ۶ در ۲۴ ساعت". احترامی ب.

چاپ اول، دانش نگار، تهران.

[36] Lee. K.S, Geem .Z.W, Lee.S.H, Bae.K.W., (2005) "The Harmony Search Heuristic Algorithm For Discrete Structural Optimization". Engineering Optimization, **37**, 7, Pp **663-684**.

[37] Wu,SJ., Chow,PT., (1995), "Steady-State Genetic Algorithms For Discrete Optimization Of Trusses,". Computers And Structures, **56**(6), Pp. **979-91**.

[38] Li, L.J. Huang, Z.B. Liu, F., (2009), "A Heuristic Particle Swarm Optimization Method For Truss Structures With Discrete Variables,". Computers And Structures, **87**, Pp. **435-443**.

[39] Kaveh A., Talatahari S. (2010), "A Charged System Search With A Fly To Boundary Method For Discrete Optimum Design Of Truss Structures,". Asian Journal Of Civil Engineering (Building And Housing) , **11**, NO. **3**

[40] Toğan,V.Daloğlu, T. A.(2008),"An Improved Genetic Algorithm With Initial Population Strategy And Self-Adaptive Member Grouping". Computers And Structures, **86**, Pp.**1204-1218**.

[41] Coello, C.A. Christiansen, A.D. (2000) , "Multiobjective Optimization Of Trusses Using Genetic Algorithms,". Computers And Structures, **75**, Pp. **647-660**.

[42] Thierauf, G. Cai, J. (1998), "Parallelization of the Evolution Strategy for Discrete Structural Optimization Problems,". Computer-Aided Civil and Infrastructure Engineering, **13**, pp. **23-30**.

[43]AISC-ASD (American Institute Of Steel Construction-Allowablestress Design). Manual Of Steel Construction-Allowable Stress Design;**1989**.

[44]Erbatur, F., Hasanc, Ebi O., Tu, " Tu " Ncu _I, Kılıc H. (2000) , "Optimal Design Of Planar And Space Structures With Genetic Algorithms,". Comput Struct,**75**,Pp,**209-224**.

[45] Hasanc, Ebi O, Erbatur F., (2002) , " On Efficient Use Of Simulated Annealing In Complex Structural Optimization Problems. ," Acta Mech;**157**,Pp.**27-50**.

Abstract:

Most of the optimization algorithms that are based on numerical linear and nonlinear programming methods, require the derivative information of the objective function and use those information for searching the range of solution in the approximately near the starting point. Because of that, there are limitations in usage of these algorithms for optimization problem in natural world. If there is more than one local optimum in the problem, the final result may depend on the value of starting point, and the obtained optimal solution may not necessarily be the global optimum. Therefore, in recent decades, application of metaheuristic algorithms for optimization has been attracted researcher's attention.

Because there is no need for heavy mathematical calculation, and having capability of reaching to the global optimal point, or at least near of that, these innovative searching methods, which are based on natural processes, are used to solve various kinds of problems.

Due to the significance role of optimization problems in engineering sciences use of metaheuristic algorithms for the solution of them, this thesis, by utilizing innovative algorithm methods, such as Genetic Algorithms (GA), Ant Colony Optimization (ACO), Harmony Search (HS) and Charge System Search (CSS), simultaneously proposed a innovative method for optimization of cross section of truss structures. By the way, the designing space of problem are investigated properly and the desired optimum solution are approximately achievable. However, by designing many different examples, the result from other algorithms and this proposed way, which is called multi-search method based on metheuristic algorithms, have been analyzed and the applicability of this method also demonstrated.

Keywords:

Optimization, Truss Structure, Genetic Algorithms, Ant Colony Optimization Harmony Search, Charge System Search, Multi Search Method Based On Metheuristic Algorithms.