طرح پژوهشی در قالب گرنت

عنوان طرح:

ارائه مدل جدیدی از الگوریتم های بهینه سازی برای افزایش بهروری ضریب اعتماد سیستم های ترابری

مجری طرح:
محمد محمدی نجف آبادی

خاتمه طرح:
شهریور ۱۳۹۶

فهرست مطالب | صفحه

فصل اول

مقدمه ای بر الگوریتم‌های تکاملی

۱-۱- مقدمه

الگوریتم‌های تکاملی شاخه‌ای از تکنیک‌های بهینه‌سازی اجتماعی هستند که در آنها از سیستم‌های بیولوژیکی و اصول حاکم بر آنها بهره‌گیری می‌شود. اگر چه این الگوریتم‌ها ارائه دهنده‌ی مدل‌های ساده‌ای از فرایندهای بیولوژیکی در شرایط واقعی هستند اما درعمل توانائی و کارایی زیادی را از خود نشان داده‌اند. ایده اولیه‌ی ارائه الگوریتم‌های تکاملی استفاده از جمعیت محدودی از عناصر است، به طوری که هر یک از آنها دقیقاً نقطه‌ای از فضای جستجو را مشخص می‌کنند. پس از ایجاد، این جمعیت اولیه در یک فرایند خودسامانی توسط فرایندهای آماری توسعه می‌یابد و در جهت یافتن مناطق بهتر و بهینه‌تر از فضای جستجو با یکدیگر ترکیب می‌شوند. فرایند انتخاب، عناصری را از لحاظ اطلاعات ژنتیک جدید تأمین کرده و عملگر بازترکیبی نیز اطلاعات را با هم ترکیب و کنترل می‌کند.

الگوریتم‌های تکاملی یکی از سیستم‌های حل مسئله با استفاده از کامپیوتر که از بعضی از مدل‌های تکاملی الگوبرداری شده است. [۷].

الگوریتم‌های تکاملی دارای انواع مختلف است که مهمترین آنها عبارتند از [۳]:

۱ـ برنامه تکاملی [Evolutionary programming (Fogl ،۱۹۶۶)]

۲ـ الگوریتم‌های ژنتیک [Genetic algorithm (Holland ،۱۹۷۵)]

۳ـ استراتژی‌های تکامل [Evolution strategic (Rechen berg ،۱۹۷۳)]

انواع الگوریتم‌های تکاملی به طور مفهومی تکامل ساختارهای افراد در طبیعت بر طبق روندهای انتخاب، جهش و باز ترکیبی می‌باشد. این عملگرها قابلیت ذخیره‌سازی ساختارهای افراد که به وسیله محیط تعریف شده، را دارا می‌باشند.

هر فرد در جامعه یک اندازه برازندگی در محیط دریافت می‌کند. کانون توجه عملگر بازترکیبی روی افراد با برازندگی بالا می‌باشد. بازترکیبی و جهش باعث آشفتگی می‌شوند.

الگوریتم‌های تکاملی اگرچه در بیشتر سطحها با هم شبیه هستند ولی در روش انجام با هم متفاوت هستند. این اختلاف بیشتر روی ویژگی‌های الگوریتم‌های تکاملی مشاهده می‌شود که شامل انتخاب اطلاعات برای ساختار افراد وانواع مکانیسم انتخاب زیرا اندازه قابلیت عملگرهای مختلف باهم متفاوت هستند.

انواع دیگر از الگوریتم‌های تکاملی که توسعه یافته‌تر از بقیه انواع دیگر است، نیز وجود دارد که عبارتند از[۱]:

ـ سیستم‌های طبقه‌بندی [Classifier systems (Holland ،۱۹۸۶]

ـ برنامه ژنتیک [Genetic programming (Garis ،۱۹۹۰ ؛Koza ،۱۹۹۱)]

ـ سیستم‌های LS [LS systems (Smith ،۱۹۸۳]

ـ سیستمهای عملگر وفقی ["adaptive operater" systems (Davis ،۱۹۸۹]

پایه‌های زیست‌شناسی [۳]

برای درک درست الگوریتم‌های تکاملی کافی است که رونـدهای زیست‌شناسـی کـه عملگرهـای الگوریتم‌های تکاملی براساس آن الگوبرداری شده است را مورد توجه قرار دهیم.

ابتدا ما باید در نظر بگیریم که تکامل (در طبیعت یا در هر چیز دیگر) یـک رونـد جهـت‌دار و بـا قصد و هدف می‌باشد. تکامل وسیله انتخاب طبیعی یا رقابت افراد مختلف بـرای منـابع محـدود در طبیعت است. افراد بهتر باقی می‌مانند و ژنهای آنها توسعه پیدا می‌کنند.

باز ترکیبی جنسـی بـه بعضـی از کروموزوم‌هـا اجـازه تولیـد فرزنـد را می‌دهـد. از نظـر ملکـولی بازترکیبی عبارت است از تغییر یا جابجائی قسمتی از کرومـوزوم بـه کرومـوزوم دیگـر اسـت. ایـن عملگر بازترکیبی از عمل crossover که باعث تغییر ژنها بین دو کروموزوم در طبیعـت می‌شـود الگوبرداری شده است.

عمل بازترکیبی در محیطی رخ می‌دهد که ابتدا افراد خـوب انتخـاب می‌شـوند و سـپس بـا هـم ترکیب می‌شوند تا افراد با برازندگی بالا تولید کند.

اغلب الگوریتم‌های تکاملی از یک تابع ساده به عنوان تابع برازندگی افراد برای انتخاب والدین (به طور تصادفی) که تحت تأثیر عملگرهای ژنتیک قرار گیرند مانند crossover استفاده می‌کنند.

تکامل نیاز به گوناگونی و تنوع در حین کار دارد. در طبیعت، یـک منبـع ایجادکننـده گونـاگونی مهم، جهش است. اهمیت جهش به خاطر ایجاد تنوع زیاد می‌باشد.

برنامه کلی الگوریتم‌های تکاملی به صورت زیر می‌باشد:

Algorithm EA is:

```
//start with an initial time
t:=0;
// initialize a usually random populayion of individuals
initpopulation p(t);
// evaluate fitness of all initial individuals in population
evaluate p(t);
//test for termination crierion (time, fitness,etc)
While not done do
//increase the time counter
t:=t+1;
// select sub-population for offspring production
p': =select parents p (t);
// recombine the "genes" of selected parents
recombine p' (t);
```

```
// perturb the mated population stochastically
mutate p' (t);
// evalutate its new fitness
evaluate p'(t);
// select the survivors from actual fitness
p: =survive p,p'(t);
do
end EA.
```

۲-۱ - برنامه‌ریزی تکاملی

برنامه‌ریزی تکاملی [۴]و[۵]، که به منظور دستیابی به اطلاعات لازم برای ماشین ایجاد شده، عمدتاً الگوئی را مورد استفاده قرار می‌دهد که با دامنه‌ی مسئله تناسب داشته باشد. فرض کنید جمعیتی به بزرگی N داشته باشیم که همه اعضای آن به عنوان والدین انتخاب می‌شوند وهمچنین الگوی عملگر جهش مشخصی برای تولید N نوزاد مورد استفاده قرار می‌گیرد. سپس همان N نوزاد تکامل خواهند یافت و نسل بعد با استفاده از یک تابع احتمال برازندگی برای تعداد ۲N عضو انتخاب خواهد شد. عملگر جهش در برنامه‌ریزی تکاملی معمولاً دارای قدرت تطابق با مسئله است.

```
Algorithm Ep is:
// start with an initial time
t: =0;
// initialize a usually random populayion of individuals
Initpopulation p(t);
// evaluate p(t);
// test for termination criterion (time, fitness, etc.)
While not done do
// perturb the whole population stochastically
p' (t): = mutate p(t);
//evaluate p' (t);
//stochastically select the survivors from actual fitness
p (t+1): =survive p (t), p' (t);
// increase the time counter
t:= t+1
do
end EP.
```

۳-۱- الگوریتم‌های ژنتیک

الگوریتم‌های ژنتیک [۴] یک مدل از الگوریتم‌های تکاملی است که رفتارش از مکانیسم‌های تکاملی در طبیعت الگوبرداری شده است به طوری که مکانیسم افراد جامعه به وسیله کروموزوم مشخص می‌شوند. جمعیت کروموزوم‌ها (افراد جامعه) بعد از این وارد مرحله شبیه تکامل می‌شوند.

الگوریتم‌های ژنتیک در سطح‌های مختلف کاربرد دارند. یک مثال از کاربردهای آنها برای مسائل بهینه‌سازی چندبعدی که کروموزوم‌های آن به وسیله‌ی مقادیر مختلف کدبندی می‌شوند تا بهینه شوند.

در عمل، محاسبات مدل ژنتیک روی آرایه‌های بیتی یا کارکترهایی که کروموزوم را مشخص می‌کند، انجام می‌شود. عملگرهای انجام کار روی یک بیت ساده اجازه انجام کار توسط عملگرهای crossover و جهش و دیگر عملگرها را می‌دهد.

الگوریتم‌های ژنتیکی به طریق و روش دوره‌ای زیر عمل می‌کنند:

ابتدا به طور تصادفی جامعه‌ای از کروموزوم‌ها پدید می‌آوریم و سپس برازندگی تمام کروموزوم‌های (افراد) داخل جامعه محاسبه و تعیین می‌شود. به وسیله عملگرهای crossover و جهش و دیگر عملگرها جامعه‌ای جدید به وجود می‌آوریم. تکرار یک بار حلقه بالا باعث به وجود آمدن یک نسل می‌شود. در هر با انجام حلقه، از جامعه قبلی صرفه‌نظر می‌شود و به جای آن جامعه جدید قرار می‌گیرد. هیچ دلیل برای این که، این مدل انجام می‌شود، وجود ندارد در حقیقت ما این مدل را مانند یک اصل و قانون نمی‌بینیم اما آن یک مدل و نسخه مناسب است.

نسل اول (نسل صفر) در حقیقت به طور تصادفی انتخاب می‌شوند و سپس با توجه به برازندگی افراد (کروموزوم‌ها) و عملگرهای موجود جامعه به طرف برازندگی بالا سوق پیدا می‌کند.

برنامه کلی الگوریتم‌های ژنتیک:

Algorithm GA is:

```
//start with an initial time
t:=0;
//initialize a usually random population of individuals
Initialize a usually random population of individuals
Initpopulation p(t);
//evaluate fitness of all initial individuals of population
Evaluate p (t);
//test for termination criterion (time, fitness, etc)
While not done do
```

```
//increase the time counter
T: =t+1;
// select a sub-population for offspring production
p' : =selectparents p(t);
 //recombine the "genes" of selected paterns
 Recombine p'(t);
 //perturb the mated population stochastically
 mutate p'(t);
//evaluate its new fitness
 evaluate p'(t);
 // select the survivors from actual fitness
   P:=survive p,p'(t);
   Do
 End GA.
```

۴-۱- استراتژی‌های تکاملی

یکی دیگر از انواع الگوریتم‌های تکاملی استراتژی‌های تکاملی [۳] و [۱] است. استراتژی‌های تکاملی الگوریتم‌هایی هستند که از قواعد تکامل طبیعی به عنوان یک راه حل در مورد مسائل بهینه‌سازی پارامتری استفاده می‌کنند. این استراتژی‌ها در دهه‌ی ۶۰ میلادی در آلمان رواج یافتند که در ادامه مورد بحث قرار می‌گیرد:

در سال ۱۹۶۳ دو دانشجوی دانشگاه فنی برلین به طور مشترک آزمایش‌هایی را انجام می‌دادند. آنها برای انجام این آزمایش‌ها از تونل هوای بخش مهندسی سیالات استفاده می‌کردند. در حین تحقیقات برای ایجاد سازه‌های بهینه در یک جریان هوا، این ایده به ذهن دانشجویان خطور کرد که این مسئله را به عنوان یک استراتژی مطرح کنند. اما تلاش‌هائی که برای استراتژی ساده و مشابه به کار گرفته شدند، همگی ناموفق بودند. سپس یکی از دانشجویان به نام ایگنوریچنبرگ که اکنون بیونیک و مهندسی تکامل است، به این فکر افتاد که تغییرات تصادفی پارامترهای معرف سازنده را مورد آزمایش قرار دهد و مثال‌های جهش‌های طبیعی را پیگیری کند بدین ترتیب استراتژی تکاملی شکل گرفت.

استراتژی‌های تکاملی ابتدائی، تقریباً همانند برنامه‌های تکاملی بودند که در آنها از یک شیوه‌ی عددی متغیر استفاده می‌شد و عملگر جهش صرفاً یک عملگر ترکیبی به شمار می‌رفت. آنها در بسیاری از مسائل بهینه‌سازی کاربرد دارند و پارامترهای آنها مدام قابل تغییر می‌باشد. اخیراً این استراتژی‌ها برای مسائل دیگر هم به کار رگفته شده‌اند که اجزای اصلی این استراتژی عبارتند از:

برازندگی

برازندگی یک عنصر، میزانارزشی است که آن عنصر تحت تابع هدف موردنظر دارد. بهترین عناصر آنهایی هستند که با استفاده از آنها تابع هدف به سمت بهینه‌ترین مقدار میل می‌کند.

جمعیت اولیه

جمعیت اولیه یا جهش از یک نقطه دیگر حرکت می‌کند که خود این نقطه ممکن است یا به طور تصادفی و یا به طور تقریبی از اطلاعات کلی از منطقه‌ای که جواب در آن واقع شده، انتخاب گردد.

جهش

در استراتژی‌های تکاملی عملگر جهش، عملگر اصلی است و نقش ابداع کننده در جمعیت را دارا می‌باشد. این تغییرات در پارامترهای استراتژی توسط عملگر جهش را «خودسازگاری» نامیده‌است.

استراتژی تکاملی بر پایه جمعیت‌های تک نقطه‌ای

اولین استراتژی تکاملی، بر پایه‌ی جمعیت‌های تک‌عضوی بنا شده است. در آن تنها از یک عملگر ژنتیکی در فرایند تکامل استفاده شده بود و آن جهش می‌باشد. اما نکته جالب توجه آن بود که یک عضو به صورت یک بردار با دو مولفه نشان داده می‌شود، یعنی $V = (x, \delta)$.

اولین مؤلفه (x) نمایانگر یک نقطه در فضای جستجو است و دومین مؤلفه، انحراف معیاری برداری است: با جایگزینی x توسط رابطه‌ی $x^{t+1} = x^t + N(0, \delta)$ جهش به وجود می‌آید، که $N(0, \delta)$ یک بردار از اعداد تصادفی و مستقل نرمال با مقدار متوسط صفر و انحراف معیار δ می‌باشد. نوزاد به وجود آمده (عضو جهش‌یافته)، به عنوان یک عضو جدید در جمعیت پذیرفته می‌شود و اگر برازندگی بهتری داشته باشد و همه‌ی قیدها را در صورت وجود ارضاء کند، جایگزین والدین خود می‌گردد. برای مثال، اگر F ، یک تابع هدف و بدون قید باشد و بخواهد ماکزیمم گردد، یک نوزاد (x^{t+1}, δ) جایگزین والدین خود (x^t, δ) می‌گردد، اگر $F(X^{t+1}) > F(X^t)$ باشد. در غیر این صورت، نوزاد حذف می‌گردد و جمعیت بدون تغییر باقی می‌ماند.

علی‌رغم این واقعیت که جمعیت تنها شامل یک عضو می‌باشد که جهش در مورد آن اعمال می‌شود، استراتژی تکاملی با عنوان «استراتژی تکاملی دو عضوی» نامیده می‌شود. دلیل این مسئله این است که نوزاد با والدین خود رقابت می‌کند.

استراتژی تکاملی چندعضوی

استراتژی تکاملی چندعضوی با استراتژی دو عضوی از لحاظ تعداد جمعیت تفـاوت دارد (تعـداد جمعیت >۱). مشخصات دیگر استراتژی تکاملی چندعضوی عبارتند از:

همه‌ی اعضای جمعیت، دارای احتمال یکسانی هستند. امکـان معرفـی یـک عملگـر ترکیبـی بـا استفاده از دو نفر از والدین که به طور تصادفی انتخاب می‌شوند وجوددارد.

برای تولید یک نوزاد، سیستم، مراحل مختلفی را طی می‌کند که بدین صورت بیان می‌گردد:

ابتدا دو عضو انتخاب شده و عملگر ترکیب بر روی آنها به کار گرفته شده، نوزاد جدید بـه دسـت می‌آید سپس عملگر جهش را در مورد نوزاد به کار می‌بریم.

استراتژی‌های تکاملی در مسائل عددی به خوبی کار می‌کنند چرا که آنها به مسـائل بهینه‌سـازی توابع، اختصاص دارند. آنها نمونه‌هائی از برنامه‌های تکـاملی هسـتند کـه از سـاختارهای اطلاعـاتی مناسب و عملگرهای ژنتیکی برای مسائل بهره می‌گیرند.

فصل دوم

معرفی الگوریتم‌های تکاملی

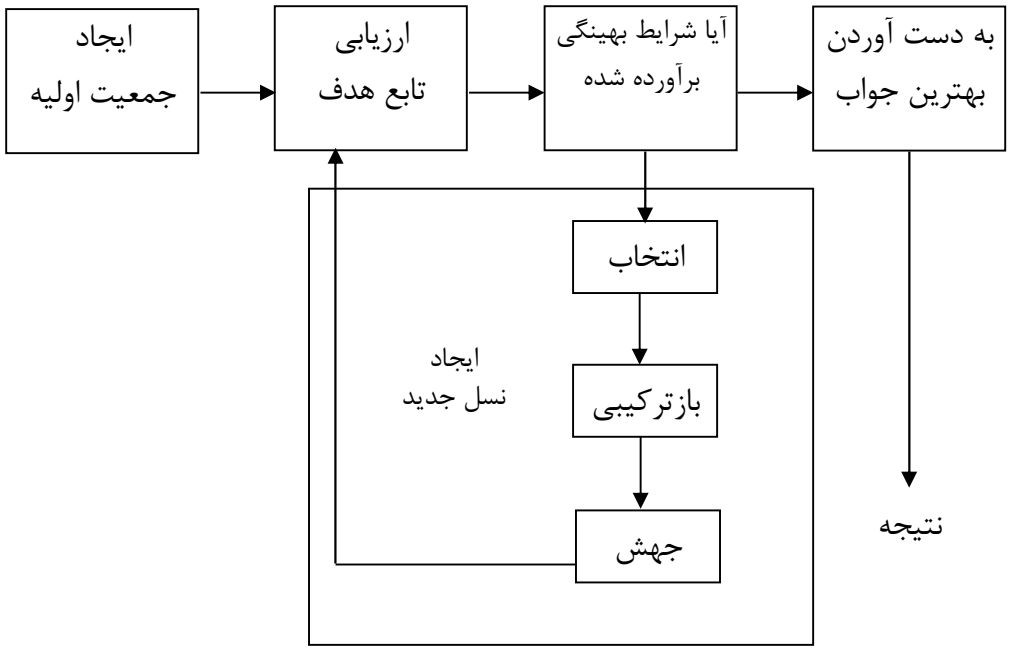
۱-۲- مقدمه

الگوریتم‌های تکاملی ، روشهای جستجوی تصادفی هستند که از روند تکامل در طبیعت تقلید و الگوبرداری شده است. این الگوریتم‌ها بیشتر برای مسائل بهینه‌سازی کاربرد دارد. و دارای این قابلیت می‌باشند که جوابی بهتر برای مسئله تولید می‌کند. روش کلی الگوریتم‌های تکاملی بدین صورت است که ابتدا برای مسئله به طور تصادفی جواب‌های اولیه که در شرایط مسئله صدق می‌کند در نظر می‌گیرد و سپس با توجه به عملگرهای موجودی الگوریتم‌های تکاملی جواب‌ها را به سوی جواب بهتر سوق می‌دهیم.

در هر نسل یک مجموعه جدید از جواب‌ها به وسیله روندهای انتخاب افراد مطابق سطح برازندگی‌شان در دامنه مسئله، و جفت‌گیری کردن آنها با یکدیگر با استفاده از عملگرهائی که از ژنتیک طبیعی و روند تکامل طبیعی، الگوبرداری شده است، تولید می‌شود. این روندها به تکامل جمعیت‌های افراد با برازندگی بهتر از افراد دیگر در محیطشان منتهی می‌شود .

عملگرهای مدل طبیعی الگوریتم‌های تکاملی عبارت است از: انتخاب، باز ترکیبی و جهش شکل زیر ساختار یک الگوریتم تکاملی ساده را نشان می‌دهد. الگوریتم‌های تکاملی روی یک جمعیت از افراد (راه حل) به جای یک فرد و یک راه حل اعمال می‌شود.

شکل صفحه بعد روند کلی الگوریتم‌های تکاملی را نشان داده است. همانطور که در شکل مشخص است ابتدا یک جمعیت اولیه که شامل جوابهایی برای مسئله می‌شود ایجاد می‌کنیم لازم به ذکر است که این تعداد اعضای این جمعیت اولی در طول مراحل الگوریتم‌های تکاملی باد ثابت باشد. بعد از اینکه جامعه اولیه جوابها ارزیابی می‌شود و سپس معیارهای بهینگی برسی می‌شوند. اگر معیارهای بهینگی براورد شوند که الگوریتم متوقف می‌شود و بهترین جواب بدست می‌آید در غیر این صورت عملکرد مختلف بر روی جواب‌های داخل جمعیت اعمال می‌شود. (روند کلی عملگرها در فصل‌های آتی به طور کامل مورد بررسی قرار می‌گیرد). سپس جمعیت جدید با همان تعداد عضو ایجاد می‌شود. دوباره این جوابها ارزیابی می‌شودو مانند حالت قبل شرایط بهبنگی بررسی می‌شود. این مراحل تا برآورد شدن شرایط بهینگی ادامه پیدا می‌کند. شکل صفحه بعد تمام این روند را نشان می‌دهد.



در شروع روند کلی الگوریتم‌های تکاملی ، جمعیت اولیه به طور تصادفی ایجاد می‌شوند و سپس تابع هدف را برای هر یک از افراد جامعه محاسبه وارزیابی می‌شودوبه این طریق است که نسل اول تولید می‌شود.حال اگر معیارهای بهینگی برای نسل کنونی برآورد شود، نتیجه می‌شود این نسل بهینه است و در غیر این صورت یک نسل جدید تولید می‌شودبدین صورت که افراد طبق برازندگی‌شان انتخاب می‌شوند وسپس باهم ترکیب می‌شوند(والدین با هم ترکیب می‌شوند و فرزند تولید می‌کنند) سپس فرزندان تحت عمل جهش قرار می‌گیرند. فرزندان در جامعه به جای والدین جایگزین می‌شوند و نسل جدید به وجود می‌آید و دوباره مثل قبل برازندگی افراد محاسبه وارزیابی می‌شود. این چرخه تا برآوردشدن معیارهای بهینگی ادامه پیدا می‌کند.

از بحث بالا اختلاف الگوریتم‌های تکاملی با دیگر روش‌های بهینه سازی و جستجوهای قدیمی مشخص می‌شود.

مهمترین اختلافات عبارتند از:

- الگوریتم‌های تکاملی ، جستجو روی یک جمعیت از نقاط به طور موازی انجام می‌دهد و نه تک نقطه‌ای
- الگوریتم‌ها ت تکاملی نیاز به اطلاعات زیاد و معین ندارد تنها نیاز به تابع هدف دارد.
- الگوریتم تکاملی ، برای کاربران بسیار ساده وقابل فهم می‌باشند.
- الگوریتم تکاملی ، تعداد زیادی جواب برای یک مسئله بدست می‌دهد و از بین این جوابها، جواب نهایی که شرایط بهینگی را برآورده، مورد استفاده قرار می‌گیرد.

در ادامه این فصل، لیست بعضی از روش‌ها وعملگرها الگوریتم‌های تکاملی آورده شده است و توضیحات بیشتر در فصل‌های آتی آورده شده است.

انتخاب

عملگر انتخاب تعیین می‌کند که کدام یک از افراد برای جفت‌گیری (بازترکیبی) انتخاب شود و این عمل جفت‌گیری، چه تعداد فرزند بوجود می‌آید.

والدین طبق برازندگی‌شان به وسیله یکی از روش‌های زیر انتخاب می‌شود:

- انتخاب بر اساس رتبه‌بندی
- انتخاب چرخ‌گردان
- انتخاب بر اساس نمونه تصادفی
- انتخاب محلی
- انتخاب هرس
- انتخاب مسابقه‌ای

برای کسب اطلاعات بیشتر به فصل ۴ مراجعه کنید.

باز ترکیبی

باز ترکیبی، افراد جدیدی را از ترکیب اطلاعات افراد انتخاب شده (والدین) به وجود می‌آورد.این عمل توسط ترکیب مقادیر متغیرهای والدین انجام می شود و فرزند به وجود می‌آید. انواع روش‌های باز ترکیبی عبارت است از:

- باز ترکیبی مجزا
- باز ترکیبی میانی
- باز ترکیبی خطی
- باز ترکیبی خطی توسعه یافته
- عملگرcrossover تک نقطه‌ای/ دونقطه‌ای/ چند نقطه‌ای
- عملگرcrossover یک شکل
- عملگرcrossover بر زننده
- عملگرcrossover با کاهش جایگزینی

باز ترکیبی‌های میانی و خطی و خطی توسعه یافته روی مقادیر حقیقی عمل می‌کند وباز ترکیبی‌های که روی متغیرهایی با مقادیر باینری عمل می‌کنند به عملگر crossover مشهور است

برای کسب اطلاعات بیشتر به فصل ۵ مراجعه کنید.

جهش

بعد از باز ترکیبی هر فرزند باید جهش را تحمل کند و این مرحلـه را طـی کنـد. عملگـر جهـش باعث تغییراتی در هر فرزند می‌شودکه این تغییرات خیلی کوچک می‌باشند

دو نوع عملگر جهش وجود دارد:

-جهش مقادیر حقیقی

- جهش مقادیر باینری

برای کسب اطلاعات بیشتر به فصل ۶ مراجعه کنید.

دوباره گنجاندن ، دوباره جا دادن

وقتی که فرزندی توسط عملگرهای انتخاب، باز ترکیبی و جهش از والدین‌شان به وجـود می‌آیـد. برازندگی افراد و فرزندان بسیار مهم و تعیین کننده است و باعث حذف شدن تعـدادی از فرزنـدان می‌شود. اگر تعداد فرزندان تولید شده کمتر از تعداد افراد جامعه عمومی باشد در این صورت بـرای رسیدن به آن تعداد کافی است که افراد جمعیت قبلـی را داخـل جامعـه فعلـی دوبـاره بگنجـانیم. دوباره گنجاندن روش‌های مختلف دارد که عبارتند از:

-دوباره گنجاندن عمومی وکلی

-دوباره گنجاندن محلی

برای کسب اطلاعات بیشتر به فصل ۷ مراجعه کنید.

فصـــل سوم

انواع عملگرهای انتخاب

۳-۱-عملگر انتخاب بر اساس رتبه‌بندی

۳-۲- عملگر انتخاب چرخ گردان

۳-۳- عملگر انتخاب بر اساس نمونه تصادفی

۳-۴- عملگر انتخاب محلی

۳-۵- عملگر انتخاب هرسی

۳-۶- عملگر انتخاب مسابقه‌ای

۳-۷- مقایسه الگوهای مختلف عملگرهای انتخاب

عملگرهای انتخاب

در قسمت انتخاب، والدین بر اساس نسبت سازگاریشان انتخاب می‌شوند. به هـر فـرد در قسـمت انتخاب یک احتمال تکثیر که وابسته به تابع احتمال است، اختصاص می‌یابد و مقدار تابع هدف همه افراد محاسبه می‌شود و در قسمت انتخاب قرار می‌گیرد.

قبل از آنکه وارد بحث شویم اصطلاحات زیر را تعریف می‌کنیم. این اصطلاحات برای مقایسه روش‌هـای مختلف انتخاب کاربرد فراوان دارند.

۱- فشار انتخابی یا تمرکز انتخابی:

احتمال اینکه بهترین افراد در مقایسه با میانگین احتمال انتخاب همه افراد، انتخاب شوند.

۲- اریب:

اختلاف بین نسبت سازگاری نرمال شده افراد ودرجه احتمال تکثیر آنها.

۳- گسترش، انتشار:

تعداد فرزندی که از والدین به وجود می‌آیند.

۴- درصد فقدان وعدم گوناگونی:

تعداد افرادی از جامعه که در طول انتخابمان، انتخاب نمی شود.

۵- شدت انتخاب :

میانگین برازندگی افراد جامعه که بعد از کاربرد متد انتخاب در توزیع گوسین نرمال شده انتظار می‌رود.

۶- پراکندگی انتخاب (واریانس انتخاب):

میزان پراکندگی ، برازندگی افراد جامعه که بعد از کاربرد متد انتخاب در توزیـع گوسـین نرمـال شـده انتظار می‌رود.

۴-۱- عملگرهای انتخاب بر اساس رتبه بندی

در انتخاب بر اساس رتبه بندی، افراد جامعه مطابق تابع هدف دسته‌بندی می‌شوند و برازندگی هر فرد تنها وابسته به موقعیت‌شان در بین افراد جامعه دارد. انتخاب بر اساس رتبه‌بندی باعث همگرایی خیلی سریع می‌شودو عمل جیتجو را کاهش می‌دهد. در انتخاب بر اساس رتبه بندی ، میزان تکثیر محدود است و هیچ کدام از والدین فرزند زیادی تولید نمی‌کنند. رتبه دهی به افراد بر اساس میزان سازگاری‌شان ، یک راه ساده و موثر برای کنترل تمرکز انتخاب ارائه می‌دهد.

روش انتخاب

فرض کنید که تعداد افراد جامعه برابر N باشد ورتبه یک فرد در این جامعه برابر P باشد.

(عدد P از ۱ شروع می‌شود و به N ختم می شودفرد با کمترین برازندگی دارای P=1 و مناسبترین فرد دارای رتبه P=N می‌باشند) و sp معیار انتخابی است.

رتبه گذاری برای هر فرد مانند روش‌های زیر محاسبه می‌شود:

الف: رتبه گذاری خطی:

Fitnes(P)=2-sp+2*(sp-1)*(p-1)/(n-1)

Sp متغیری است که به طور تصادفی در فاصله [۲،۱] انتخاب می‌کنیم $sp \in [1,2]$

ب)رتبه‌گذاری غیر خطی: استفاده از رتبه‌گزاری غیر خطی اجازه انتخاب sp بزرگتر از روش رتبه‌گزاری خطی را می‌دهد.

$$Fitness(p) = N * x^{p-1} / \sum_{i=1}^{N} x^{\frac{i-1}{}}$$

که در آن x ریشه چند جمله‌ای زیر است.

$$(sp-1) * x^{n-1} + sp * x^{N-2} + sp * x + sp = o$$

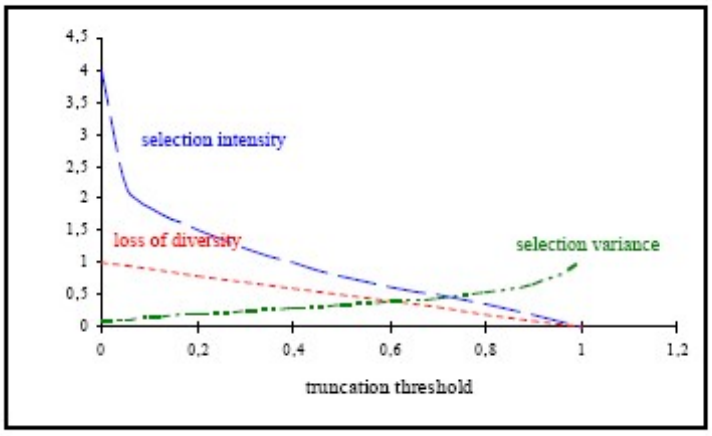
مقدار sp مربوط به رتبه‌گذاری غیر خطی در فاصله [1,N-2] انتخاب می‌شود.

$$sp \in [1, N-2]$$

حال با توجه به معادله بالا داریم :

$$Finess(p) = N * x^{p-N}$$

شکل زیر رتبه‌گذاری خطی را از طریق نمودار با هم مقایسه می‌کند و با توجه به پارامترهای مختلف می‌توان این دو روش رتبه‌گذاری مختلف را بررسی کرد.



شکل ۱-۴ برازندگی رتبه‌گزاری خطی و غیر خطی

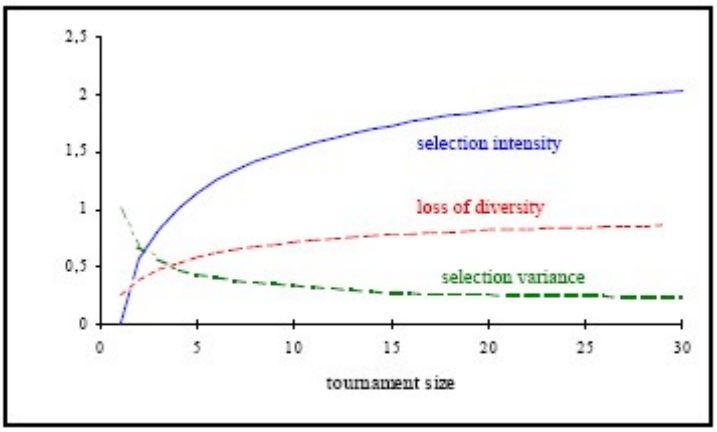
جدول زیر (جدول ۱-۴) شامل مقادیر مختلف sp با توجه به اینکه جمعیت ما شامل ۱۱ فرد و مسئله ما مینیمم سازی است. جدول صفحه بعد مقادیر رتبه‌دهی خطی و غیر خطی برای جمعیت ۱۱ نفری را بررسی می‌کند و بر اساس برازندگی هر فرد با توجه به فرمول‌های خطی و غیر خطی به هر فرد در جمعیت یک رتبه اختصاص پیدا می‌کند.

جدول۱-۴- مقایسه رتبه‌گزاری خطی و غیر خطی

		مقدار برازندگی هر یک از افراد			
رتبه‌گذاری غیرخطی			رتبه‌گزاری خطی		
۲/۰	۳/۰	رتبه فرد در جامعه	۱/۱	۲	مقدار تابع هدف
۲/۰۰	۳/۰۰	۱۱	۱/۱۰	۲/۰	۱
۱/۶۹	۲/۲۱	۱۰	۱/۰۸	۱/۸	۳
۱/۴۳	۱/۶۲	۹	۱/۰۶	۱/۶	۴
۱/۲۱	۱/۹۹	۸	۱/۰۴	۱/۴	۷
۱/۰۳	۰/۸۸	۷	۱/۰۲	۱/۲	۸
۰/۸۷	۰/۶۵	۶	۱/۰۰	۱/۰	۹
۰/۷۴	۰/۴۸	۵	۰/۹۸	۰/۸	۱۰
۰/۶۲	۰/۳۵	۴	۰/۹۶	۰/۶	۱۵
۰/۵۳	۰/۲۶	۳	۰/۹۴	۰/۴	۲۰
۰/۴۵	۰/۱۹	۲	۰/۹۲	۰/۲	۳۰
۰/۳۸	۰/۱۴	۱	۰/۹۰	۰/۰	۹۵

در جدول صفحه قبل در ستون اول مقدار تابع هدف هر فرد نوشته شده است که با توجه به اینکه فرد با برازندگی بیشتر دارای رتبه بهتر است لذا تابع برازندگی ۹۵ دارای رتبه ۱ و به همین ترتیب رتبه تابع برازندگی ۱ برابر ۱۱ می‌باشد که با توجه به فرمولهای رتبه خطی و غیر خطی و مقادیر مختلف SP مقادیر برازندگی برای افراد جامعه بدست می‌آید.

این روش انتخاب دارای این قابلیت می‌باشد که توسط انتخابهای دیگر این روش کاملتر می‌شود که دارای قابلیت‌های فراوانی می‌باشد.شکل زیر مشخصات عملگر انتخاب رتبه‌گزاری خطی را نشان می‌دهد.



شکل۲-۴ مشخصات عملگر انتخاب رتبه‌گزاری خطی

شدت انتخاب

$$Sel\ln t_{LinRank}(sp) = (sp-1)/\sqrt{\pi} \qquad (۴-۴)$$

فقدان یا عدم گوناگونی

$$LossDiv_{linRank}(sp) = (sp-1)/4 \qquad (۵-۴)$$

میزان پراکندگی انتخاب

$$Sel\,\mathrm{var}_{linRank}(sp) = 1 - \frac{(sp-1)^2}{\pi} = 1 - (sel\ln t_{linRank}(sp))^2 \qquad (۶-۴)$$

جدول زیر سه پارامتر ، شدت انتخاب و فقدان یا عدم گوناگونی و میزان پراکندگی انتخاب را با هم مقایسه کرده است .مقادیر مختلف این سه پارامتر برای ۱۱ فراد جمعیت قبلی بدست آمده است.

جدول ۲-۴ مقایسه پارامترهای انتخاب

Sp	Selection intensity	Loss of diversity	Selection variance
۲	۰/۵۶۴	۰/۲۵۰	۰/۶۸۲
۱/۹	۰/۵۰۸	۰/۲۲۵	۰/۷۴۲
۱/۸	۰/۴۵۱	۰/۲۰۰	۰/۷۹۷
۱/۷	۰/۳۹۵	۰/۱۷۵	۰/۸۴۴
۱/۶	۰/۳۳۹	۰/۱۵۰	۰/۸۸۵
۱/۵	۰/۲۸۲	۰/۱۲۵	۰/۹۲۰
۱/۴	۰/۲۲۶	۰/۱۰۰	۰/۹۴۱
۱/۳	۰/۱۶۹	۰/۰۷۵	۰/۹۷۱
۱/۲	۰/۱۱۳	۰/۰۵۰	۰/۹۸۷
۱/۱	۰/۰۵۶	۰/۰۲۵	۰/۹۹۷
۱	۰	۰	۱

۳-۲- عملگرهای انتخاب چرخ گردان

ساده‌ترین روش انتخاب ، روش انتخاب چرخ گردان است [6]و نیز این روش نمونه گیری تصادفی جایگزینی نامیده می‌شود. این یک الگوریتم تصادفی است و شامل تکنیک‌های زیر است.

۱- افراد به طور پیوسته روی یک پاره خط تصویر می شوند و فاصله‌ای که هر فرد اشغال می‌کند معادل اندازه برازندگی آن فرد است.

۲- عدد تصادفی انتخاب می‌کنیم و فردی که معادل عدد تصادفی است انتخاب می‌شود این روند را تا زمانی که تعداد کل جمعیت تکمیل نشود و تعداد افراد انتخاب شده به تعداد افراد جامعه اصلی نرسد، انجام می شود. این مشابه این تکنیک است که یک چرخ گردان را نسبت برازندگی افراد برش دهیم .

جدول زیر ، احتمال انتخاب و برازندگی هر فرد ونیز فشار انتخابی هر فرد را نشان می‌دهد.

فرد اول مناسبترین است لذا بیشترین فاصله را اشغال می‌کند وفرد دهم بعد از فرد یازدهم دارای کمترین برازندگی است و لذا کوچکترین فاصله را روی خط اشغال می‌کند. فرد یازدهم دارای مقدار برازندگی صفر می‌باشد در روند انتخاب ندارد.

جدول ۳-۴ انتخاب چرخ گردان

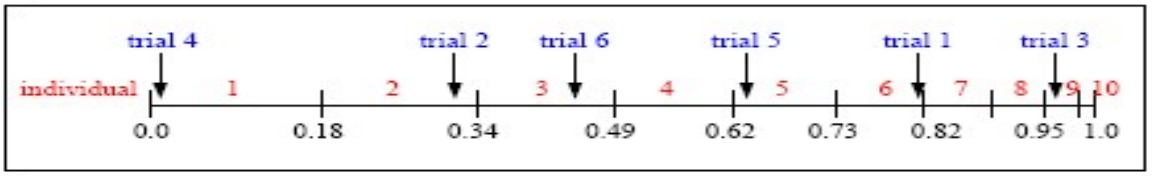
شماره افراد	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱
مقدار برازندگی	۲/۰	۱/۸	۱/۶	۱/۴	۱/۲	۱/۰	۰/۸	۰/۶	۰/۴	۰/۲	۰
احتمال انتخاب	۰/۱۸	۰/۱۶	۰/۱۵	۰/۱۳	۰/۱۱	۰/۰۹	۰/۰۷	۰/۰۶	۰/۰۳	۰/۰۲	۰

مقدار احتمال انتخاب افراد از تقسیم مقدار برازندگی هر فرد بـر مجمـوع کـل برازنـدگی افـراد بدسـت می‌آید. حال برای هر فرد احتمال تجمعی هر فرد برابر است با احتمال همان فرد بعلاوه احتمال های افراد قبل از آن و سپس این احتمال‌های تجمعی را روی یک پاره خط تصـویر مـی‌کنیم. رونـد انتخـاب بـدین صورت است که به همان تعداد که میخواهیم جواب مناسـب از جمعیـت ، جـواب انتخـاب کنـیم، عـدد تصادفی در فاصله صفر و یک حال انتخاب می‌کنیم. حال هر کدام از اعداد در فاصله‌ای روی پاره خط قرار گیرد فرد نظیر به‌آن انتخاب می‌شود. مانند مثال زیر:

شش عدد تصادفی در فاصله $[0,1]$ انتخاب می‌کنیم.

۰,۴۲و۰,۶۵و۰,۰۱و۰,۹۶و۰,۳۲و۰,۸۱

شکل زیر عمل و روند انتخاب افراد را نشان می‌دهد . افراد توسط رابطه $\frac{f_i}{\sum f_i}$ روی پـاره خـط تصـویر می‌شوند.



شکل۳-۴ انتخاب چرخ گردان

بعد از انتخاب جامعه شامل افراد زیر می‌باشد

شماره افراد: ۱و۲و۳و۵و۶و۹

در روش انتخاب چرخ گردان مقدار اریب برابر صفر بدست می‌آید.

در اغلب روشهای اولیه برای انتخاب چرخ گردان روش بدین صورت بوده است که برازندگی افراد جامعه روی محیط دایره (دایره به شعاع یک) تصویر می‌شود . لذا مانند شکل، قبل هر فردی که دارای برازندگی بیشتری باشد طول قطاع بیشتری را روی دایره اشغال می‌کند و لذا احتمـال انتخـاب شـدن آنهـا بیشـتر است. این روند از روی روند مکانیکی چرخ متحرکی که دارای یک پیکان در مرکز می‌باشـد الگـو بـرداری شده است. چرخ را می‌چرخانیم و بعد از مدتی که چرخ متوقف شود پیکان روبروی یک قطاع قرار گرفتـه است لذا فرد معادل قطاع انتخاب می‌شود.

ما از این روند الگـوبرداری کـرده‌ایم و افـراد را بـا توجـه بـه فرمـول $\frac{2\times\pi\times f_i}{\sum f_i}$ کـه در آن f_i میـزان برازندگی فرد i است، روی محیط دایره تصویر می‌شوند و با توجه به برازندگی هر فرد آن فـرد، کمـانی از دایره را اشغال می‌کند . حال توسط رایانه یک عدد به طور تصادفی در فاصـله $[o,2\pi]$ انتخـاب می‌شـود وسپس فرد معادل آن عدد انتخاب می‌شودو این روند را تا جایی که تمام افراد جامعه ، انتخاب شوند ادامه پیدا می شود.

به همین دلیل است که عنوان این عملگر انتخاب را چرخ گردان گذاشته شده است.

۳-۳-انتخاب بر اساس نمونه گیری کاملاً تصادفی

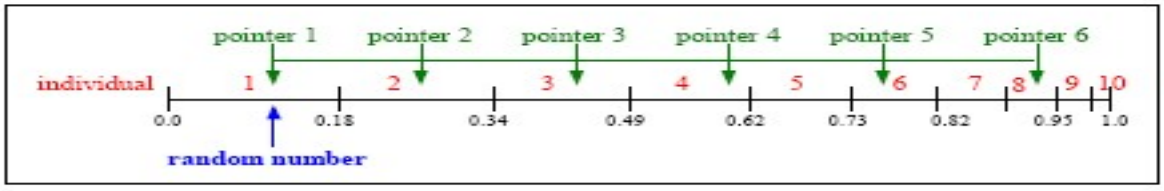
در این روش [5] انتخاب افراد روی قسمتی از خط تصویر می‌شوند. مانند قسمت چرخ گـردان بـا ایـن تفاوت که در این فاصله اشارگرها به طور مساوی می‌باشند.

- روش انتخاب

حال فرض کنید تعداد افراد که می‌خواهیم انتخاب کنیم برابر NP باشـد در ایـن صـورت فاصـله بـین اشارگرها برابر 1/NP و موقعیـت اولـین اشـارگر بـه وسـیله انتخـاب یـک عـدد تصـادفی کـه در فاصـله $[0,1/NP]$ تعیین می‌شود. به طور مثال برای انتخاب شش نفر ، فاصله اشاره‌گر از هم باید 1/6=0.167 باشد. و اولین فرد به طور کاملاً تصادفی در فاصله [0,0.167] انتخاب می‌شود.

روند بدین صورت است که اولین اشارهگر در فاصله [0,0.167] به طور کاملاً تصادفی انتخاب می‌شود و اشاره‌گر اول در این فاصله قرار می‌گیرد . مکان اشاره‌گر دوم با اضافه شدن مقدار ۰/۱۶۷ به اشاره‌گر اول بدست می‌آید.

و این روند را ادامه می‌دهیم تا به تعداد مورد نیاز فرد انتخاب شود. شکل زیر رونـد انتخـاب بـر اسـاس نمونه گیری کاملاً تصادفی را نشان می‌دهد.



شکل ۳-۴ روند انتخاب بر اساس نمونه گیری تصادفی

بعد از عمل انتخاب ، شماره افراد انتخاب شده عبارت است از:

۱و۲و۳و۴و۶و۸

در انتخاب بر اساس نمونه‌گیری کاملاً تصادفی، افرادی که انتخاب می‌شوند نسبت به افرادی که توسـط تکنیک انتخاب چرخ گردان انتخاب می‌شوند، مناسب‌تر هستند.

در انتخاب بر اساس نمونه‌گیری کاملاًتصادفی، مقدار اریب برابر صفر بدست می‌آید.

۳-۴- عملگرهای انتخاب محلی

در انتخاب محلی[7] برای هر فرد، یک محیط محدود شده (همسایگی محلی) در نظر می‌گیریم و عمل انتخاب روی این همسایگی محلی صورت می‌گیرد. (در روش دیگر انتخاب، عمل انتخاب روی کل جمعیت انتخاب می‌شود). وقتی از این روش انتخاب استفاده می شود ، کـه در جامعـه مـا غیـر متمرکـز و گسترده باشد. این همسایگی را می‌توان طوری انتخاب کرد که افـراد داخـل ایـن همسـایگی از پتانسـیل بالایی برای ترکیب شدن و جفت گیری داشته باشد.

در روش انتخاب محلی ابتدا جمعیت جوابها محدود می‌شود، و سـپس در ناحیـه جدیـد توسـط دیگـر عملگرها انتخاب، مانند انتخاب چرخ گردان یا انتخاب کاملاً تصادفی و یا دیگر روش‌هـای انتخـاب، افـراد جامعه انتخاب می‌شود.

روش انتخاب محلی قابلیت ترکیب شدن با دیگر روش‌های انتخابی را دارا می‌باشد.

ساختمان همسایگی‌ها می‌تواند به صورت زیر باشد.

۱- خطی

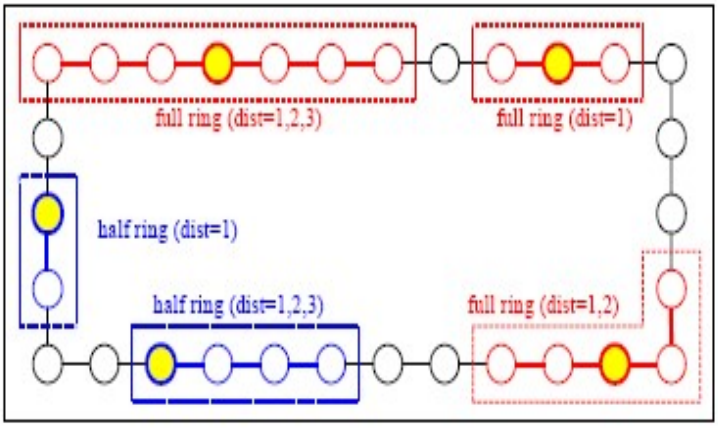
۱ـ۱ حلقه کامل، حلقه نیمه(شکل ۵-۳)

۲- دو بعدی

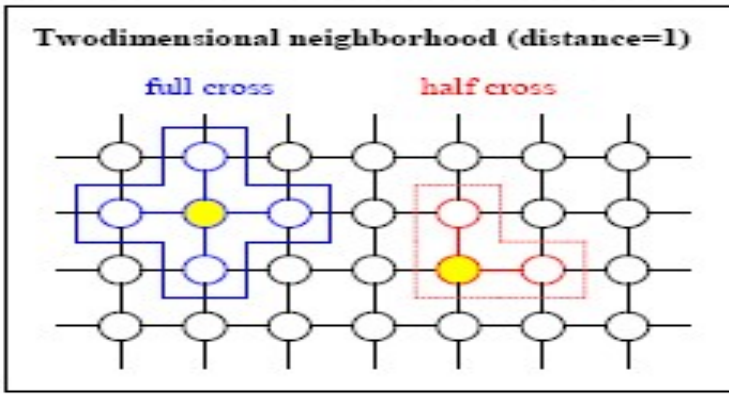
۲-۱- متقاطع کامل، مقاطع نیمه (شکل۶-۳)

۲-۲- ستاره کامل، ستاره نیمه(شکل ۷-۳)

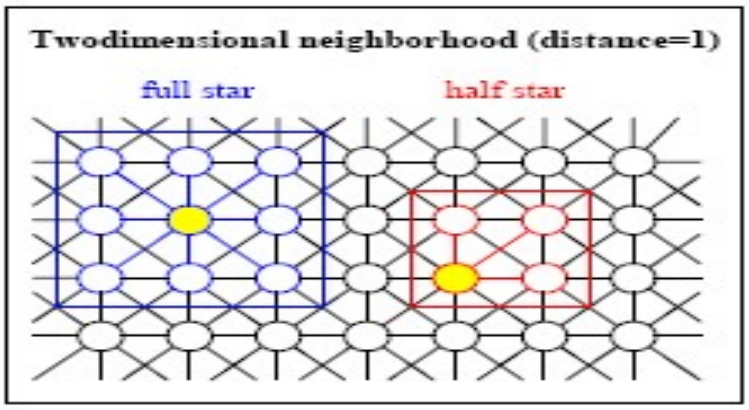
۳- سه بعدی وبا پیچیدگی زیاد با هر ترکیبی از ساختارهای بالا



شکل ۵-۳ حلقه کامل وحلقه نیمه



شکل ۶-۳ همسایگی دو بعدی، متداخل نیمه و متداخل کامل



شکل ۷-۳ همسایگی دو بعدی، ستاره کامل و ستاره نیمه

روش عملگرهای انتخاب

اولین گام، انتخاب فردی از جامعه که قابلیت بالایی برای جفت گیری رد، است. که این انتخاب به طور تصادفی و با روش‌های انتخاب به طور تصادفی و با روش‌های انتخاب که قبلاً ذکر شده برای مثال انتخاب بر اساس نمونه گیری تصادفی یا انتخاب چرخ گردان و یا انتخاب کوتاه سازی که بعداً توضیح داده می‌شود.

حال یک همسایگی محلی برای هر فرد انتخاب شده، تعریف می‌کنیم. سپس داخل این همسایگی، افرادی که می‌توانند با هم جفت شوند، تعریف می‌کنیم. سپس داخل این همسایگی، افرادی که می‌توانند با هم جفت شوند (بر اساس معیار بهترین یا مناسبترین و به طور یکنواخت و تصادفی).

در جدول صفحه بعد مثال‌هایی از اندازه همسایگی و نیز فاصله آنها برای ساختارهای مختلف داده شده است.

ساختار	فاصله	
Structure of selection	۱	۲
Full ring	۲	۴
Half ring	۱	۲
Full cross	۴	۸(۱۲)
Half cross	۲	۴(۵)
Full star	۸	۲۴
Half star	۳	۸

اندازه همسایگی ، سرعت تکثیر اطلاعات بین افراد جامعه را تعیین می‌کند. بنابراین بین تکثیر سریع و تنوع زیاد در جامعه تصمیم گیری می‌شودو اغلب تمایل به تنوع زیاد می‌شود بنابراین از همگرایی سریع به یک مینیمم محلی جلوگیری می‌شود.

انتخاب محلی در یک همسایگی کوچک قابل اجرارتر از انتخاب محلی در یک همسایگی بزرگتر است. با این وجود همبندی کل جامعه باید برقرار بماند. همسایگی دو بعدی با ساختار نیمه ستاره با شعاع همسایگی برابر یک برای انتخاب محلی توصیه می‌شود.

به هر جهت اگر تعداد افراد جامعه ما زیاد باشد در این صورت یا شعاع همسایگی را زیاد در نظر می‌گیریم و یا از همسایگی دو بعدی دیگر باید استفاده کرد.

۳-۵- عملگرهای انتخاب **هرسی (کوتاه سازی)**

مقایسه مدل‌های انتخابی نشان می‌دهد که روش‌های انتخاب طبیعی الگو برداری شده است اما مدل انتخابی هرسی روش انتخابی مصنوعی است.

در انتخاب کوتاه‌سازی، افراد بر اساس برازندگی‌شان ذخیره می‌شوند. و تنها بهترین افراد به عنوان والدین انتخاب می‌شوند. والدین انتخاب شده به این روش، فرزندانی به صورت تصادفی و یکنواخت تولید می‌کنند. این روش انتخاب مانند انتخاب محلی دارای قابلیت ترکیب شدن با دیگر روش‌های انتخاب (انتخاب چرخ گردان یا انتخاب کاملاً تصادفی) را دارا می‌باشد.

پارامتر انتخاب هرسی، trunce (پارامتر کوتاه سازی مرزی) می‌باشد. (اغلب مقدار پارامتر برابر $trunce = 1/sp$ که در آن $sp \in [1, Np-2]$ می‌باشد که در آن NP برابر تعداد افراد جامعه است).

پارامتر trunce نسبت برازندگی جمعیت را به برازندگی والدین های انتخاب شده را تعیین می‌کند و مقدار بدست آمده بین ده درصد و پنجاه درصد می‌باشد. افراد زیر مرز پارامتر کوتاه سازی صاحب فرزند نمی‌شوند. جمله شدت انتخاب اغلب در انتخاب هرسی استفاده می‌شود.

جدول زیر رابطه بین پارامتر کوتاه سازی و شدت انتخاب را نشان می‌دهد.

جدول ۴-۳ رابطه بین شدت انتخاب و مرز کوتاه سازی.

Truncation threshold	۱٪	۱۰٪	۲۰٪	۴۰٪	۵۰٪	۸۰٪
Selection intensity	۲/۶۶	۱/۷۶	۱/۲	۰/۹۷	۰/۸	۰/۳۴

بررسی پارامترهای مختلف انتخاب هرسی

تراکم انتخاب

$$sel\ln t_{Truncatin}(Trunc) = \frac{1}{Trunc}.\frac{1}{\sqrt{2\pi}}.e^{\frac{-fc^2}{2}}$$

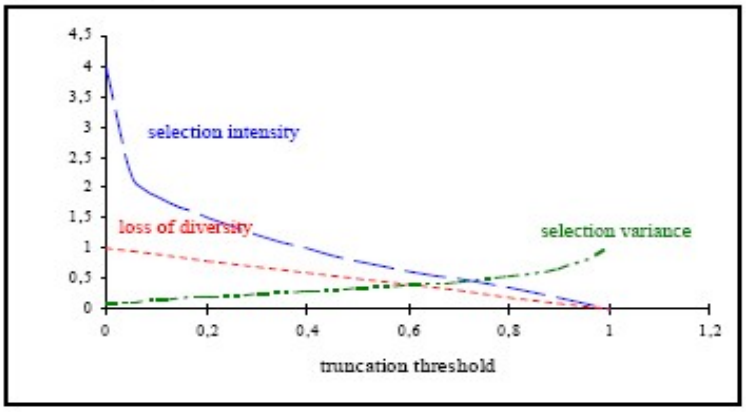
فقدان یا عدم گوناگونی

$$lossDiv_{Truncation}(Trunc) = 1 - Trunc$$

میزان پراکندگی انتخاب

$$selver_{Truction}(Trunce) = 1 - sel\ln t_{Truction}(Trunc).(sel\ln t_{Truncation}(Trunc) - f_c)$$

شکل زیر ویژگیهای انتخاب هرسی را نشان می‌دهد



شکل ۷-۳ ویژگی‌های انتخاب هرسی

روش عملگرهای انتخاب هرسی

در این قسمت با ارائه یک مثال قصد داریم[3] روش انتخاب هرسی را بررسی کنیم . روند بدین صورت است که ما یک مرز در نظر می‌گیریم. افرادی که بالای مرز باشند را نگه می‌داریم و افرادی که زیر مرز می باشند را حذف می‌کنیم و بعد با استفاده از روشهای انتخاب تعداد افراد مورد نیاز را انتخاب می‌کنیم.

مثال: فرض کنید مقادیر برازندگی افراد جامعه به صورت زیر باشد

Fitnv=[2,0.7,0.7,3.1,4.12.01.60.90.1]

حال مقدار پارامتر Tranc را محاسبه می‌کنیم و با توجه به اینکه sp=2 انتخاب شده است

Trunce = 1/sp =1/2=0/5

حال برای تک تک افراد جامعه مقدار $\frac{FitnV}{\max(FitnV)}$ را حساب می‌کنیم.

$$\frac{FitnV}{\max(Fitnv)} = [0/05,0/45,0,8/1,0/2,0/65,0/85,0/35,0/1]$$

حال افرادی که مقدار $\frac{FitnV}{\max(FitnV)}$ آن کمتر از ۰/۵ است حذف می‌کنیم و از بین بقیه افراد تعدادی را انتخاب می‌کنیم. به طور مثال شش نفر را انتخاب کرده که عبارتند از:

$$Nem\ chrom\ = [3,4,7,4,6,7]$$

انتخاب هرسی، انتخابی است که قابلیت ترکیب شدن با دیگر روشهای انتخابی را دارا می‌باشد.

۳-۶- عملگرهای تکاملی به روش مسابقه‌ای (قرعه کشی)

این روش نیز مانند روش انتخاب هرسی و بر خلاف روشهای قبلی که از روشهای انتخاب طبیعی الگو برداری شده است ، یک روش انتخاب مصنوعی است.

در عمل انتخاب به روش مسابقه‌ای[3] هر فرد جامعه دارای یک عدد (شماره) می‌باشند. این روش انتخاب مانند روشهای قرعه کشی می‌باشد و بر اساس قرعه کشی افراد انتخاب می‌شود. البته می‌توان این روش انتخاب را طوری طراحی کرد که احتمال انتخاب افراد با برازندگی بالا بیشتر از افراد دیگر باشد. در این روش انتخاب یک عدد (ToUr) را به طور تصادفی انتخاب می‌کنیم و سپس آن فرد به عنوان والدین انتخاب می‌شود. این روند انتخاب والدین به طور مکرر تکرار می‌شود. این روند انتخاب والدین، به طور تصادفی باعث تولید فرزند می‌شود. پارامتر انتخاب به روش مسابقه‌ای ،(tour)، اندازه مسابقه می‌باشد. مقدار (tour) از ۱ تا NP (تعداد افراد جامعه) قابل تغییر است.

بررسی پارامترهای مختلف انتخاب به روش مسابقه‌ای

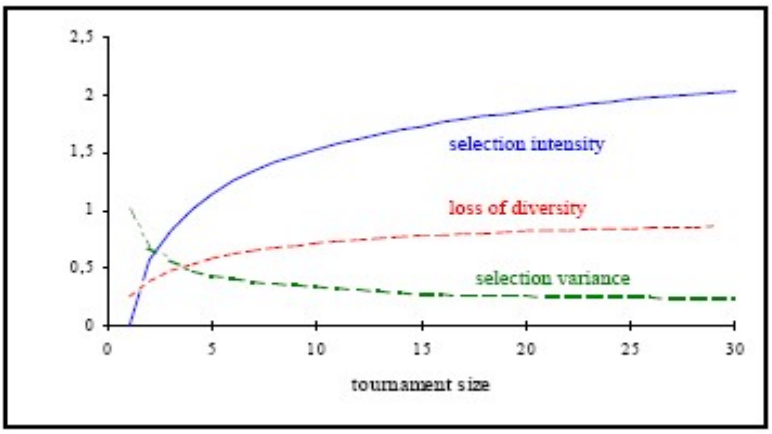
شدت انتخاب

$$sel\ln t_{Turnier}(Tour)\sqrt{2.\left(In(Tour)-In(\sqrt{4.14.In(Tour)}\right)}$$

درصد عدم گوناگونی

$$lossiv_{urniter}(tour)\frac{0.918}{In(1.186+1.328.Tour)}$$

شکل ۸-۳ رابطه بین اندازه مسابقه و پارامتر تمرکز انتخابی را نشان می‌دهد.



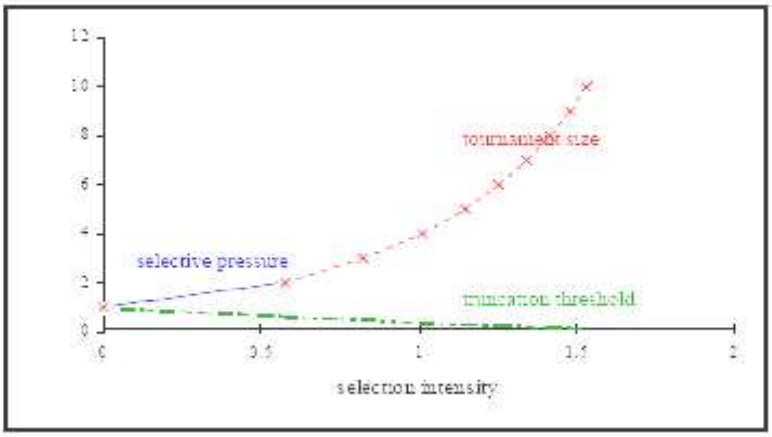
شکل۸-۳ رابطه بین اندازه مسابقه و پارامتر تمرکز انتخابی

مانند روشهای دیگر انتخابی ، انتخاب مسابقه‌ای دارای این ویژگی است که می توان بـا دیگـر روشـهای انتخابی ترکیب شود و نتایج بهتری برای مسئله ایجاد کند. مثلاً می‌توان افـرادی کـه در مرحلـه انتخـاب مسابقه‌ای انتخاب می‌شوند را دوباره توسط روشهای دیگر انتخاب افراد بهتری برای مسئله انتخاب کرد.

۴-۷-مقایسه الگوهای مختلف عملگرهای انتخاب

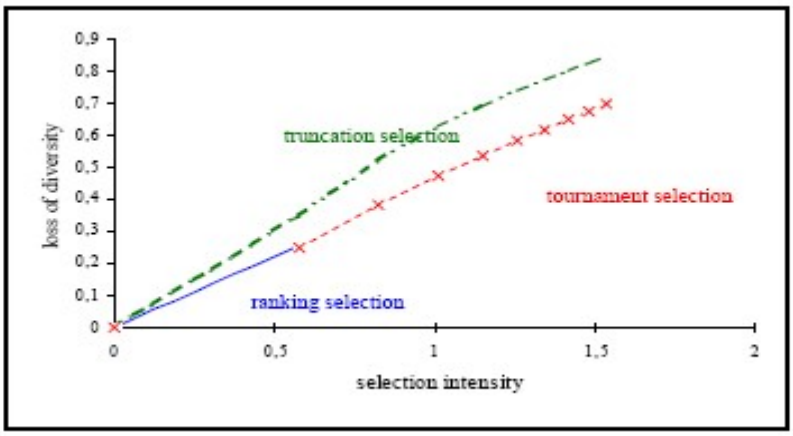
با توجه به مطالب که در قسمتهای قبل نشان داده شد. چنین نتیجه می‌شود که رفتار روشـهای مختلف انتخابی شبیه شدت انتخاب می باشد.

شکل زیر (شکل ۹-۳) رابطه بین شدت انتخاب وویژگی‌های پارامترهای روشهای مختلف انتخابی (فشار انتخابی، مرز کوتاه سازیو اندازه مسابقه) را نشان می دهد. شکل زیر شرح می دهد که در انتخاب به روش مسابقه ای تنها از مقادیری جدا، اجازه داده می‌شود و در انتخاب به روش رتبه‌گذاری خطی تنها مجاز بـه استفاده کوچکترین مقدار برای شدت انتخاب می‌باشیم.



شکل ۹-۳- وابستگی بین پارامترانتخاب وشدت انتخاب

به هر جهت رفتار روشهای مختلف باهم متفاوت است. لذا روشهای انتخاب با تعداد افرادی که در طـول انتخاب حذف می‌شوند، با شدت انتخاب مقایسه می شوند(شکل ۱۰-۳).



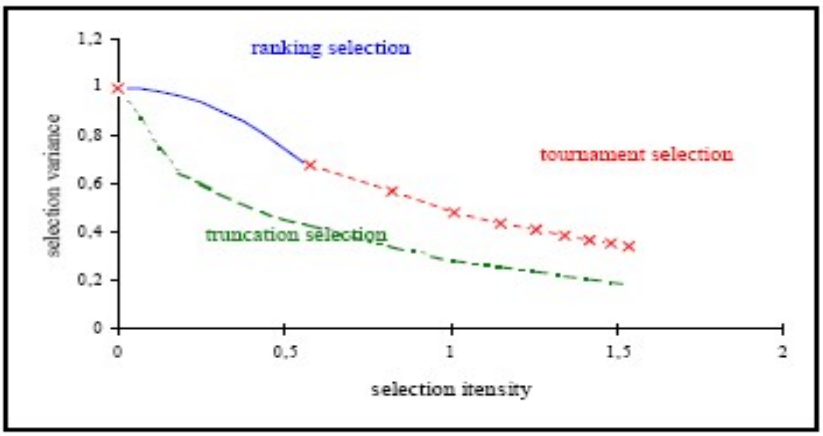
شکل ۱۰-۳ وابستگی تعداد افراد انتخاب نشده به شدت انتخاب

انتخاب به روش کوتاه‌سازی نسبت به روشهای دیگر انتخاب، دارای تعداد زیادی افراد است که انتخـاب نشده‌اند.

در مقایسه با دیگر روشـهای انتخـابی(روش انتخـاب رتبه‌گـذاری ،روش انتخـاب مسـابقه‌ای) . در روش انتخاب هرسی درجه برازندگی افرادی که کمتر از بقیه هستند حذف می‌شوندو به جای آنها فرزندانی کـه دارای درجه برازندگی بیشتر است، جایگزین می شوند زیرا افرادی که درجه برازندگی آنها زیر حد انتخاب باشند انتخاب نمی شوند.

روش انتخاب رتبه‌گزاری و روش انتخاب مسابقه‌ای دارای رفتار شبیه به روش انتخاب هرسـی هسـتند . به هر جهت انتخاب به روش رتبه‌گزاری در یک سطحی عمل می‌کند کـه انتخـاب بـه روش کوتاه‌سـازی نمی‌تواند عمل کند زیرا دارای مشخصه مجزا از انتخاب به روش مسابقه‌ای است.

در شکل(۱۱-۳) میزان پراکندگی انتخاب و شدت با هم مقایسه می‌شوند.



شکل ۱۱-۳ وابستگی بین پراکندگی انتخاب و شدت انتخاب

با توجه به شکل بالا با افزایش شدت انتخاب مقدار پراکندگی انتخاب در سه روش انتخاب،انتخاب رتبه‌گذاری ، هرسی و مسابقه‌ای کاهش می‌یابدمانند قبل روش انتخاب رتبه‌گزاری روی قسمت ومنطقه ای کار می‌کند که روش انتخاب مسابقه ای کار نمی‌کند و محدوده عمل این دو روش انتخاب از هم جداست زیرا دارای مشخصه مجزا از یکدیگر می‌باشند.

فصـــل چهارم

انواع عملگرهای بازترکیبی

۴-۱- عملگرهای باز ترکیبی **مجزا**

۴-۲- عملگرهای باز ترکیبی **مقادیر حقیقی**

۴-۲-۱- عملگرهای باز ترکیبی **میانی**

۴-۲-۲- عملگرهای باز ترکیبی **خطی**

۴-۲-۳- عملگرهای باز ترکیبی **خطی توسعه یافته**

۴-۳- عملگرهای باز ترکیبی مقادیر باینری

۴-۳-۱- عملگرهای باز ترکیبی **تک نقطه‌ای/دو نقطه‌ای/چند نقطه ای**

۴-۳-۲- عملگرهای باز ترکیبی **یک شکل**

۴-۳-۳- عملگرهای باز ترکیبی **برزننده**

۴- **عملگرهای** باز ترکیبی

باز ترکیبی از ترکیب اطلاعاتی و یا جابجایی ژنها که والدین دارا باشند افراد جدیدی (فرزند) بوجود می‌آورند. این عمل توسط ترکیب مقادیر متغیرهایی از والدین انجام می‌شود.

بخش ۱-۵ عملگر باز ترکیبی مجزا را توصیف می‌کند این عملگر برای هر نوع متغیر ، قابل استفاده است بخش ۲-۵- روش‌های مختلف باز ترکیبی برای مقادیر حقیقی را توضیح می دهد و روش‌های مختلف باز ترکیبی برای متغیر باینری را نیز در بخش ۳-۵ توضیح می‌دهیم.

روش‌های باز ترکیبی برای متغیر با مقادیر باینری مرحله خاصی از باز ترکیبی مجزا می‌باشند. این روش‌ها می‌تواند برای همه متغیرها با مقادیر حقیقی به خوبی قابل استفاده باشد.

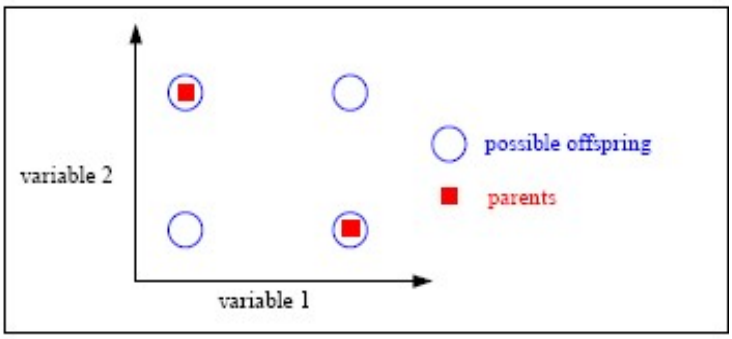
۴-۱-باز ترکیبی مجزا

عملگر باز ترکیبی مجزا روی هر متغیر (حقیقی ، باینری و یا نشانه‌ای) قابل استفاده است. باز ترکیبی مجزا،مقادیر یا ژنهای بین افراد جامعه را تغییر می‌دهد. جابجایی مقادیر متغیرهای بین والدین و ایجاد فرزندان دارای احتمال یکسان هستند. مقادیر متغیرها بین والدین طبق فرمول زیر تغییر می‌کند و فرزند به وجود می‌آید.

$$Var_i^o = Var_i^{p_1} * a_i + Var_i^{p_2} * (1 - a_i)$$
$$a_i \in \{0,1\}, i \in \{1,2,...,NVar\}$$

در رابطه بالا a_i به طور یکنواخت و تصادفی برای هر I انتخاب می‌شود.

باز ترکیبی مجزا، نقاط گوشه‌ای مربع تعریف شده، توسط والدین را تولید می‌کند. شکل زیر با توجه به رابطه (۵-۱) تاثیر هندسی عملگر بازترکیبی مجزا را بر روی والدین و همچنین موقعیتهای ممکن فرزندان بعد از انجام عمل باز ترکیبی مجزا روی والدین را نشان می‌دهد.



شکل ۱-۵ موقعیت های ممکن فرزندان بعد از بازترکیبی مجزا

حال با ارائه مثال زیر روند اجرائی عملگر باز ترکیبی مجزا را بررسی می‌کنیم.
دو فرد زیر را با سه متغیر (ژن) در نظر بگیرید:

فردیا والدین اول: ۵ ۲۵ ۱۲

فرد یا والدین دوم: ۳۴ ۴ ۱۲۳

حال برای هر فرد a_i نظیر به خود را به طور تصادفی در مجموعه [۰و۱] انتخاب می کنیم

$a_1 =$ ۱ ۰ ۰

$a_2 =$ ۱ ۰ ۱

مقادیرمتغیرهای والدین مطابق رابطه (۵-۱) و با توجه به a_i های بالا ترکیب می‌شوند و به طور تصادفی و با احتمال برابر ،فرزندانی به وجود می‌آید که به صورت زیر می‌باشد.

فرزند اول: ۵ ۴ ۱۲۳

فرزند دوم: ۵ ۴ ۱۲

عملگر باز ترکیبی مجزا برای هر نوع متغیر (باینری، حقیقی یا نشانه‌ای) قابل استفاده است.

۴-۲- عملگرهای باز ترکیبی مقادیر حقیقی

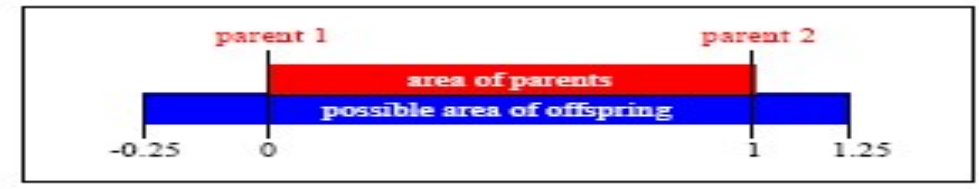
روشهای باز ترکیبی در این بخش برای بازترکیبی افراد با مقادیر (ژنها) حقیقی کاربرد دارد.

۴-۲-۱- عملگرهای باز ترکیبی میانی

عمل باز ترکیبی میانی[6] روشی است که فقط برای متغیرهای حقیقی قابل استفاده است. از این رو مقادیر متغیرهای(ژنها) فرزندان قدری نزدیک وبین مقادیر متغیرهای (ژنها) والدین بدست می‌آید.
عملگر بازترکیبی میانی طبق رابطه زیر عمل می‌کند.

$$Var_i^o = Var_i^{p_1} * a_i + Var_i^{p_2} * (1 - a_i)$$
$$i \in \{1,2,...,NVar\}$$

که در آن a_i به طور تصادفی در بازه [-d,1+d] انتخاب می‌شود.
هنگامی که d=0 انتخاب می‌شودسطح تعریف شده برای فرزندان مشابه اندازه سطح گسترش یافته به وسیله والدین شان می‌باشد . این روش، روش بازترکیبی میانی استاندارد نامیده می‌شود. در عمل بازترکیبی میانی توسعه یافته $d\rangle 0$ یک انتخاب خوب و نرمال است. شکل (۲-۵) نشان دهنده محدود تغییرات متغیر فرزند که توسط متغیرهای والدین تعریف شده است.



شکل ۲-۵ عملگر بازترکیبی میانی

حال با ارائه مثال زیر بیشتر با روند اجرائی عملگر بازترکیبی میانی که فقط بـر روی ژنهـای بـا مقـادیر حقیقی اثر می‌کند آشنا می‌شویم.

فرد یا والدین اول: ۵ ۵ ۱۲

فرد یا والدین دوم : ۳۴ ۴ ۱۲۳

حال مقادیر a_i را به طور تصادفی در بازه [۱/۲۵و۰/۲۵] انتخاب می‌کنیم

a_1= ۰/۵ ۱/۱ -۰/۱

a_2= ۰/۱ ۰/۸ ۰/۵

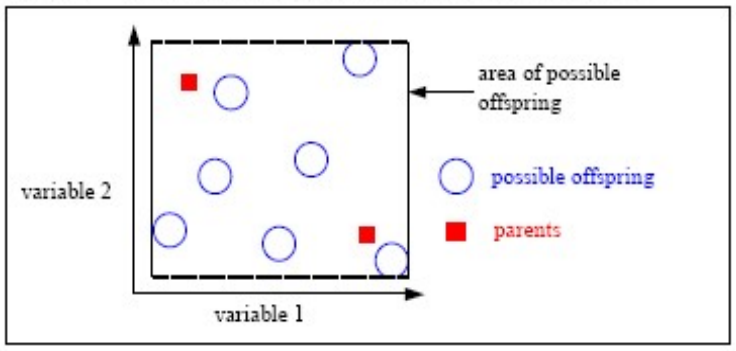
فرزندانی که توسط رابطه (۲-۵)و مقادیر مختلف a_i بدست می‌آیند

فرزند اول: ۵/۶۷ ۱/۹ ۲/۱

فرزند دوم: ۲۳/۱ ۸/۲ ۱۹/۵

عمل باز ترکیبی میانی قادر است که هر نقطه مربع فرزند را کمی بزرگتر از قسمتی که توسـط والـدین تعریف شده است تولید کند.

شکل (۵-۳) سطح که توسط فرزندان بعد از عمل بازترکیبی میانی را اشغال می‌کند را نشان می‌دهد.



شکل ۳-۵ عملگر باز ترکیبی میانی

۴-۲-۲- عملگرهای باز ترکیبی خطی

باز ترکیبی خطی شبیه بازترکیبی میانی[3] است با این تفاوت که تنها یک مقدار برای a همه متغیرها استفاده می‌شوند.

$$Var_i^o = Var_i^{p_1} * a_i + Var_i^{p_2} * (1 - a_i$$

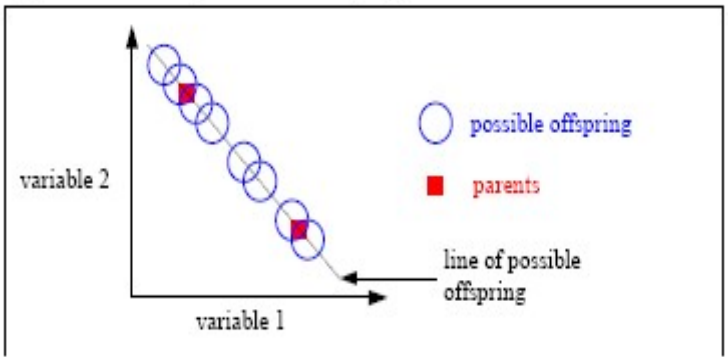
$$i \in \{1,2,..,NVar\}$$

که در آن a به طور تصادفی در بازه [-d,1+d] انتخاب می‌شود.

هنگامی که d=0 انتخاب می‌شود سطح تعریف شده برای فرزندان مشابه اندازه سطح گسترش یافته به وسیله والدین‌شان می‌باشد. این روش ، روش بازترکیبی میانی استاندارد نامیده می شود.

در عمل بازترکیبی میانی توسعه یافته $d\rangle 0$ می‌باشد. d=0/25 یک انتخاب خوب ونرمال است توصیه می شود از این مقدار در محاسبات استفاده شود.

شکل(۲-۵) نشان دهنده محدوده تغییرات متغیر فرزند که توسط متغیرهای والدین تعریف شده است.



شکل ۲-۵ عملگر بازترکیبی خطی

حال با ارائه یک مثال روند اجرائی عملگر بازترکیبی خطی را بررسی می‌کنیم. در آخر می‌توانیم با اجرای برنامه این عملگر بیشتر با روش آن آشنا شوید.

دو والدین زیر که هر کدام شامل سه متغیر (ژن) است در نظر بگیرید:

فرد یا والدین اول: ۵ ۲۵ ۱۲

فرد یا والدین دوم: ۳۴ ۴ ۱۲۳

حال مقادیر a_i را به طور تصادفی در بازه [1/25,-0/25] انتخاب می‌کنیم.

$a_i =$ ۰/۵

$a_2 =$ ۰/۱

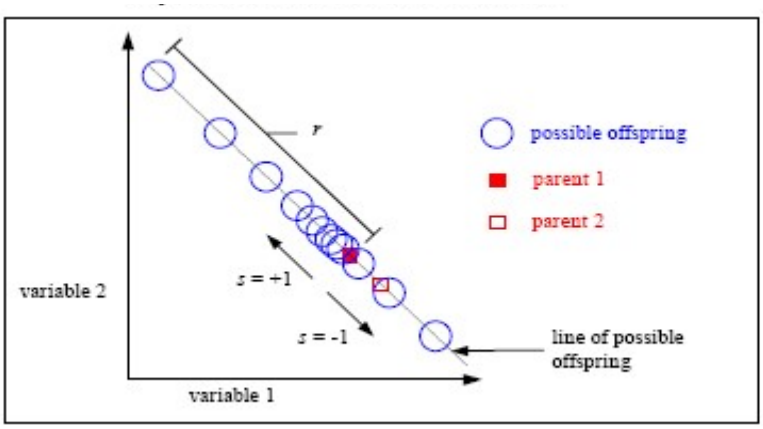
فرزندانی که توسط رابطه (۳-۵) و مقادیر مختلف a_i بدست می‌آیندعبارتند از:

فرزند اول: ۱۹/۵ ۱۴/۵ ۶۷/۵

فرزند دوم: ۷/۹ ۲۲/۹ ۲۳/۱

روش باز ترکیبی خطی قادر است هر نقطه‌ای که روی خطی که توسط والدین تعریف شده است را تولید کند.

شکل (۴-۵) موقعیت‌های ممکن فرزندان را بعد از انجام عمل بازترکیبی خطی را نشان می‌دهد.



شکل ۴-۵ بازترکیبی خطی

۴-۲-۳- عملگرهای باز ترکیبی خطی توسعه یافته

عمل بازترکیبی خطی توسعه یافته[1] فرزندان را در سمت و سوی مقادیر متغیرهای (ژن‌های) والدین به وجود می‌آورند. به هر جهت عمل بازترکیبی خطی توسعه یافته محدود به خط بین والدین وفاصله کوچک بیرون آنها نمی‌باشد و از روی ژن‌های والدین، فرزندان به وجود می‌آیند. میزان سطح ممکن برای ایجاد کردن فرزند به وسیله دامنه‌یمتغیرها تعریف شده است. در منطقه تولید فرزند ، فرزندان به طور یکنواخت و تصادفی ایجاد می‌شود. اگر میزان برازندگی والدین در سطح بالایی باشد، تولید فرزند می‌کند.

فرزندان طبق قانون و رابطه زیر تولید و تکثیر هستند.

$$VAR_I^O = Var_i^R + s_i.r_i.a_i.\frac{Var_i^{p_2} - Var_i^R}{Var^{p_1} - Var^{p_2}} i \in (1,2,...N\,\mathrm{var})$$

$$a_i = 2^{??}, kmutation\ preciosion.u \in [0,1] uniform\ atrandom, a_i\ for\ all\ i\ identical,$$
$$r_i = r\ damainr : ranje\ of\ recombination\ steps,$$
$$s_i \in \{-1+1\}, uniform\ at\ random : underected\ recombination,$$
$$+1 With\ \ probability \rangle 0.5 : directed\ recombination$$

$$r \in [0.110\hat{}-6]$$
$$k \in \{4,5,...,20\}$$

متغیر s_i به طور یکنواخت و تصادفی انتخاب می‌شود.

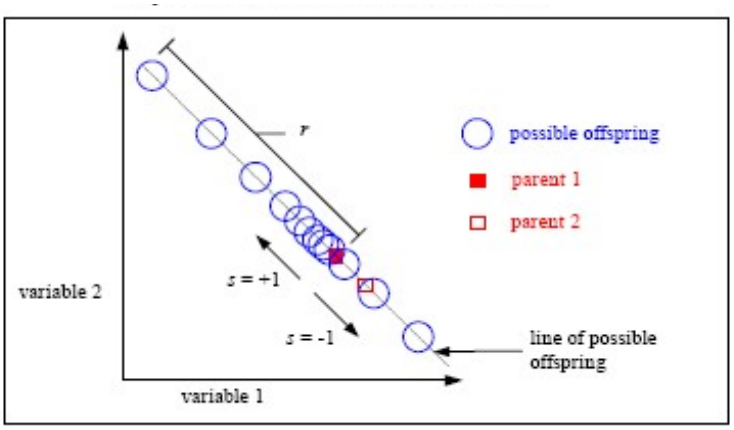
پارامتر a_i متناسب با اندازه گام است و پارامتر s رابطه مستقیم با اندازه گام بازترکیبی دارد.

شکل۵-۵ سعی می‌کند تاثیر روش بازترکیبی توسعه یافته را نشان دهد.

پارامتر k تعیین کننده دقت اندازه گام‌های روش بازترکیبی است. هر چه k بزرگتر باشد اندازه گام کوچکتر می‌شود. بسته به نوع مسئله کاربر می‌تواند مقدار k را خود از اعداد ۴تا ۲۰ انتخاب کند.

برای هر مقدار k، بزرگترین مقدار a برابر ۱ می‌باشد(u=1) و کوچکترین مقدار a که وابسته به k دارد برابر $a = 2^{\wedge} - k$ است (u=1) به طور نمونه مقادیر که برای پارامتر k در نظر گرفته می‌شود در سطحی از ۴ تا ۲۰ می‌باشد.

شکل (۵-۵) موقعیت های ممکن فرزندان بعد از اعمال عملگر بازترکیبی خطی توسعه یافته روی والدین وتعریف سطح متغیرها را نشان می‌دهد که با توجه به مقدار s که مثبت یک یا منفی یک انتخاب شود فرزندان در دو سوی مختلف تکثیر می‌شوند.



شکل ۵-۵ عملگر بازترکیبی خطی توسعه یافته

بهترین مقدار که می‌توانیم برای r انتخاب کرد دخ درصد از دامنه متغیرها است. به هر جهت مطابق تعریف دامنه متغیرها یا برای مراحل خاص این پارامتر انتخاب شودسطحی از والدین که تولید فرزند می‌کند، کوچک می‌شود. همچنین در این روش پارامتر s از مجموعه]۱و۱-[به طور تصادفی انتخاب می‌شود.

۴-۳- بازترکیبی مقادیر باینری

در این قسمت روشهای بازترکیبی برای افراد با مقادیر باینری توصیف می‌شود. معمولاً این روشها، عملگر crossover نامیده می‌شود. بنابراین عبارت crossover به جای نام روش بازترکیبی استفاده می‌شود.

۴-۳-۱- عملگر crossover تک نقطه‌ای

روش این عملگر بدین صورت است که به طور تصادفی عدد صحیح مثبت k (موقعیت عملگر crossover) را از مجموعه [۱-Nvar ...و۲و۱] انتخاب می‌کنیم که در آن Nvar برابر طول افراد می‌باشد و ژنها یا متغیرهای بعد از عدد تصادفی دو والدین با هم جابجا می‌شوند که این تغییر ژنها باعث به وجود آمدن فرزندان جدید می‌شود. حال با ارائه مثال زیر بیشتر با روند کار این عملگر آشنا می‌شویم.

مثال: دو فرد به طول ۱۱ بیت در نظر بگیرید:

۰	۱	۰	۱	۱	۰	۰	۱	۱	۱	۰	فرد یا والدین اول
۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	فرد یا والدین دوم

به طور تصادفی عدد k را در مجموعه [۱۰و...و۲و۱](چون طول کروموزوم یا والدین برابر ۱۱ می‌باشد) انتخاب می‌کنیم.

K=5

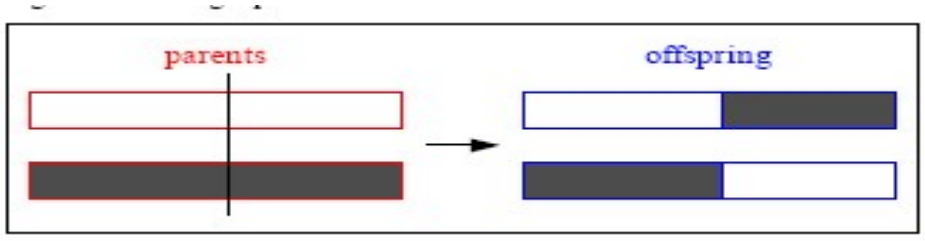
حال محل اثر عملگر را روی دو والدین یا کروموزوم مشخص می‌کنیم.

۰	۱	۰	۱	۱	۰	۰	۱	۱	۱	۰	فرد یا والدین اول
۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	فرد یا والدین دوم

بعد از اعمال عملگر crossover تک نقطه‌ای فرزندان جدید به وجود می‌آیند که عبارتند از:

۱	۰	۱	۰	۰	۱	۰	۱	۱	۱	۰	فرزند اول
۰	۱	۰	۱	۱	۰	۱	۰	۱	۰	۱	فرزند دوم

شکل زیر روش کار عملگر crossover تک نقطه‌ای را نشان می‌دهد.



شکل۱-۳-۵ عملگر crossover تک نقطه‌ای

۴-۳-۲-عملگر crossover دو نقطه‌ای

روش عملگر بدین صورت است که به صورت تصادفی دو عدد صحیح مثبت k_1, k_2 را از مجموعه [۱-nvarو...و۲و۱] انتخاب می‌کنیم که در آن Nvar طول افراد می‌باشد و روش این عملگر مانند عملگر قبل است با این تفاوت که ژنهای بین دو عدد جابجا می‌شودکه باعث به وجود آمدن فرزندان جدید می‌شود. روش این عملگر را با ارائه مثال زیر بیشتر توضیح می‌دهیم.

مثال: دو نفر به طول ۱۱ بیت در نظر بگیرید

۰	۱	۰	۱	۱	۰	۰	۱	۱	۱	۰	والدین اول
۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	والدین دوم

به طور تصادفی عددهای k_1, k_2 را در مجموعه‌های [۱۰و...و۲و۱] انتخاب می‌کنیم.

$k_1 = 3$

$k_2 = 8$

حال محل اثر عملگر را روی دو والدین یا کروموزوم مشخص می‌کنیم.

والدین اول	۰	۱	۱	۱	۰	۰	۱	۱	۰	۱	۰
والدین دوم	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱

بعد از اعمال عملگر crossover فرزندان جدید زیر به وجود می‌آید.

فرزند اول	۰	۱	۱	۰	۱	۱	۰	۰	۰	۱	۰
فرزند دوم	۱	۰	۱	۱	۰	۰	۱	۱	۱	۰	۱

۴-۳-۳- عملگر crossover چند نقطه‌ای

برای عملگر crosover چند نقطه‌ای [6] تعداد m عدد صحیح مثبت k_i در مجموعه [۱-Nvar و...و۲و۱] که در آن $i=1,\ldots,m$ به طور تصادفی انتخاب می‌کنیم سپس مقادیر بین نقاط crossover به طور پیاپی بین والدین جابجا می‌شوند و فرزندان جدید به وجود می‌آورند. حال با ارائه مثال زیر روند این عملگر را بررسی می‌کنیم.

مثال: دو فرد یا کروموزوم زیر را به طول ۱۱ بیت در نظر بگیرید.

والدین اول	۰	۱	۱	۱	۰	۰	۱	۱	۰	۱	۰
والدین دوم	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱

به طور تصادفی عملگر crossover را انتخاب می‌کنیم(m=۳)

$k_1 = 2$

$k_2 = 6$

$k_3 = 10$

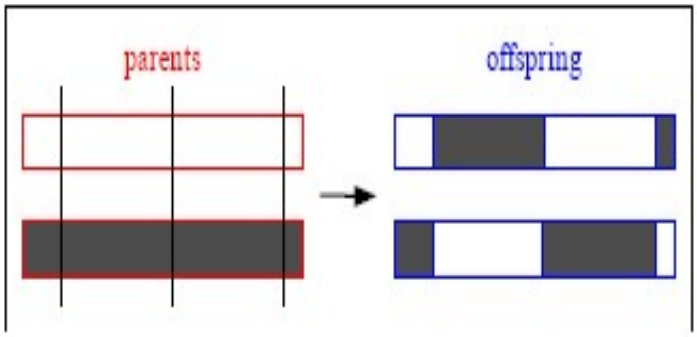
حال محل اثر عملگر را روی دو والدین یا کروموزوم مشخص می‌کنیم.

والدین اول	۰	۱	۱	۱	۰	۰	۱	۱	۰	۱	۰
والدین دوم	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱

بعد از انجام عملگر crossover و تولید فرزند جدید خواهیم داشت.

فرزند اول	۰	۱	۱	۰	۱	۱	۰	۱	۱	۱	۰
فرزند دوم	۱	۰	۱	۱	۰	۰	۰	۰	۱	۰	۱

شکل زیر روش کار عملگر crossover چند نقطه‌ای را نشان می‌دهد.



شکل ۲-۳-۵ عملگر crossover چند نقطه‌ای

۴-۳-۴-عملگر crossover یکنواخت

عملگر crossover تک نقطه‌ای، دو نقطه‌ای و یا چند نقطه‌ای[3] بین نقاطی از مکان هندسی افراد قابل تعریف است، که می‌توان آنها را برش داد . عملگر crossover یکنواخت این عمل را به هر نقطه از مکان هندسی از والدین برای تولید فرزند بهتر تعمیم می‌دهد. روش عملگر crossover یکنواخت به دو صورت است که تفاوت آنها در تعداد الگوئی است که استفاده می‌شود و والدین بر طبق الگوهای ارائه شده تغییر ژن می‌دهند وهمین تغییر ژن باعث به وجود آمدن فرزندان جدید می‌شوند. این دو روش را با ارائه مثال‌های مختلف بررسی می‌کنیم.

دو فرد زیر با متغیرهای باینری در نظر بگیرید:

والدین اول	۰	۱	۱	۱	۰	۰	۱	۱	۰	۱	۰
والدین دوم	۱	۰	۱	۰	۱	۱	۰	۱	۰	۰	۱

در این روش مقادیر فرزندان به طور تصادفی از روی والدین انتخاب می‌شوداز این رو ژن فرزند اول از روی ژن فرد اول یا والدین اول ایجاد می‌شود، هر گاه مقدار بیت نظیر فرد یا والدین اول در الگوی نظیر به آن برابر یک باشد و همچنین بیت نظیر به فرد دوم نظیر صفر باشد و همچنین برای ژن فرزند دوم بر عکس فرزند اول به وجود می‌آید بدین طریق که هر گاه بیت نظیر به فرد دوم برابر یک باشد لذا فرزند دوم از روی فرد یا والدین دوم ایجاد می‌شود.

حال با توجه به توضیحات بالا ما ابتدا دو الگو که رشته‌هائی از مقادیر صفر و یک و با طول مساوی با طول والدین می‌باشد را به طور تصادفی انتخاب می‌کنیم. مانند صفحه بعد:

نمونه اول	۰	۱	۱	۰	۰	۰	۱	۱	۰	۱
نمونه دوم	۱	۰	۰	۱	۱	۱	۰	۰	۱	۰

بعد از اعمال عملگر crosover یکنواخت فرزندان جدیدی به وجود می‌آیند که عبارتند از:

فرزند اول	۱	۱	۱	۰	۱	۱	۱	۱	۱	۱
فرزند دوم	۰	۰	۰	۱	۰	۰	۰	۰	۰	۰

روش دوم به این صورت است که فقط یک الگو برای هر دو والدین انتخاب می‌شود و هر دو والدین طبق این الگو با هم ترکیب می‌شوند (ازدواج می‌کنند) با ارائه مثال زیر بیشتر با این عملگر آشنا می‌شویم.

والدین اول	۰	۱	۰	۱	۱	۰	۰	۱	۱	۰
والدین دوم	۱	۰	۱	۰	۰	۱	۱	۰	۱	۱

حال یک الگو (رشته‌ای با مقادیر صفر و یک به طول مساوی طول والدین) به طور تصادفی انتخاب می‌شود. حال ژنهایی از والدین را جابجا می‌کنیم که بیتهای به آنها در الگو برابر یک باشند، مانند زیر:

الگو با نمونه	۱	۰	۱	۰	۰	۰	۱	۱	۰	۱

حال با توجه به الگو و مطالب گفته شده دو فرزند زیر حاصل می‌شوند.

فرزند اول	۱	۱	۱	۱	۰	۱	۰	۱	۰	۱
فرزند دوم	۰	۰	۰	۰	۱	۰	۱	۱	۱	۰

۴-۳-۵-عملگر crossover بر زدن

عملگر crossover برزدن مانند عملگر crossover یکنواخت است با این تفاوت که مقدار متغیرهای والدین (والدین اول و والدین دوم) قبل از بازترکیبی برزده می‌شودو مقادیر متغیرهای والدین جابجا می‌شودبعد در عمل باز ترکیبی شرکت می کنند. روش کار این عملگر کمی شبیه عملگر crossover یکنواخت است با این تفاوت که دیگر نیاز به الگو نداریم.

فصـــل پنجم

انواع عملگرهای جهش

۵-۱- عملگرهای جهش مقادیر حقیقی

۵-۲- عملگرهای جهش مقادیر باینری

به وسیله جهش افراد به طور تصادفی اصلاح ودگرگون می‌شوند. ایـن تغییـرات خیلـی کوچـک است. ایت تغییرات روی متغیرهایی اعمال می‌شود که دارای احتمال جهش پایین باشد. به طـور معمـول بعد از باز ترکیبی، فرزندان تحت جهش قرار می گیرند . و متغیرهای فرزندان به اندازه یک مقدار تصادفی کوچک افزایش پیدا می‌کند که این مقدار را گام جهش می‌گویند.

۵-۱- عملگرهای جهش مقادیر حقیقی

جهش مقادیر حقیقی بدین معنی است که به طور تصادفی مقدار عددی ایجاد می‌شـود کـه بـه متغیرهای با احتمال پایین جمع می‌شود. بنابراین احتمال جهش یک متغیر و اندازه تغییـر بـرای جهـش متغیر (گام جهش) باید تعریف شود.

احتمال جهش یک متغییر با تعداد متغیر (بعد فرد) رابطه عکس دارد هر چه بعد یک فرد بیشتر باشد، احتمال جهش آن کمتر است. مقالات مختلف نتایج مختلف برای میزان جهش بهینه، گزارش داده‌انـد. در بیشتر این مقالات میزان جهش را $1/n$ (n، تعداد متغیرهای یک فرد) در نظر گرفته‌اند که نتـایج خـوبی برای یک تنوع وسیع از تابع هدف بدست آمده است . این بدین معنی است که تنها یک متغیر هر فرد در هر بار جهش تغییر می‌کند. بنابراین میزان جهش مستقل از اندازه جمعیت است . نتایج مشابهی نیز برای جهش با مقادیر باینری وجود دارد.

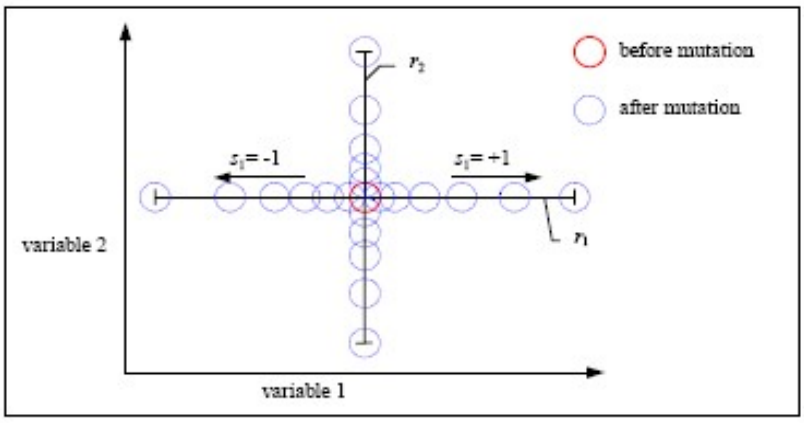
اندازه گام جهش معمولاً به طور مختلف انتخاب می‌شود. بهترین اندازه جهش وابسته به مسئله‌ایست که در نظر می‌گیریم و ممکن است در هنگام اجرای الگوریتم و رسیدن به جواب بهینه تغییر کنـد. مشـخص است که جهش با اندازه گام کوچک اغلب کافی است مخصوصاً وقتی که میزان سازگاری افراد جامعه بـالا باشد. بنابراین یک عملگر جهش خوب، اغلب توسط اندازه گامهای کوچک تولید می‌شود.

رابطه (۱-۶) عملگر جهش برای مقادیر حقیقی به صورت زیر می‌باشد:

$$Var_i^{mt} = Var_i + s_i.r_i.a_i \quad i \in \{1,2,...,n\} \; unifcim \; at \; random$$
$$s_i \in \{-1,+1\} \; uniform \; at \; random$$
$$r_i = r \; domain_i \; .r.mutationrange(s \tan dard : 10\%)$$
$$a_i = 2^{.k}, u \in [0,1] uniform \;\; at \; random, \lambda mutaion \;\; precision$$

بیشتر افراد جهش یافته مقادیرشان نزدیک به افراد قبل از جهش است. تنها تعداد کمی از افراد جهـش یافته از افراد قبل از جهش دور می‌باشند این بدین معنی است که احتمال جهش بـا انـدازه گـام کوچـک بیشتر از احتمال جهش با اندازه گام بزرگتر است.

شکل زیر سعی کرده است که تاثیر عملگر جهش را روی افراد نشان دهد.



شکل ۱-۶ جهش مقادیر حقیقی

پارامتر k (دقت جهش) به طور مستقیم به صورت مینیمم اندازه گام ممکـن، تعریـف می‌شـود و کوچکترین اندازه گام برابر $2^{\wedge} - k$ است و بزرگترین آن برابر $2^{\wedge}0 = 1$بنابراین گام‌های جهش داخل بـازه $r, r^{*}2^{-ki}$ که میزان جهش قرار می‌گیرد.

با دقت جهش k=16 کوچکترین گام جهش ممکن برابر $r.2^{\wedge} - 16$ می‌باشد. بنابراین وقتی متغیرهای یک فرد خیلی نزدیک به بهینه است یک مقدار جهش زیاد لازم نیست.

نوع‌های مختلف برای پارامترهای عملگر جهش برای معادله (۱-۶) موجود است.

$mution \quad precision\ k : k \in \{4,5,...,20\}$

$mutation \quad ranger : r \in \left[0.1.10^{-6}\right]$

۵-۲- عملگرهای جهش مقادیر باینری

برای افراد با متغیرهای باینری[2]، جهش وسیله برای تغییر سریع مقادیر متغیرها زیرا هر متغیر دارای دو موقعیت است بنابراین اندازه گام همیشه برابر ۱ می‌باشد. برای هر فـرد مقـدار متغیـر بـه طـور تصادفی انتخاب می‌شود. نمودار زیر روش جهش یک فرد با ۱۱ متغیر را نشان می‌دهد به طوری که متغیر شماره ۴ انتخاب شده است.

روش بدین صورت است که ابتدا عددی تصادفی در بازه یک و طول فرد (تعداد متغیرهای فرد) در نظر می‌گیریم اگر متغیر فرد برابر ۱ است به صفر تبدیل می‌شود و اگر صفر باشد با ۱ عوض می شـود. ماننـد مثال بالا عدد ۴ را به طور تصـادفی در بـازه‌ی [۱, ۱۱] در نظـر می‌گیـریم و محتـوای خانـه ۴ را عـوض می‌کنیم.

before mutation	0	1	1	*1*	0	0	1	1	0	1	0
				⇓							
after mutation	0	1	1	*0*	0	0	1	1	0	1	0

فصـــل ششم

انواع عملگرهای جایابی مجدد

۶- عملگر جایابی مجدد

۶-۱- عملگر جایابی مجدد عمومی و کلی

۶-۲- عملگر جایابی مجدد محلی

زمانی که فرزندی از والدینش توسط عملگرهای انتخاب، باز ترکیبی و جهش به وجود می‌آید سپس برازندگی افراد محاسبه و ارزیابی می‌شود. برازندگی فرزند تعیین کننده است و اگر فرزند دارای برازندگی کم باشد از جامعه حذف می‌شود لذا اگر تعداد فرزندان تولید شده بیشتر از اندازه جمعیت عمومی، کافی است افراد جامعه قبلی را داخل جمعیت فعلی دوباره بگنجانیم. به طور مشابه اگر تعداد فرزندان تولید شده بیشتر از جمعیت عمومی باشد در این صورت یک طرح دوباره گنجاندن استفاده می‌شود تا اندازه جمعیت فعلی به اندازه جمعیت عمومی برسد.

از عملگر انتخاب برای دوباره گنجاندن استفاده می‌شود مانند دوباره گنجاندن محلی که از عملگر انتخاب محلی و دوباره گنجاندن عمومی و کلی که از همه دیگر عملگرهای انتخاب استفاده می‌کند.

۷-۱- عملگر جایابی مجدد عمومی و کلی

طرح‌های مختلف از دوباره گنجاندن عمومی و کلی [4]موجود است که عبارتند از :

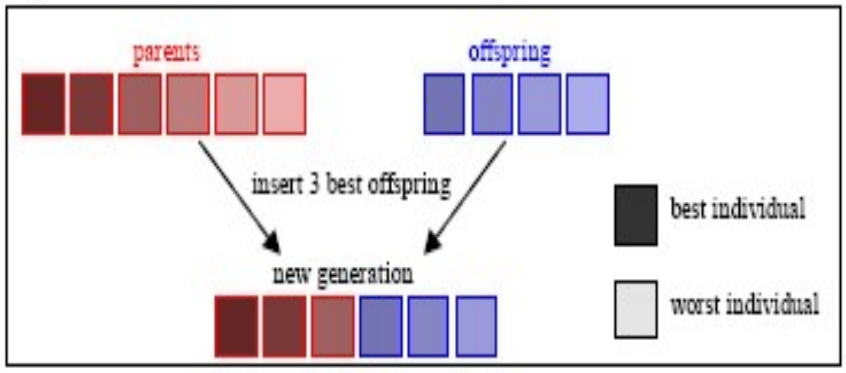
- دوباره گنجاندن خالص: تولید تعداد زیادی فرزند از والدینشان و جایگزین کردن فرزندان به جای والدینشان
- دوباره گنجاندن یکنواخت : تولید فرزند کم از والدین و جایگزین کردن والدین به طور تصادفی به جای فرزندانشان
- دوباره گنجاندن برگزیده: تولید فرزند کم از والدین و جایگزین کردن و الدین قوی
- دوباره گنجاندن بر اساس برازندگی : تولید بیش از حد نیاز فرزندان و دوبباره گنجاندن فقط فرزندان بهتر.

دوباره گنجاندن خالص، ساده‌ترین روش دوباره گنجاندن است. در این طرح هر فرد فقط به اندازه یک نسل زندگی می‌کند. این طرح بیشتر در الگوریتم‌های ژنتیک ساده مورد استفاده قرار می‌گیرد. لذا خیلی محتمل است که افرادی با برازندگی کم به جای افراد مناسب با برازندگی بالا قرار گیرد که این باعث از دست دادن اطلاعات خوب می شود. حال اگر دوباره گنجاندن برگزیده با دوباره گنجاندن بر اساس برازندگی ترکیب شود باعث جلوگیری از بین رفتن اطلاعات می‌شود و این روش توصیه می‌شود.

شکل ۱-۷ نشان می‌دهد که در هر نسل یک تعداد از والدین با برازندگی کم با تعدادی از فرزندان با برازندگی بالا جابجا می‌شود.

الگوی دوباره گنجاندن بر اساس برازندگی، یک انتخاب به روش هرسی بین فرزندان قبل از دوباره جایگزینی آنها در جامعه صورت می‌دهد یعنی قبل از اینکه آنها بتوانند در روند تولید دوباره

شرکت کنند) به عبارت دیگر افراد بهتر می‌توانند تعداد نسل‌های زیادی زندگی کنند. به هر جهت، در هر نسل بعضی افراد جدید دوباره جایگزین می‌شوند واین می‌تواند خود ب اعث کاهش میانگین برازندگی جامعه شود به هر جهت اگر هم فرزندان ناخلف دوباره گنجانده شود در نسل بعدی آنها توسط فرزندان جایگزین می‌شوند.



شکل ۱-۶ الگویی برای دوباره گنجاندن

۶-۲- عملگر جایابی مجدد محلی

در انتخاب محلی، در یک همسایگی کراندار شده انتخاب می‌شوند و در همان همسایگی، دوباره گنجاندن اتفاق می‌افتد. بنابراین محلی بودن اطلاعات پایدار است.

ساختار همسایگی استفاده شده همان ساختار انتخاب محلی است. و همانند آن ابتدا والدین فرزند در آن همسایگی انتخاب می‌شود و سپس عمل دوباره گنجاندن صورت می‌گیرد.

برای انتخاب والدین‌ها برای دوباره گنجاندن و همچنین برای انتخاب فرزندان برای دوباره گنجاندن الگوهای زیر استفاده می‌شود.

- انتخاب هر فرزند و جایگزین کردن بجای افراد داخل همسایگی به طور یکنواخت و تصادفی
- انتخاب هر فرزند و جایگزین کردن به جای افراد ضعیفتر داخل همسایگی
- انتخاب فرزند مناسب تر از افراد ضعیفتر در همسایگی، و جایگزین کردن به جای افراد ضعیفتر در همسایگی
- انتخاب فرزند مناسبتر از افراد ضعیفتر در همسایگی، و جایگزین کردن به جای افراد به طور یکنواخت و تصادفی
- انتخاب فرزند مناسبتر از والدین و جایگزین کردن به جای والدین

فصل هفتم

معیار همگرایی یا توقف الگوریتم های تکاملی

همانند دیگر الگوریتم‌های بهینه‌سازی دیگر، در اینجا نیز باید برای توقف الگوریتم باید از معیار همگرائی استفاده کنیم. کلا شرطهای توقف بر دو نوع‌اند.[۴]و[۵]:

۱. شرط‌های غیرفعال که توقف الگوریتم مستقل از نتیجه بدست آمده از الگوریتم است این شرط را می‌توان به صورت گذشت تعداد ثابتی نسل یا گذشت مدت زمان معینی در هنگام اجرای الگوریتم توسط رایانه، تعریف کرد.

۲. شرط‌های دنباله‌ای که توقف الگوریتم به جواب بدست آمده از آن وابسته است. شرطی از این نوع عبارت است از اینکه جمعیت از حد معینی یکنواخت‌تر شود. برای مثال اگر ۹۸ درصد ژنهای دارای موقعیت مکانی مشابه و مقداری برابر داشته‌باشند، آن الگوریتم همگرا شده فرض می‌شود. این شرط می‌تواند اشتباه برانگیز باشد زیرا ممکن است بر اثر انتخاب نامناسب پارامترهای الگوریتم، الگوریتم خیلی زود به بهینه مناسب برسد.

شرط دیگری را می‌توان با در نظر گرفتن تغییرات برازندگی بهترین فرد جمعتی اجرا نمود، به این معنا که اگر برازندگی در طی چند نسل از حدی بهبود نیابد، اجرای الگوریتم متوقف می‌شود.

دنباله‌ای اطمینان‌بخش تکاملی از هر دو نوع شرط توقف استفاده می‌شود ولی اغلب شرط های دنباله‌ای اطمینان بخشتراز شرط‌های دیگر (غیرفعال) است. ولی باز هم نمی‌توان فاصله جواب بدست آمده را به جواب بهینه کلی تضمین کنند.

ما در حالت کلی از سه شرط توقف در این الگوریتم‌ها استفاده می‌کنیم که عبارتند از :

الف- اگر درصد اختلاف بین میانگین برازندگی تمام افراد جمعت و برازندگی بهترین فرد جمعیت به مقدار بسیار کوچک $C_{\min}$ برسد، الگوریتم متوقف می‌شود . بنابراین :

$$\left|\frac{f^{-} - f_i(best)}{f^{-}}\right| \times 100 \le C_{\min}$$

بطوریکه

$$f^{-} = (\sum_{i=1}^{N} f_i) / N$$

در رابطه قبل f^{-} میانگین برازندگی جمعیت در یک نسل می‌باشد و $f_i(best)$ مقدار تابع هدف برازنده‌ترین کروموزوم، $C_{\min}$ نسبت همگرایی و N اندازه جمعیت می‌باشد.

ب- در صورتی که تعداد نسل که همان متغیر کنترل حلقه می‌بشد به میزان حداکثر مقدار خود که کاربر در نظر گرفته است برسد، اجرای الگوریتم متوقف می‌شود.

ج- در صورتیکه اجرای الگوریتم توسط رایانه از یک مدت زمان مشخصی که توسط کاربر تعیین شده است بگذرد، کاربر رایانه را متوقف می‌کند و لذا الگوریتم متوقف می‌شود.[۲]

فصـــل هشتم

احتمال‌های وفقی عملگرهای الگوریتم تکاملی

برای بهینه‌ساز ی با الگوریتم‌های تکاملی ضروریست کـه خاصـیت را د اشـته ب اشـیم. ا ولـین مشخصه عبارت است از توانائی و ظرفیت همگرا شدن الگوریتم به یک جـواب بهینـه(مطلـق یـا نسـبی) می‌باشد و دومین مشخصه توانایی و ظرفیت جستجوی فضاها و ناحیه‌های جدید برای یافتن نقطه بهینـه می‌باشد. تعادل میان این دو مشخصه از الگوریتم‌های تکاملی توسط مقادیر P_C, P_m و نوع عملگـر تقـاطع که به کار گرفته می‌شود، دارد.[۳]

افزایش مقادیر P_C, P_m میزان فضاهای جستجو را افزایش می‌دهد اما باعث از دست رفتن قابلیت دوم – یعنی توانائی همگرا شدن الگوریتم به جواب بهینه- می‌شود. به طور معمول مقادیر نسبتا بـزرگ احتمـال باز ترکیبـی P_c (۰/۱-۵)و مقـادیر کوچـک(۰/۰۵ و ۰/۰۰۱) بـرای احتمـال جهـش P_m در عمـل بـرای الگوریتم‌های تکاملی مورد استفاده قرار می‌گیرد.

در اینجا هدف ما این است که با بکارگیری مقادیر متغیر P_C, P_m به صورت وفقی مـرتبط بـا برازنـدگی نقاط تعادلی بین جستجوی جواب‌های جدید و همگرائی الگوریتم بیابیم بدین صورت که مقـادیر P_C, P_m زمانی که الگوریتم تمایل به سمت نقطه بهینه موضعی دارند، افـزایش یافتـه و زمـانی کـه پراکنـدگی در جمعیت زیاد شد، کاهش یابند.

محاسبه P_C, P_m وفقی [۷]و[۶]:

برای تغییر دادن P_C, P_mبه صورت وفقی برای جلوگیری از همگرایی زودرس الگوریتم به سـوی جـواب بهینه موضعی، ضروری است که بتوان تشخیص داد که آیا الگوریتم به سوی یک نقطه بهینه همگرا شـده است یا نه.

یکی از راه‌های ممکن برای این کار مشاهده فاصله متوسط برازندگی جمعیت با برازنـدگی مـاکزیمم در جمعیت می باشد یعنی $f_{\max} - \bar{f}$. به نظر می‌رسد که این مقدار برای جمعیتی که به سوی یـک نقطـه بهینه همگرا شده اسـت کمتـر از زمـانی باشـد کـه جمعیـت پراکنـده اسـت. همـان طـور کـه مشـاهده می‌شود $f_{\max} - \bar{f}$ زمانی که الگوریتم به سوی یک نقطه بهینـه نسـبی همگـرا می‌شـود کـاهش یافتـه و برعکس. ما از مقدار $f_{\max} - \bar{f}$ به عنوان یک معیار برای تشخیص همگرایـی الگـوریتم تکـاملی اسـتفاده می‌نمائیم.

از آنجائی که بایستی P_C, P_mزمانی که الگوریتم تکاملی به سوی یک نقطه بهینـه نسـبی همگـرا شـده است افزایش یابند. بنابراین مقادیر آنها بایستی عکس مقدار $f_{\max} - \bar{f}$ تغییر یابد. بنابراین شـکلی کـه مـا برای P_C, P_m در نظر می‌گیریم،به صورت زیر است :

$$P_c = \frac{K_1}{f_{\max} - \bar{f}}$$

$$P_m = \frac{K_2}{f_{\max} - \bar{f}}$$

همانطور که مشاهده می‌شود در حالت فوق مقادیر P_C, P_m بستگی به مقدار برازندگی جواب خاصی از مسئله ندارد و برای تمام نقاط جواب مسئله به صورت یکسان تغییر می‌یابد. در نتیجه جواب‌های با برازندگی بالا و جواب‌های با برازندگی پایین به یک نسبت تحت عمل بازترکیبی و جهش قرار می‌گیرند. زمانی که الگوریتم به سوی یک جواب بهینه مطلق یا نسبی همگرا می‌شود، مقادیر P_C, P_m افزایش یافته و باعث شکستن جواب‌های با برازندگی بالا و جواب‌های با برازندگی پایین به یک نسبت تحت عمل بازترکیبی و جهش قرار می‌گیرند. زمانی که الگوریتم به سوی یک جواب بهینه مطلق یا نسبی همگرا می‌شود، مقادیر P_C, P_m افزایش یافته و باعث شکستن جواب‌های نزدیک به بهینه می‌گردند. در این حالت ممکن است جمعیت هیچگاه به سوی جواب بهینه همگرا نگردد. با وجود اینکه ما از گیرافتادن الگوریتم در دام نقاط بهینه نسبی(موضعی) جلوگیری کرده‌ایم اما کارائی الگوریتم تکاملی را از بین خواهد برد.

برای مقابله با مشکل فوق نیاز داریم که جواب‌های خوب را نگه داریم و این با داشتن مقادیر پایین P_C, P_m برای جواب‌های با برازندگی بالا و مقادیر بالای P_C, P_m برای جواب‌های با برازندگی پایین به دست می‌آید. هنگامی که جواب‌ها با برازندگی بالا به همگرایی الگوریتم تکاملی کمک کرده و جواب‌های با برازندگی پایین از به دام افتادن الگوریتم در دام نقاط بهینه موضعی جلوگیری می‌نماید. مقادیر P_m بایستی نه تنها به مقدار $f_{\max} - \bar{f}$ بلکه به مقدار برازندگی هر جواب بستگی داشته باشد. به طور مشابه P_c بایستی به مقادیر برازندگی هر دو والد بستگی داشته باشد. هرچه مقدار f به $f_{\max}$ نزدیک‌تر باشد مقدار P_m کمتری باید داشته باشیم، بنابراین P_m به طور مستقیم با مقدار $f_{\max} - \bar{f}$ تغییر خواهد کرد و به طور مشابه مقدار P_c بایستی مستقیما با مقدار $f_{\max} - \bar{f}$ تغییر کند. در جائی که f بزرگترین مقدار برازندگی جواب‌هائی است که عمل بازترکیبی روی آنها انجام شده است بنابراین شکل P_C, P_m اکنون به صورت زیر می‌باشد[5]:

$$P_c = \frac{k_1\left(f_{\max} - f\right)}{\left(f_{\max} - \bar{f}^{i}\right)} \qquad k_1 \le 1$$

$$P_m = \frac{k_2\left(f_{\max} - f\right)}{\left(f_{\max} - \bar{f}^{i}\right)} \qquad k_2 \le 1$$

(مقادیر k_1, k_2 بایستی کوچکتر از یک باشند تا مقادیر P_C, P_m در فاصله ۱و۰ قرار گیرند).

در اینجا بایستی توجه کرد که مقادیر P_C, P_m بایستی برای جواب‌های با برازندگی حداکثر، مقدار صفر به خود گیرند.همچنین $P_c = k_1$ برای جواب‌هایی که $f = \bar{f}$ برای جواب‌هایی که $P_c = k_2, f = \bar{f}$. برای جواب‌های با مقدار برازندگی زیر حد متوسط، برازندگی جمعیت، ممکن است به نظر رسد که مقادیر P_C, P_m از مقدار ۱ بیشتر می‌شود. برای جلوگیری از این مسئله ما دو محدودیت زیر را اضافه می‌نمائیم.

$$P_c = k_3 \qquad f \le \bar{f}$$

$$P_m = k_4 \qquad f \le \bar{f}$$

در جائی که $k_3,k_4<1$ هستند.

ملاحظات عملی و انتخاب مقادیر k_1,k_2,k_3,k_4.

در قسمت قبل دیدیم که برای یک جواب با مقادیر برازندگی حداکثر، P_C, P_m هر دو معادل صفر بودند. بهترین جواب جمعیت بدون هیچگونه دستکاری به نسل بعد منتقل می‌گردد. همراه با مکانیزم انتخاب، این عمل ممکن است به یک رشد نمایی جواب‌ها در جمعیت منجر شود و ممکن است باعث همگرائی زودرس گردد. برای غلبه بر این مشکل یک نرخ تعریف شده برای جهش معادل (۰/۰۰۵) برای تمام جواب‌ها معرفی می‌نمائیم[7].

حال به انتخاب مقادیر k_1,k_2,k_3,k_4 می‌پردازیم. برای راحتی کار مقادیر P_C, P_m را به صورت زیر در نظر می‌گیریم:

$$\begin{cases} P_c = \dfrac{k_1\left(f_{\max} - f\right)}{\left(f_{\max} - \bar{f}\right)} & f \ge \bar{f} \\ P_c = k_3 & f \le \bar{f} \end{cases}$$

$$\begin{cases} P_m = \dfrac{k_2\left(f_{\max} - f\right)}{\left(f_{\max} - \bar{f}\right)} & f \ge \bar{f} \\ P_m = k_4 & f \le \bar{f} \end{cases}$$

که دراینجا $k1,k2,k3,k4<1$ هستند.

همانطور که قبلا گفتیم مقادیر نسبتا بالای $p_c (5 < p_c < 1)$ و مقادیر کوچک $p_m (0.001 < p_m < 0.5)$ برای عملکرد موفق الگوریتم تکاملی ضروری به نظر می‌رسد.مقادیر نسبتا بزرگ P_c تکثیر وسیع نمونه‌ها را ترقی می‌دهد، در حالی که مقادیر کوچک P_m برای جلوگیری از تخریب جواب‌ها لازم می‌باشند. این راهنمائی‌ها برای زمانی که متغیر نیستند مفید می‌باشد[3].

یکی از اهداف این روش جلوگیری از به دام افتادن الگوریتم تکاملی در دام نقاط بهینه موضعی می‌باشد. برای رسیدن به این هدف ما نقاط بهینه با برازندگی زیر برازندگی متوسط را به کار می‌گیرند تا فضاهای جستجوی مختلفی را جستجو کنیم تا به ناحیه برسیم که نقطه بهینه در آن قرار دارد. بعضی از این

جوابها کلا بایستی تخریب گردند، بنابراین ما مقدار ۰/۵ را برای k_4 انتخاب می‌کنیم. همچنین جواب‌های با مقدار برازندگی $\bar{f}$ بایستی کاملا بهم بریزیم، بنابراین مقدار ۰/۵ را به k_2 اختصاص می‌دهیم.

بنا به دلایل قبلی مقدار یک را به k_1,k_2 اختصاص می‌دهیم، بدین طریق مطمئن هستیم که کلیه جواب‌ها با مقدار برازندگی کمتر یا مساوی $\bar{f}$ اجرابا تحت عمل بازترکیبی قرار می‌گیرند. احتمال بازترکیبی هنگامی که مقدار برازندگی به سمت $f_{\max}$ میل کند کاهش یافته و برای جواب‌های با برازندگی $f_{\max}$ مقدار این احتمال برابر صفر است.

فصل نهم

نتایج بهروری ضریب قابلیت اطمینان در سیستم های ترابری

۱. مقدمه

در هر جامعه مدرن، مهندسان و مدیران فنی مسئول برنامه ریزی، طراحی ، ساخت و بهره برداری از ساده ترین محصول تا پیچیده ترین سیستم ها هستند. از کار افتادن محصول ها و سیستم ها موجب وقوع اختلال در سطوح مختلفی می شود و می تواند حتی به عنوان تهدیدی شدید برای جامعه و محیط زیست نیز تلقی شود. از این رو مصرف کنندگان به طور کلی مردم جامعه انتظار دارند که محصولها و سیستم ها پایا، اطمینان بخش و ایمن باشد. بنابراین به عنوان یک پرسش اساسی چنین مطرح است که قابلیت اطمینان سیستم در طول عمر کاری آینده اش چه میزانی است و ایمنی آن چقدر است؟

شیوه های ارزیابی قابلیت اطمینان از نظر تاریخچه پیدایش بدواً در ارتباط با صنایع هوا-فضا و کاربردهای نظامی شکل گرفت، ولی سریعاً توسط صنایع هسته ای که تحت فشار شدیدی جهت تضمین ایمنی و قابلیت اطمینان راکتورهای هسته ای در تأمین انرژی الکتریک می باشند و یا صنایع فرایندهای پیوسته مانند صنایع فولاد و صنایع شیمیایی که هر ساعت از توقف آنها به علت وقوع معایب می تواند موجب تحمیل خسارتهای بزرگ مالی و جانی و آلودگی محیط زیست شود مورد توجه و کاربرد قرار گرفت.

در معنای گسترده ، قابل اعتماد بودن یکی از معیارهای اجرای سیستم است. از آنجائیکه سیستم ها پیچیده تر شده اند تأثیرات رفتار، غیرقابل اعتمادشان در زمینه های مختلف آن شدیدتر شده است و گرایش به سوی ارزیابی ضریب اطمینان سیستم ها و نیاز برای بهبود قابلیت اطمینان سیستم از اهمیت زیادی برخوردار شده است برای این منظور تلاش محققان و مهندسین به سمت بهینه سازی قابلیت اطمینان سیستم ها، برای این کار از روش های مختلفی برای بهینه سازی قابلیت اطمینان استفاده شد که با پیشرفت علم و استفاده رایانه این روشها گسترش بیشتری پیدا کردند [۲]و[۱]. یکی از این روشهای بهینه سازی که از طبیعت الگوبرداری شده و از ابزار رایانه نیز بهره می برد الگوریتم های ژنتیک می باشد. اخیراً مهندسان و محققان از این روش بیشتر استفاده می کنند. در این مقاله نیز از الگوریتم های ژنتیک برای بهینه سازی مسئله ای از قابلیت اطمینان که تابع هدف و محدودیت های آنها فازی می باشد استفاده شده است.

در ابتدا لازم است ساختارهای اقبلیت اطمینان را بررسی کنیم سپس روند کلی و عملگرهای مورد استفاده از الگوریتم های ژنتیک را ذکر می کنیم و سپس جواب بهینه تابع هدف را بدست می آوریم. و همچنین مدل بهینه سازی قابلیت اطمینان با چندین الگوی شکست بررسی می شود.

۲.معرفی مدل ریاضی ضریب قابلیت اطمینان

قابل اعتماد بودن یک سیستم را می توان بصورت احتمالی تعریف کرد که سیستم در طول یک زمان مشخص و تحت شرایط ذکر شده بصورت موفقیت آمیزی عمل کرده و نتیجه موردنظر حاصل شود. قابلیت اطمینان دارای ساختارهای سری و موازی می باشند و یا حالت ترکیبی از این دو حالت می باشد لذا لازم است ابتدا ساختار این دو حالت بیان شود.

۱.۲.ساختار سری :

از دیدگاه قابلیت اطمینان، یک مجموعه n عضوی را یک رشته سری می گوئیم هرگاه موفقیت کل سیستم وابسته به موفقیت همه اعضاء مجموعه باشد . در این صورت کل سیستم موفق است اگر و فقط اگر همه اعضای آن سیستم موفق عمل کنند. نمودار چنین سیستمی در شکل (۱-۲) نشان داده شده است . حال فرض کنید X_i پیشامدی باشد که i امین واحد آن موفقیت آمیز عمل کند و $\bar{X}_i$، پیشامدی که i امین واحد آن موفقیت آمیز نباشد در این صورت ضریب قابلیت اطمینان سیستم بصورت زیر تعریف می شود .[10]

$$R = P(x_1.x_2.....x_n) = P(x_1).P(x_2 \mid x_1).P(x_3 \mid x_2, x_1)...P(x_n \mid x_1, x_2, ..., x_{n-1})$$

حال اگر پیشامدها مستقل باشند در این صورت خواهیم داشت .

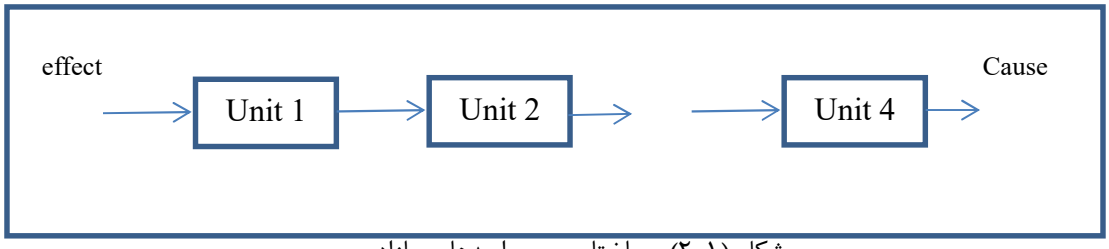
$$R = P(x_1.x_2. \dots .x_n) = P(x_1)P(x_2)...P(x_n) = \prod_{i=1}^{n} P(x_i) = \prod_{i=1}^{n} R_i$$

و نیز ضریب غیرقابل اطمینان بودن سیستم بصورت زیر تعریف می شود :

$$Q = P(\bar{x}_1 + \bar{x}_2 + ... + \bar{x}_n) = 1 - P(x_1.x_2.....x_n) = 1 - R$$

حال اگر پیشامدها مستقل باشند داریم :

$$Q = 1 - R = 1 - \prod_{i=1}^{n} P(x_i) = 1 - \prod_{i=1}^{n} (1 - Q_i)$$



شکل (۱-۲) : ساختار سری واحدهای مازاد

۲,۲. ساختار موازی

از دیدگاه قابلیت اطمینان ، یک مجموعه n عضوی را موازی گوییم هرگاه سیستم موفق است اگر و تنها اگر حداقل یکی از اعضاء موفق باشند که نمودار این سیستم در شکل (۲-۲) مشخص شده است[۹].

حال فرض کنید X_i پیشامدی باشد که i امین واحد آن موفقیت آمیز باشد و همچنین فرض کنید $\bar{X}_i$ پیشامدی باشد که iامین واحد آن موفقیت آمیز نباشد در این صورت ضریب قابلیت اطمینان سیستم بصورت زیر تعریف می شود :

$$R = P(x_1 + x_2 + ... + x_n) = 1 - P(\bar{x}_1 . \bar{x}_2 .... \bar{x}_n) =$$

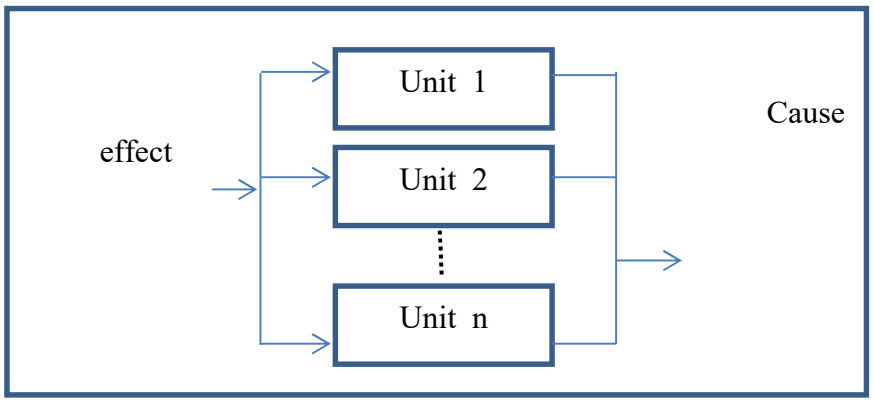
$$1 - P(\bar{x}_1)P(\bar{x}_2 \mid \bar{x}_1)P(\bar{x}_3 \mid \bar{x}_1, \bar{x}_2)...P(\bar{x}_n \mid \bar{x}_1, \bar{x}_2, ..., \bar{x}_{n-1})$$

حال اگر پیشامد ها مستقل باشند در این صورت داریم :

$$R = 1 - \prod_{i=1}^{n} P(\bar{x}_i) = 1 - \prod_{i=1}^{n} Q_i$$

و لذا ضریب غیر قابل اطمینان بودن سیستم بصورت زیر تعریف می شود :

$$Q = \prod_{i=1}^{n} Q_i = \prod_{i=1}^{n} (1 - R_i)$$



شکل (۲-۲) ساختار موازی واحدهای اضافی

۳,۲. نمادها و تعریف ها

حال برای بیان فرمول و مدل ریاضی قابلیت اطمینان یک سیستم ابتدا لازم است چندنماد و تعریف ذکر شود[۸].

نماد ۱ : نماد $\binom{n}{k}_F$ را برای بیان و نماد سیستمی است که عملکر این سیستم شکست می خورد هرگاه k تا از n واحد مازاد آن منجر به شکست شود.

نماد ۲ : نماد $\binom{n}{k}_G$ را برای بیان و نماد سیستمی است که عملکرد این سیستم خوب و مطلوب است هرگاه k تا از n واحد مازاد آن خوب عمل کنند.

حال فرض کنید سیستم (m) دارای N زیر سیستم باشد که هر زیر سیستم i آن دارای m_i عضو باشد لذا داریم $m=[m_1,m_2,\ldots,m_N]$

از طرفی تعداد عضوهای زیرسیستم i (m_i) دارای کران بالای u_i می باشند. برای هر زیر سیستم i، دو نوع الگوی شکست وجود دارد که بصورت زیر تعریف می شوند. الگوی شکست کلاس O: زیرسیستم i دارای کلاس الگوی O است، هرگاه عملکرد این زیرسیستم بصورت $\binom{m_i}{1}_F$ باشد به عبارت دیگر عملکرد این زیرسیستم منجر به خطا می شود هرگاه علمکرد یکی از اعضای دارای خطا باشد.

الگوی شکست A: زیرسیستم i دارای الگوی A است هرگاه عملکرد این زیر سیستم بصورت $\binom{m_i}{1}_G$ باشد به عبارت دیگر عملکرد این زیر سیستم خوب و مطلوب است هرگاه عملکرد یکی از اعضای آن مطلوب باشد .

S_i : تعداد کل الگوهای شکست در زیر سیستم i چه از جنس کلاس O یا کلاس A

h_i : تعداد الگوهای شکست کلاس O در زیر سیستم i ($u=1,2,\ldots,h_i$)

S_i-h_i : تعداد الگوهای شکست کلاس A در زیر سیستم i ($u=h_i+1,h_i+2,\ldots,S_i$

q_{iu} : احتمال شکست الگوی u برای هر واحد در زیر سیستم i

$Q^O_u(m_i)$: احتمال شکست در زیر سیستم i که دارای m_i واحد است برای مدل شکست u از کلاس O

$$Q^O_u(m_i) = 1- (1-q_{iu})^{mi} \qquad u=1,2,\ldots,h_i \qquad (۱-۱)$$

$Q^A_u(m_i)$: احتمال شکست در زیر سیستم i که دارای m_i واحد است برای مدل شکست u از کلاس A

$$Q^A_u(m_i)=(q_{iu})^{mi} \qquad u=h_i+1,h_i+2,\ldots,S_i \qquad (۲-۱)$$

$Q^O(m_i)$: احتمال شکست در زیر سیستم i که دارای m_i واحد است با توجه به الگوهای شکست O

$$Q^O(m_i)=\sum_{u=1}^{h_i} Q^o{}_u(m_i) \qquad (۳-۱)$$

$Q^A(m_i)$: احتمال شکست در زیر سیستم i که دارای m_i واحد است با توجه به الگوهای شکست A

$$Q^A(m_i)=\sum_{u=h_i+1}^{Si} Q^A{}_u(m_i) \qquad (۴-۱)$$

$Q_i(m_i)$: احتمال بروز خطا و شکست در زیر سیستم i که دارای m_i واحد می باشد.

$$Q_i(m_i) = Q^O(m_i)+Q^A(m_i) \qquad (۵-۱)$$

$R(m)$: ضریب قابل اعتماد بودن کل سیستم.

حال با توجه به تعریف اصطلاحات مورد نیاز می توان مدل بهینه سازی ضریب قابلیت اطمینان سیستم با چندین مدل الگوی شکست را بصورت زیر ارائه کرد :

$$\max \quad R(m)= \prod_{i=1}^{N}(1-Q_i(m_i)) \qquad (۶-۱)$$

$$\text{S.t.} \quad G_t(m)= \sum_{i=1}^{N} g_{ti}(mi) \le b_t \qquad (۷-۱)$$

$$1<m_i< u_i \quad \text{integer} , \qquad (۸-۱)$$

$$i=1,2,\ldots,N$$

که در آن b_t مقدار منابع موجود از منبع t $(t=1,2,\ldots,T)$ و همچنین $g_{ti}(m_i)$ مقدار منابع لازم برای زیر سیستم i که دارای m_i واحد مازاد می باشد و در نهایت $G_t(m)$ میزان نیاز کل سیستم از منبع t موقعی که سهم کل عناصر سیستم برابر m می باشد.

در این مدل هدف تعیین واحدهای مازاد هر زیر سیستم (mi) بطوریکه ضریب قابلیت اطمینان کل سیستم به ماکزیمم حد ممکن با توجه به محدودیت های سیستم برسد $(m=[m_1,m_2,\ldots,m_N])$

۲. معرفی الگوریتم های تکاملی

الگوریتم های تکاملی شاخه ای از تکنیک های بهینه سازی اجتماعی هستند که در آنها از سیستم های بیولوژیکی و اصول حاکم بر آنها بهره گیری شده است. اگر چه این الگوریتم ها ارائه دهنده مدل های ساده ای از فرایند بیولوژیکی در شرایط طبیعی هستند، اما در عمل توانائی و کارائی زیادی از خود نشان داده اند. ایده اولیه ارائه الگوریتم های تکاملی استفاده از جمعیت های محدودی از عناصر که هر یک از آنها دقیقا نقطه ای از فضای جستجو را مشخص می کنند، می باشد. که در آن الگوریتم افراد جامعه به وسیله کروموزوم ها مشخص می شوند[7]. جمعیت کروموزوم ها (افراد جامعه) بعد از این وارد مرحله شبیه روند تکاملی موجود در طبیعت می شوند. ابتدا بطور تصادفی جامعه ای از کروموزوم ها پدید می آید و سپس عملگرهای نوترکیبی و جهش و انتخاب و دیگر عملگرهای تکاملی را بسته به نیاز،روی این کروموزوم ها اعمال کرده که نسل جدیدی از کروموزوم ها حاصل می شود و سپس برای این نسل جدید معیارهای بهینگی[6]، بررسی می شود. اگر برای این نسل جدید معیارهای بهینگی برآورده شود که الگوریتم متوقف می شود و بهترین کروموزوم نسل را به عنوان جواب بهینه مسئله و مدل ترابری بدست می آید. در غیر این صورت عملگرهای مختلف ژنتیکی بر روی کروموزوم های داخل جمعیت اعمال می شود و نسل جدیدی حاصل می شوند و این مراحل را تا برآورده شدن معیارهای بهینگی ادامه پیدا می کند. برتری الگوریتم های تکاملی نسبت به روشهای دیگر بهینه سازی باعث شود که تمایل به این الگوریتم ها بیشتر شده است.از ابتدای بوجود آمدن الگوریتم های ژنتیک صاحب نظران این رشته سعی کردند که این الگوریتم ها را ساده تر و قابل فهم تر کنند و نیز مدل هایی از الگوریتم های ژنتیک ارائه دهند که دارای عملیات کمتر و نیز دارای قابلیت های زیاد در زمینه های مختلف بهینه سازی باشند و خروجی های این الگوریتم ها، بهترین جواب را برای مسئله مورد نظر ارائه دهد.

۳. مدل الگوریتم تکاملی ویژه بهینه سازی قابلیت اطمینان سیستم های ترابری

حال فرض کنید مسئله بهینه سازی قابلیت اطمینان را با سه محدودیت غیرخطی با واحدهای مازاد موازی در زیر سیستم های آن که هر زیر سیستم دارای الگوی شکست A می باشد.
مدل ریاضی آن بصورت زیر می باشد :

$$\text{Max} \quad R(m) = \prod_{i=1}^{3}[1-[1-(1-q_{i1})^{m_i+1}] = \sum_{u=2}^{u}(q_{iu})^{mli+1}]$$

$$S.t. \quad G_1(m) = (m_1+3)^2 + (m_2)^2 + (m_3)^2 \le 51$$

$$G_2(m) = 20\sum_{i=1}^{3}(m_i + \exp(-m_i)) \ge 120$$

$$G_3(m) = 20\sum_{i=1}^{3}(m_i * \exp(-m_i/4)) \ge 65$$

$$1 \le m_1 \le 4 \quad , \quad 1 \le m_2, m_3 \le 7$$

جایی که ($m=[m_1,m_2,m_3]$) و کلیه زیر سیستم ها دارای چهار الگوی شکست می باشند($S_i=4$) که از این چهار الگو، یک الگوی آن از کلاس O می باشد. ($h_i=1$) و سه الگوی دیگر از کلاس A هستند.
جدول احتمال بروز هر الگوی شکست برای هر زیر سیستم ($i=1,2,3$)i بصورت زیر آورده شده است :

جدول (۱-۱) احتمال بروز خطا در هر زیر سیستم با توجه به الگوهای خطا

زیر سیستم i	الگوی شکست $S_i=4$, $h_i=1$	احتمال بروز خطا و شکست q_{iu}
1	O	0.04
	A	0.05
	A	0.10
	A	0.18
2	O	0.08
	A	0.02
	A	0.15
	A	0.12
3	O	0.04
	A	0.05
	A	0.20
	A	0.10

۱,۴. نمایش کروموزوم

در الگوریتم های ژنتیک متداول و رایج است که کروموزوم به صورت یک رشته باینری نمایش داده می شود[۴]. برای این مسئله مقادیر صحیح متغیر m_i به صورت رشته باینری نمایش داده می شود. طول رشته وابسته به کران بالای $(u_i)m_i$ از واحدهای اضافی دارد. برای مثال اگر کران بالای $(u_i)m_i$ برابر ۴ باشد ما نیاز به سه بیت باینری برای نمایش m_i داریم در این مثال کرانهای بالایی واحدهای اضافی در هر زیر سیستم برابر $u_1=4, u_2=7$, $u_3=7$ در این صورت هر متغیر m_i نیاز به سه مکان باینری دارد. در این صورت کل بیت های موردنیاز برابر ۹ می باشد.

اگر $m_1=2, m_2=3, m_3=3$ کروموزوم به صورت زیر نمایش داده می شود:

$$V=[x_{33}\ x_{32}\ x_{31}\ x_{23}\ x_{27}\ x_{21}\ x_{13}\ x_{12}\ x_{11}] = [1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0]$$

به طوریکه x_{ij} نمادی از jامین بیت باینری برای متغیر m_i می باشد.

جامعه اولیه : جامعه اولیه کروموزم ها به صورت تصادفی ایجاد می شود. هر کروموزوم شامل ۹ بیت باینری می باشد . یک راه کار تصادفی انتخاب کروموزوم ممکن شکل کروموزوم را از دو نظر غیرقابل قبول کند. یک حالت آن ممکن است محدودیت سیستم را نقض کند یا کران بالای را نقض کند. بنابراین برای جلوگیری از این دو حالت و تولید کروموزوم قانونی روش زیر را اجرا می کنیم[۵]:

```
Prpcedure : Initialization
Begin
For   i=1   to   pop-Size  do
        Produce  a  random  chromosome  vi ;
        If  ( vi  is  not  feasible )   then
        i=i-1;
         end
   end
end.
```

حال یک جمعیت شامل ۵ کروموزوم تشکیل می دهیم

$V_1=[1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1]$ $V_2=[1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1]$ $V_3=[1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0]$

$V_4=[0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1]$ $V_5=[1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1]$

مقادیر صحیح معادل آنها به صورت زیر می باشد :

$V_1=[\overset{m_3}{6}\ \overset{m_2}{2}\ \overset{m_1}{1}]$ $V_2=[4\ 1\ 3]$ $V_3=[5\ 1\ 2]$ $V_4=[3\ 2\ 3]$ $V_5=[4\ 2\ 1]$

ارزیابی کروموزوم ها : از فرمول زیر برای ارزیابی کروموزوم های قانونی و غیرقانونی استفاده می شود :

$$\text{Eval}(V_k)=\begin{cases} R(m) & G_t(m)\le b_t \quad \forall t \quad and \quad 1\le m_i\le u_i \quad \forall_i \\ -M & otherwise \end{cases}$$

V_K، kامین کروموزوم می باشد و m عدد صحیح مثبت می باشد. در این صورت مقادیر برازندگی برای کروموزم ها مانند زیر بدست می آید :

eval(v_3)=0.610062 eval (v_1) = 0.543625 eval (v_2)=0.632703
eval (v_5)=0.589642 eval (v_4)= 0.629119

۲,۴. بازترکیبی:

در این جا از بازترکیبی تک نقطه ای استفاده شده است. فرض کنید احتمال بازترکیبی برابر Pc=0.4 و مجموعه اعداد تصادفی بین صفر و یک به صورت زیرباشد[۶]:

0.550279 0.379650 0.243294 0.494583 0.771811

اگر عدد معادل کروموزم کمتر از 0.4 باشد در این صورت کروموزم معادل آن انتخاب می شود. حال با توجه به اعداد بالا، کروموزم های V_2 , V_3 برای عمل بازترکیبی انتخاب می شود. حال یک عدد تصادفی در فاصله [1,9] انتخاب می کنیم[۷]:

به طور مثال عدد ۳ ،

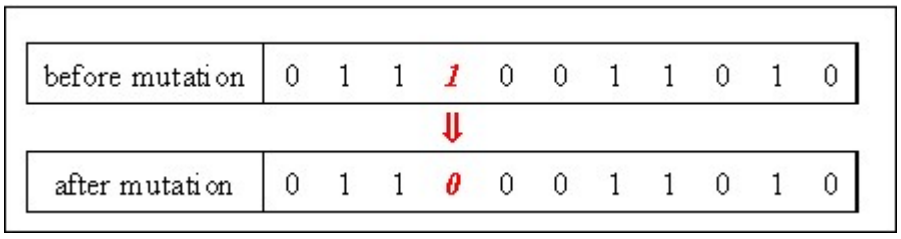
→ O_1 = [1 0 1 0 0 1 0 1 0] V_2=[1 0 0 | 0 0 1 0 1 1]
V_3=[1 0 1 | 0 0 1 0 1 0] O_2= [1 0 0 0 0 1 0 1 1]

مقادیر صحیح آن برابر است با :

O_1= [$\overset{m_3}{4}$ $\overset{m_2}{1}$ $\overset{m_1}{2}$] → eval (O_1)=0.607591
O_2= [5 1 3] eval (O_2) = 0.635277

۳,۴. جهش :

فرض کنید احتمال جهش آنها در کروموزم های یک نسل برابر $P_m=0.1$ باشد این بدین معنی است که ده درصد از کل کروموزم های هر نسل تحت عمل جهش قرار می گیرد[۱۱]. از طرفی ژنهای کروموزومها بصورت باینری می باشند لذا عمل جهش نیز روی صفر و یک ها صورت می گیرد. در این مقاله از عملگر جهش باینری استفاده شده است. روند کلی این عملگر بدین صورت است که یک کروموزم بصورت تصادفی انتخاب می شود سپس یک عدد تصادفی بین یک تا تعداد کل ژنهای کروموزوم انتخاب می کنیم ژن مرتبط اگر مقدار صفر باشد به یک تبدیل و ا گر یک بود به صفر تغییر می کند.



۴,۴. انتخاب:

عملگرجدیدی که در این قسمت معرفی شده است حاصل از ترکیب عملگر انتخاب چرخ گردان و همچنین تکنیک رتبه بندی کروموزوم ها می باشد[6]. در این عملگر ابتدا به هر یک از کروموزوم های نسل یک رتبه داده می شود که روند رتبه بندی برخلاف رتبه بندی متداول می باشد بدین صورت که بدترین کروموزوم رتبه اول و بهترین کروموزوم رتبه آخر می باشد و براساس فرمول زیر برازندگی مربوط به کروموزوم های یک نسل بدست می آید، که در آن sp یک عدد تصادفی بین اعداد یک و دو می باشد و pos رتبه کروموزوم در نسل می باشد.

$$fittnes(pos) = 2 - SP + 2 \times (SP - 1) \times \frac{(POS - 1)}{(N - 1)}$$

حال فرض کنید تعداد کروموزوم های نسل آغازین ما برابر با n باشد در اینصورت رتبه بدترین جواب برابر با یک و به همین ترتیب رتبه بهترین جواب برابر با n می باشد و مطابق فرمول فوق مقدار برازندگی آنها بدست می آید. مانند $f_1, f_2, \ldots, f_n$ حال براساس زیر عمل می شود. ابتدا مقدار

$$F = \sum_{i=1}^{n} f_i$$

بدست می آید و سپس احتمال انتخاب شدن هر جواب را بصورت زیر بدست می آید:

$$p_i = \frac{f_i}{F}$$

و بر اساس آن احتمال تجمعی هر یک از جوابها بصورت زیر محاسبه می شود.

$$q_1 = p_1, q_2 = p_2 + q_1, ..., q_n = q_{n-1} + p_n$$

و سپس آنها را روی یک پاره خط بطول واحد از صفر تا یک تصویر می شود.

حال به تعداد انتخابی که باید انجام شود، عدد تصادفی در فاصله [0 1] انتخاب می شود جوابهائی که نظیر به این اعداد هستند انتخاب می شوند. واضح است که جوابهای قویتر، برازندگی بزرگتری دارند و احتمال انتخاب آنها نسبت به سایر جوابهای یک نسل بیشتر است و این براساس اصل انتخاب طبیعی داروین می باشد که جانداران قویتر باقی می مانند و ضعیف ها محکوم به فنا هستند. حال فرض کنید توسط عملگر انتخاب مدل ترابری تعداد n کروموزم جدید انتخاب شده است که ممکن است تعداد زیادی از آنها تکراری باشند. حال بعد از اعمال عملگر انتخاب، عملگر نوترکیبی را بر روی جوابهای این نسل اثر می کند تا کروموزومهای جدیدی حاصل شود.

۵. مراحل اجرائی الگوریتم

حال ابتدا براساس ناحیه شدنی مسئله نسل صفر یا نسل آغازین را بصورت تصادفی ایجاد می کنیم و مقدار برازندگی آنها رابدست می آوریم سپس برای این نسل معیارهای بهینگی را بررسی کرده اگر معیارهای بهینگی برآورده شود که مراحل الگوریتم متوقف می شود و بهترین کروموزوم آن نسل به عنوان جواب مسئله انتخاب می شود در غیر این صورت عملگر انتخاب بر روی این نسل از کروموزمها اجرا می شود و تعداد n کروموزم جدید انتخاب می شود. بعد از این مرحله عملگر بازترکیبی روی کروموزمهای این نسل اعمال می شود و n کروموزم را به 2n کروموزم تکثیر می کند که این باعث تنوع و تکثر در کروموزمها می شود . بعد از این مرحله عملگر جهش اعمال می شود بدین صورت که n کروموزوم را تبدیل به n کروموزم جدید می کند. طبق روندی که عملگر جهش دارد. لذا بعد از این مراحل تعداد 4n کروموزم جدید حاصل می شود. حال چون تعداد افراد یک نسل باید ثابت باقی بماند لذا از عملگر انتخاب استفاده می شود و تعداد n کروموزم از 4n کروموزم موجود انتخاب می شود و این n کروموزم تشکیل نسل جدیدی را می دهند.حال مانند روندی که در بالا اشاره شد معیارهای بهینگی را برای این نسل بررسی می کنیم اگر معیارهای بهینگی برآورده شود، بهترین کروموزم انتخاب می شود در غیر این صورت

عملگرهای تکاملی بر روی این نسل اعمال می شوند و نسل جدیدی حاصل می شود این مراحل را تا جائی ادامه می دهیم که معیارهای بهینگی برآورده شده و الگوریتم فوق متوقف شود و بهترین جواب حاصل شود. این الگوریتم بر اساس ویژگی های خاص مدل ترابری طراحی شده است و توانائی کاربرد برای مسائل دیگر را نیز دارد[۳].

الگوریتم را در حالت کلی می تواند بصورت زیر گام به گام طراحی کرد

گام اول: ایجاد جوابهایی بطور تصادفی به عنوان نسل آغازین

گام دوم:بدست آوردن مقدار تابع هدف برای این نسل و تعیین برازندگی آنها

گام سوم:ارزیابی معیار بهینگی و درصورت بهینه بودن برو به گام هفتم

گام چهارم:اعمال عملگر بازترکیبی

گام پنجم: اعمال عملگر جهش

گام ششم: اعمال عملگر انتخاب و برو به گام دوم

گام هفتم: بدست آوردن جواب بهینه

۶. نتایج:

امروزه با پیشرفت علوم و توسعه تکنولوژی و پیچیده شدن سیستم های مهندسی که حاصل آن ایجاد یک رفتار غیر قابل اطمینان که کوچکترین رفتار غیر قابل اعتماد اینگونه سیستمها باعث خساران جبران ناپذیر جانی و مالی و محیط زیستی خواهد شد به مانند صنایع نظامی و صنایع هسته ای وصنایع برق. لذا اهمیت و بهینه سازی و بهبود ضریب قابل اعتماد بودن اینگونه سیستم ها که هم دارای هزینه های فراوان است و هم این تکنولوژی ها از اهمیت فوق العاده ای برخوردار است. برای بهینه سازی قابلیت اطمینان سیستم ها از روشهای مختلفی استفاده شده است که در این مقاله مدل خاصی از الگوریتم های ژنتیک که هم دارای ساختار ساده نسبت به دیگر روشهای بهینه سازی دارد و هم توانایی بالایی در بهینه سازی قابلیت اطمینان دارد و جوابهایی که از این روش بدست آمده در مقایسه با دو روش بهینه سازی با استفاده از ضرایب لاگرانژ و همچنین روش بهینه سازی برنامه ریزی خطی به جواب بهینه نزدیکتر بوده و در سیستم قابل اجرایی بهتری دارند.

منابع

[1]. Bazara, S.M., and Jarvis, J.J., Linear programming and Net work flows, by John wiley and sons Inc, 1977.
[2]. Taha, H.A., Operations Research and Introduction, Macmillan, NewYork, 1971.
[3]. Dantzig, G.B., Linear Programming and Extension, Princeton University Press, Princeton, New Jersey, 1964.
[4. Gen, M. and R.Chang,Genetic Algorithms and Engineering designs, John Wiley and Sons,INC.,New Yurok, pp.262-289,1997.
[5]. Pohlheim, H. Genetic and Evolutionary Algorithm Toolbox for use with matlab.www.geatbx.com, 1994-1999.
[6]. Mohammadi, m.The introduction of Evolutionary Algorithms. Unpublished manuscript, M.S.Thesis, Department of Mathematics Shahid Bahonar University of Kerman, Kerman, 2003.
[7]. Deb K., Multi-Objective Optimization using Evolutionary Algorithms, First Edition, John Wiley & Sons Pte Ltd, 2002.