تحلیل و طراحی سیستم ها
الگوریتم ژنتیک

دکتر مهدی روانشادنیا

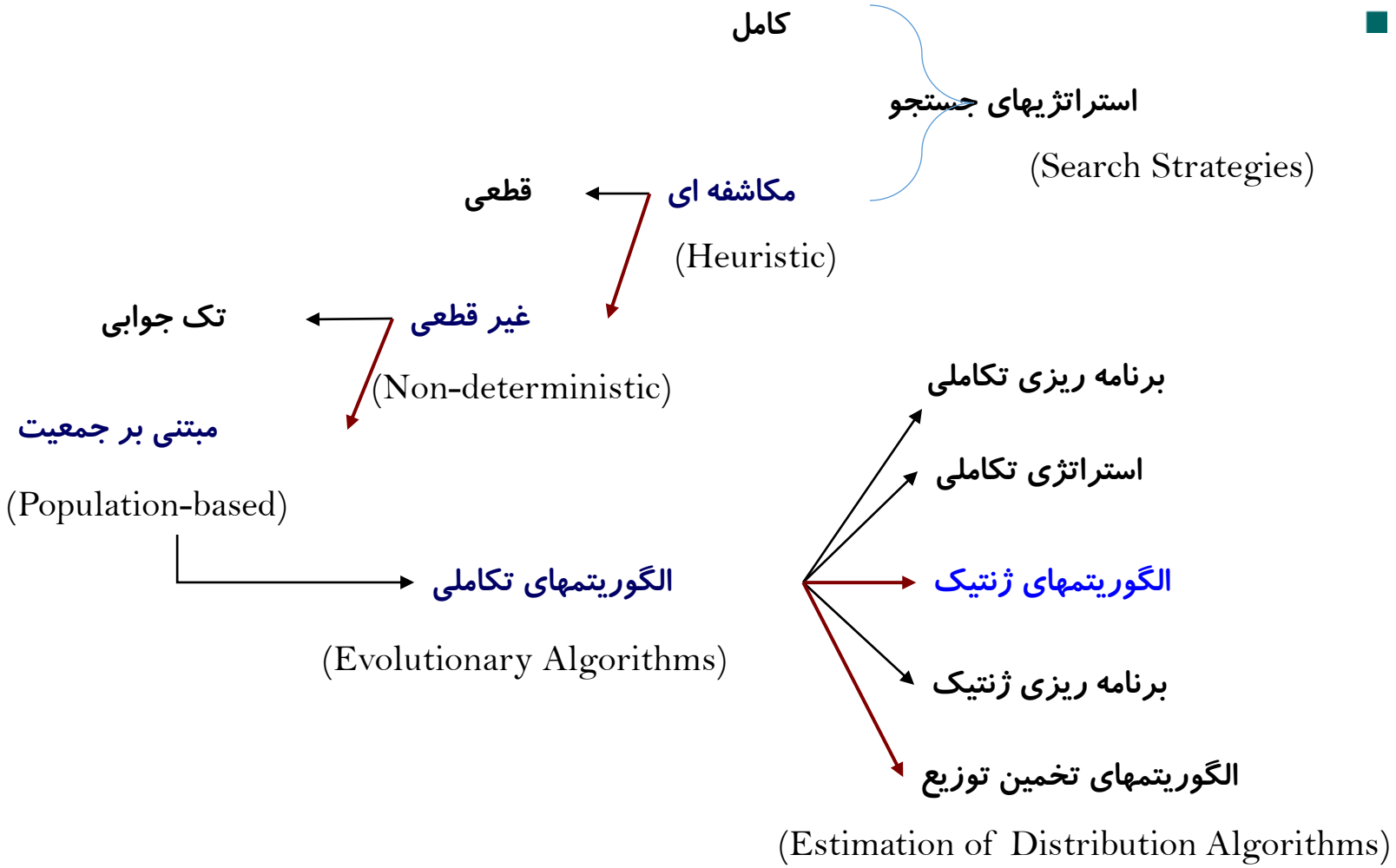
بخش دهم: الگوریتم ژنتیک

- انواع استراتژیهای جستجو برای حل مسأله
- علم ژنتیک
- مبانی الگوریتم ژنتیک
- فرایند مدلسازی و حل مسائل الگوریتم ژنتیک
- کاربرد MATLAB toolbox
- مثال

انواع استراتژیهای جستجو برای حل مسأله

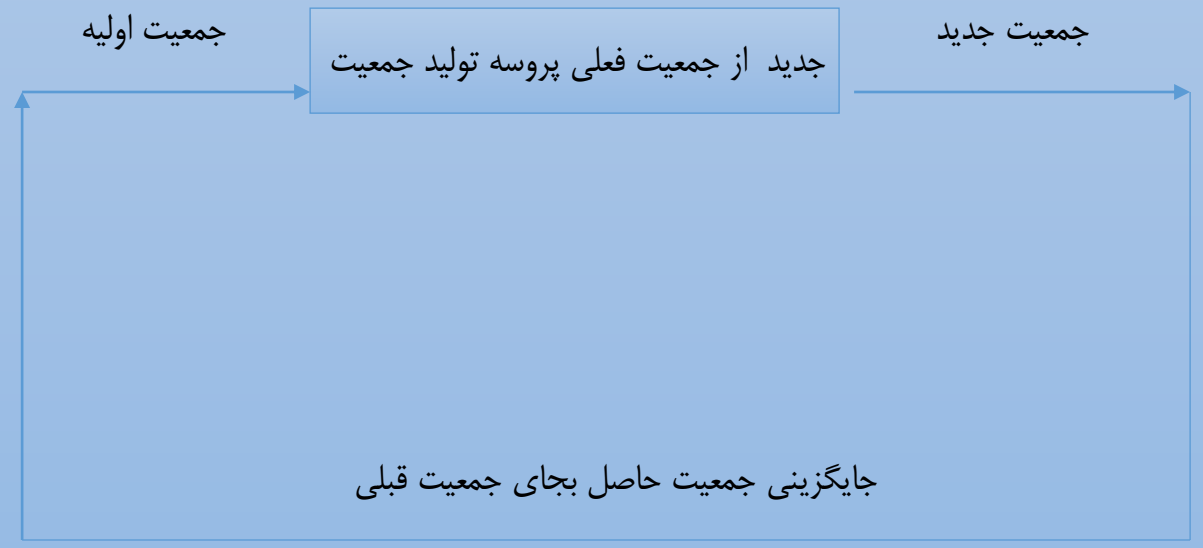
محاسبات
نرم

مقدمه : استراتژیهای جستجو



نمودار گردشی فرآیند یک الگوریتم تکاملی

- فرایند تولید تا وقتی که جواب مورد نظر حاصل شود ادامه می یابد
  (اغلب جمعیت اولیه بصورت تصادفی تولید می شود)



علم ژنتیک

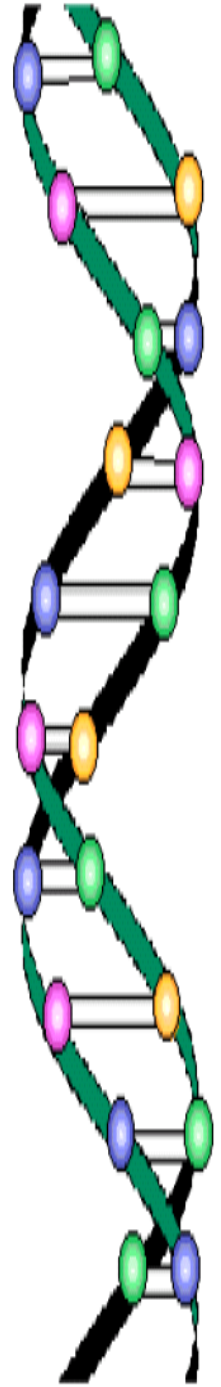
زمینه بیولوژیکی

تمام ارگانیسم های زنده شامل سلول ها هستند. در هر سلول قسمت های مشابهی از کروموزوم ها (Chromosomes) وجود دارند.

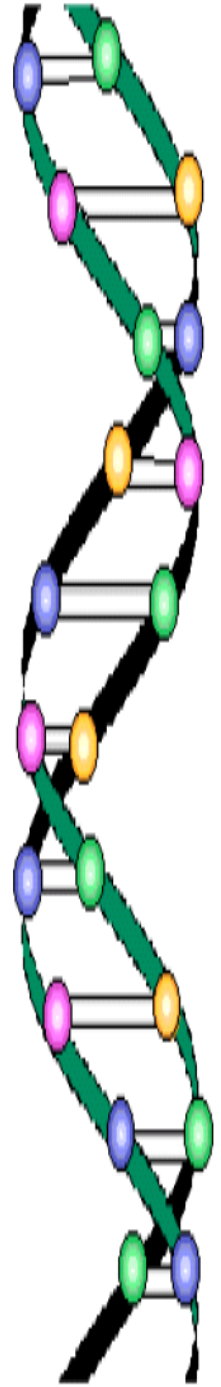
کروموزوم ها شامل ژن ها (Genes) و بلوکهای DNA هستند.

هر ژن (یا دسته ای از ژنها) پروتئین مخصوصی را رمزگذاری (کنترل) می کند، بعبارت دیگر هر ژن (یا دسته ای از ژنها) یک خصیصه (ویژگی) -برای مثال رنگ چشم ها- را رمزگذاری می کند.

امکانِ بروز (کارگذاری) برای یک خصیصه - در مورد رنگ چشمها مثلاً آبی یا قهوه ای – آلل (Allele) نامیده می شود. در واقع یک کروموزوم بصورت زنجیره ای از ژنها قابل تصور است.



زمینه بیولوژیکی



هر ژن موقعیت خاصی در کروموزوم دارد که Locus نامیده می شود.

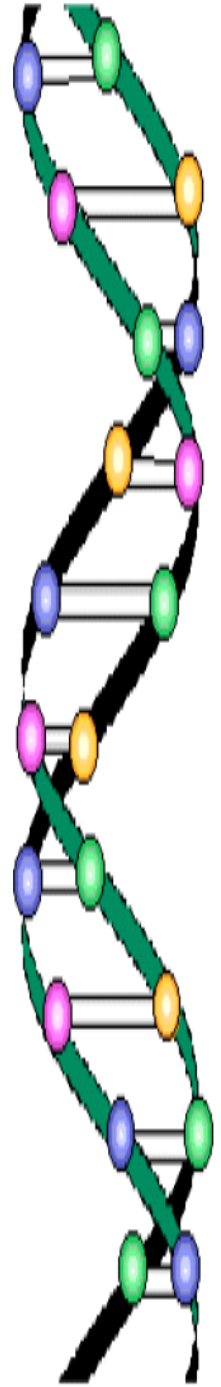
دسته کاملی از ماده ژنتیکی (تمام کروموزوم ها) Genome نام دارد. بخش ویژه ای از ژنها در Genime ژنوتیپ (Genetype) نامیده می شود.

ژنوتیپ با آخرین توسعه بعد از تولد پایه ای برای فنوتیپ (phenotype) ارگانیسمی است.

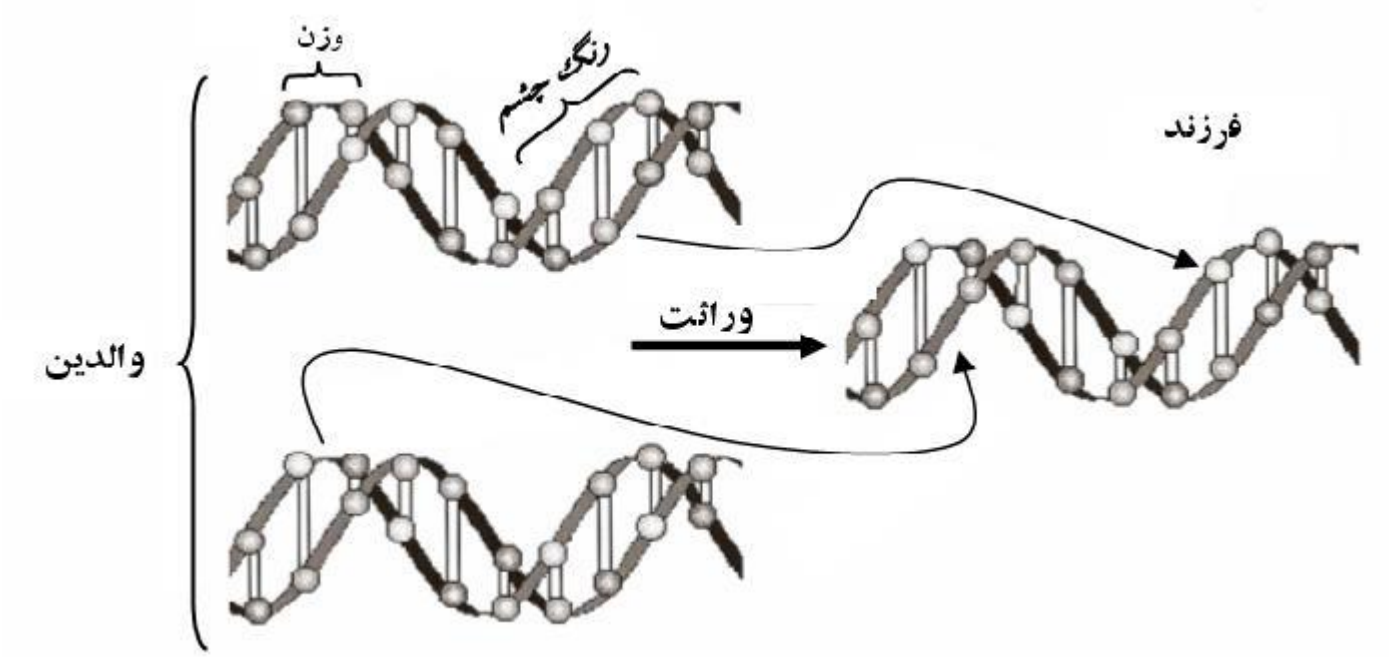
فنوتیپ همان خصوصیات روحی و فیزیکی است (مانند رنگ چشم، هوش و غیره).

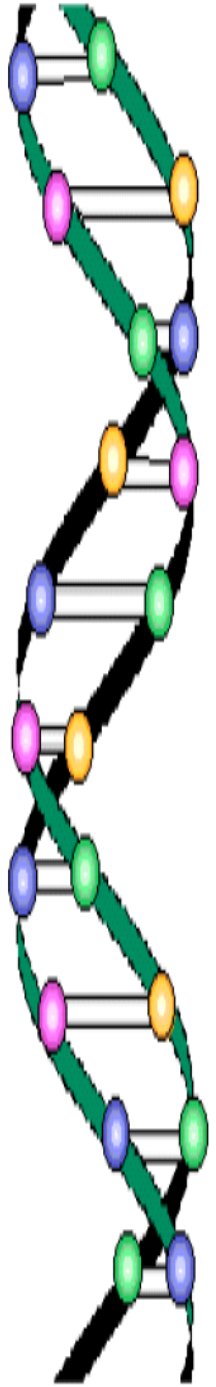
زمینه بیولوژیکی

ژنها (خصیصه ها) می توانند از والدین به فرزندان انتقال یابند. مثلاً ممکن است فرزند رنگ چشم را از یکی از والدین (مثلاً مادر) و طول قد را از دیگری (مثلاً پدر) به ارث ببرد.شکل صفحهٔ بعد نمایشی از این موضوع را نمایش می دهد.



زمینه بیولوژیکی
(نمومه ای از وراثت طبیعی)





مبانی الگوریتم ژنتیک

چند اصطلاح (زمینه بیولوژیکی)

کروموزوم(Chromosome): در هر سلول مجموعه ای از موجودات هم شکل بنام کروموزوم وجود دارد .

ژن(Gene) : هر کروموزوم از تعدادی ژن تشکیل یافته است ؛ هر ژن یک خصیصه را کد می کند.(مثل رنگ چشم)

آلل(Allele) : مجموعه های ممکن برای یک خصیصه آلل نامیده می شود.

لوکس(Locus): هر ژن در کروموزوم موقعیت خاصی را داراست.

ژنوم(Genome) : مجموعه کامل همه کروموزوم ها.

ژنوتیپ(Genotype) : یک مجموعه خاص از ژن ها در ژنوم .

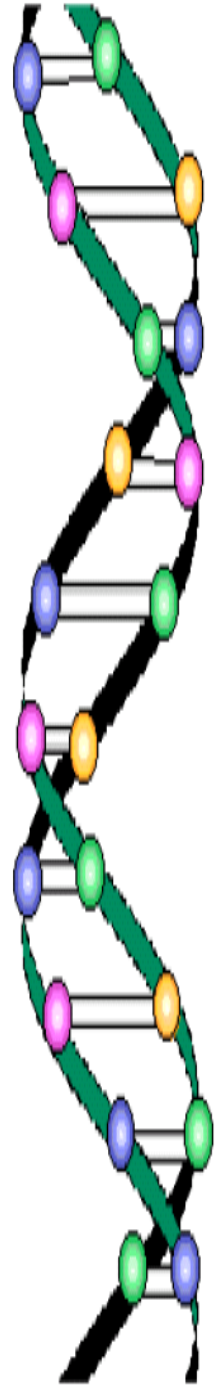
فنوتیپ (Phenotype): ژنوتیپ ها بعد از تکامل بیشتر به فنوتیپ ها (که همان خصوصیات فیزیکی و روانی مانند رنگ چشم یا هوش و ...)تبدیل می شوند .

الگوریتم ژنتیک چیست؟

- ✓الگوریتم ژنتیک روش یادگیری بر پایه تکامل بیولوژیک است. این الگوریتم ها با الهام از روند تکاملی طبیعت مسائل را حل می نمایند . یعنی مانند طبیعت یک جمعیت از موجودات را تشکیل می دهند و با اعمالی بر روی این مجموعه به یک مجموعه بهینه و یا موجود بهینه دست می یابند.
- ✓این روش در سال 1970 توسط John Holland معرفی گردید
- ✓این روشها با نام Evolutionary Algorithms نیز خوانده میشوند. الگوریتم های ژنتیک یکی از شاخه های پردازش تکاملی می باشند.
- ✓با توجه به خصوصیات خاص خودشان به خوبی از عهده حل مسائلی که نیاز به بهینه سازی دارند بر می آیند.

زخی الگوریتمهای طبیعی مبتنی بر جمعیت

- ✓ الگوریتم های تکاملی : EAs
- ✓ الگوریتم تکاملی : EA
- ✓ الگوریتم های ژنتیکی : GAs
- ✓ الگوریتم ژنتیکی : GA
- ✓ برنامه نویسی ژنتیکی : GP



ایده کلی

- یک GA برای حل یک مسئله مجموعه بسیار بزرگی از راه حلهای ممکن را تولید میکند.
- هر یک از این راه حلها با استفاده از یک " تابع تناسب "مورد ارزیابی قرار میگیرد.
- آنگاه تعدادی از بهترین راه حلها باعث تولید راه حلهای جدیدی میشوند .که اینکار باعث تکامل راه حلها میگردد.
- بدین ترتیب فضای جستجو در جهتی تکامل پیدا میکند که به راه حل مطلوب برسد
- در صورت انتخاب صحیح پارامترها، این روش میتواند بسیار موثر عمل نماید.

ویژگی ها

الگوریتم های ژنتیک در مسائلی که فضای جستجوی بزرگی داشته باشند میتواند بکار گرفته شود.

- همچنین در مسایلی با فضای فرضیه پیچیده که تاثیر اجزا آن در فرضیه کلی ناشناخته باشند میتوان از GA برای جستجو استفاده نمود.
- برایdiscrete optimization بسیار مورد استفاده قرار میگیرد.
- امکان به تله افتادن این الگوریتم در مینیمم محلی کمتر از سایر روشهاست.
- از لحاظ محاسباتی پرهزینه هستند.
- تضمینی برای رسیدن به جواب بهینه وجود ندارد.

تفاوت GA با سایر روشهای جستجو

- GAبجای کد کردن پارامترها مجموعه آنها را کد میکند
- GAبجای جستجو برای یک نقطه بدنبال جمعیتی از نقاط میگردد.
- GAبجای استفاده از مشتق و یا سایر اطلاعات کمکی مستقیما ازاطلاعات موجود در نتیجه بهره میگیرد.
- GAبجای قوانین قطعی از قوانین احتمال برای تغییر استفاده میکند.

کاربردها

- بهینه سازی
- اقتصاد مهندسی
- تحقیق در عملیات
- آموزش شبکه های عصبی
- ارزیابی های محیط زیستی
- سیستم های اجتماعی
- حل مسائل هوش مصنوعی
- رباتیک
- سایر کاربردها...

مزایای استفاده از الگوریتم های ژنتیک

✓ این نکته که الگوریتم های ژنتیک خوب یا بد هستند، تا حد زیادی به مساله مربوط می شود .

این الگوریتم ها پارامترهای بسیار زیادی دارند که با تنظیم صحیح این پارامترها می توان نتایج بسیار متفاوتی بدست بیاوریم.

✓ مزایای الگوریتم های ژنتیک

- این الگوریتم ها همیشه یک جواب نسبتاً خوب پیدا خواهند کرد
- در هر مرحله از کار می توانیم الگوریتم را متوقف کنیم. در این حالت نیز یک جواب خواهیم داشت.
- به راحتی می توانیم این الگوریتم ها را بصورت موازی بر روی چند پردازنده اجرا کنیم
- فهم آسان و مجزا بودن
- پشتیبانی از بهینه سازی چند تابعی
- همیشه یک جواب داریم که با گذشت زمان بهتر می شود
- روشهای مختلفی برای افزایش سرعت و پیشرفت الگوریتم وجود دارد.
- بهره برداری ساده از جواب قبلی.
- انعطاف پذیر برای کاربرد های ترکیبی.

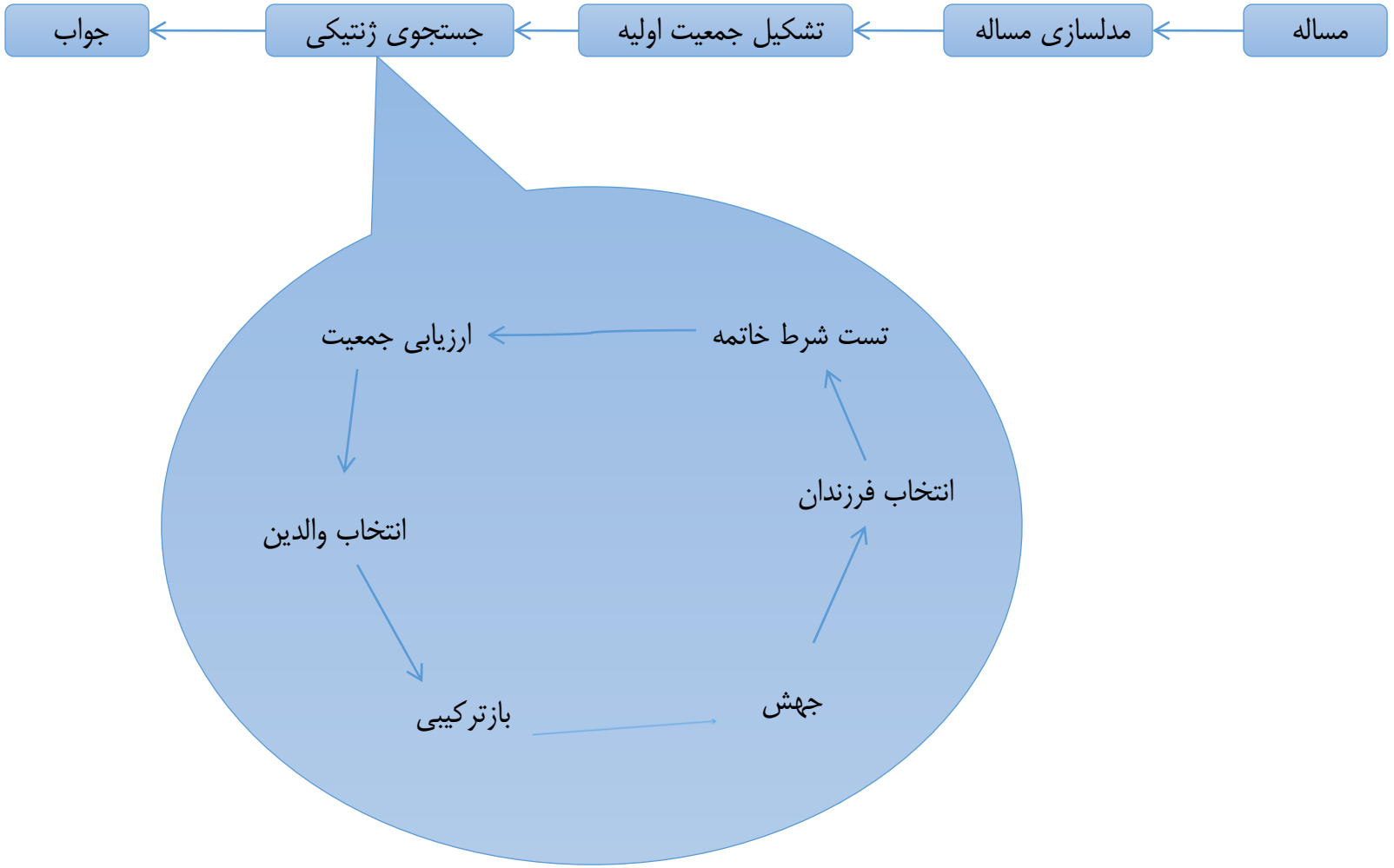
معایب الگوریتم های ژنتیک

✓معایب

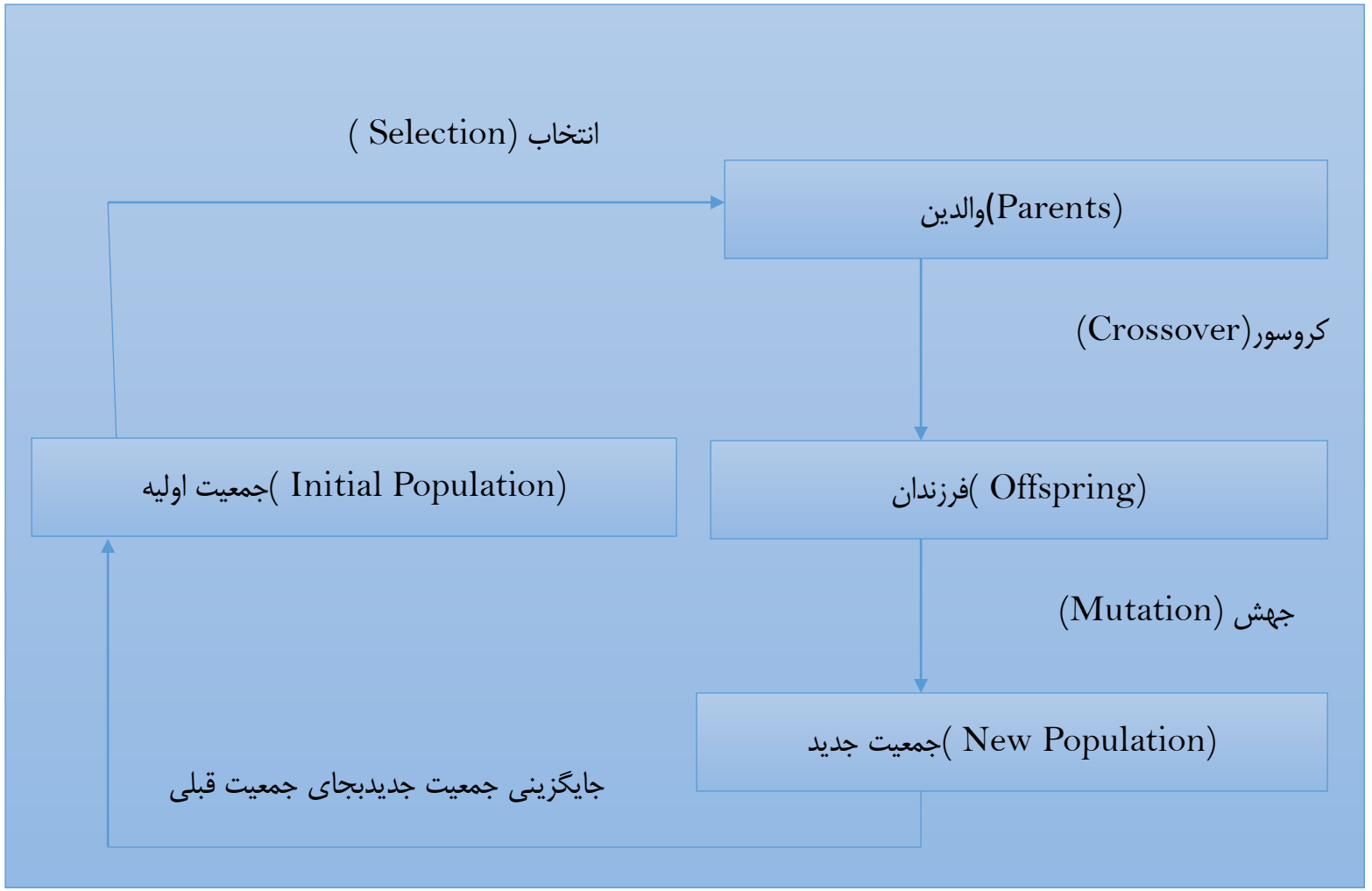
- یک جواب خوب پیدا می کنند ولی ممکن است جواب بهینه را پیدا نکنند
- به حافظه و محاسبات زیادی نیاز دارند
- در مورد اینکه جواب پیدا شده چقدر خوب است و آیا جواب بهتری وجود دارد، نمی توانیم هیچگونه ادعائی داشته باشیم
- پشتوانه ریاضی ضعیفی دارند
- در دو بار اجرای مختلف، جواب های متفاوتی دریافت می کنیم
- تعدد پارامترهای الگوریتم: هر چند الگوریتمهای ژنتیکی در دسته الگوریتمهای بهینه‌سازی قرار می‌گیرند، اما تنظیم کردن این پارامترها گاهی به یک مساله بهینه‌سازی دیگر تبدیل خواهد شد.
- نحوه کدینگ، نوع عملگر ترکیب، نرخ ترکیب، نوع عملگر جهش، نرخ جهش

فرایند مدلسازی و حل مسائل الگوریتم ژنتیک

ساختار الگوریتم های ژنتیک



ساختار الگوریتم های ژنتیک



ساختار الگوریتم های ژنتیک

برای حل یک مساله با الگوریتم های ژنتیک مراحل ذیل را داریم:

۱)مدلسازی مساله یا بازنمایی

۲)تشکیل جمعیت اولیه

۳)ارزیابی جمعیت

۴)انتخاب والدین

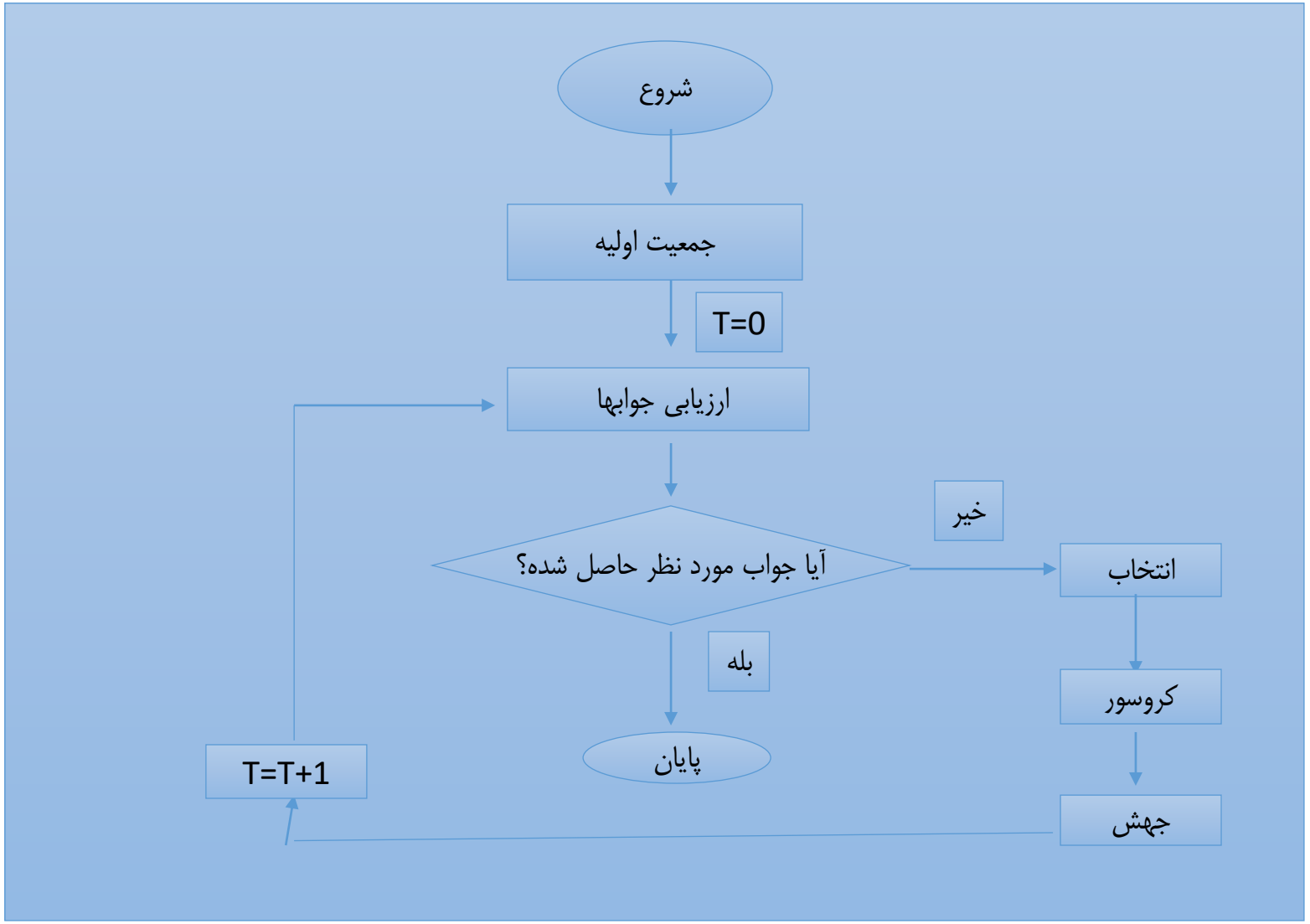
۵)باز ترکیبی

۶)جهش

۷)انتخاب فرزندان

۸)تست شرط خاتمه الگوریتم

مکانیزم یک الگوریتم ژنتیک



GAجدول هم ارزی مفاهیم بیولوژیکی و عناصر

GA	سیر تکاملی طبیعی
محیط	مساله برای حل
جمعیت	مجموعه جوابهای جایگزین
تولید	گام تکرار
فرد	جواب داوطلب
والدین	جوابهای برگزیده شده
اندازهٔ انطباق	مقدار برازندگی (تابع ارزیاب)

GAجدول هم ارزی مفاهیم بیولوژیکی و عناصر

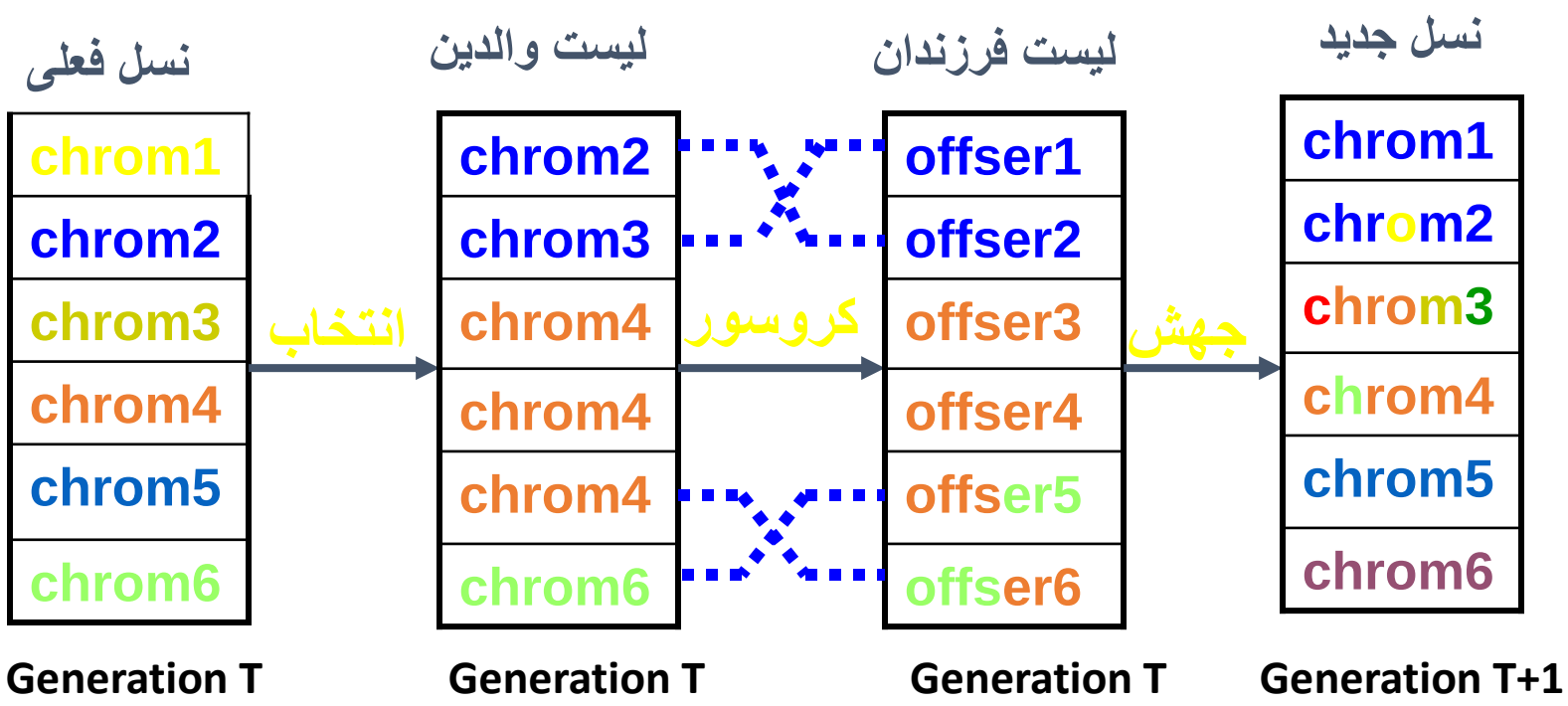
GA سیر تکاملی طبیعی

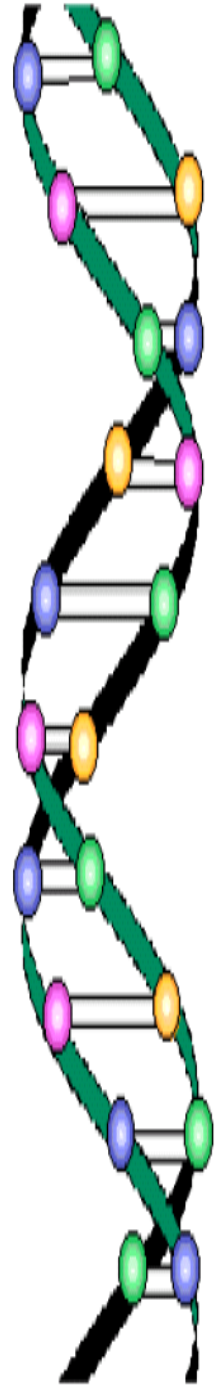
بقاء	انتخاب والدین متناسب با مقدار برازندگی
ژنوتیب	جواب رمزگذاری شده
فنوتیب	جواب رمزگشایی شده
ژن	موقعیت در رشته
آلل	مقدار در یک موقعیت معین در رشته
تناوب تولید	پروسه رسیدن از جمعیت فعلی به جمعیت بعدی

شبه کد یک الگوریتم ژنتیک ساده

```
BEGIN
        t=0;
        Initialize P(t);          {جمعیت اولیه ایجاد می‌شود}
        Evaluate P(t);            {عناصر P(0) توسط مقادیر برازندگی نشاندار می‌شوند}

        WHILE شرایط خاتمه ارضا نشده DO

           BEGIN

              t=t+1;
              Select P(t) from P(t-1);   {اجرای عملگر انتخاب و لیست والدین فراهم می‌شود}
              Crossover P(t);            {اجرای عملگر کروسور و لیست فرزندان فراهم می‌شود}
              Mutation P(t);             {اجرای عملگر جهش و لیست جمعیت جدید حاصل می‌شود}
              EvaluateP(t);              {عناصر P(t) توسط مقادیر برازندگی نشاندار می‌شوند}

           END

    END.
```

روند اجرای عملگرها





مدلسازی مساله (بازنمایی)

برای اینکه بتوانیم یک مساله را بوسیله الگوریتم های ژنتیک حل کنیم، بایستی آنرا به فرم مخصوص مورد نیاز این الگوریتم ها تبدیل کنیم.

✓در این روند ما بایستی راه حل مورد نیاز مساله را به گونه ای تعریف کنیم که قابل نمایش بوسیله یک کروموزوم باشد.

✓اینکه چه نوع بازنمائی را برای مساله استفاده شود، به شخص طراح و فرم مساله بستگی دارد.

✓ چند نمونه از بازنمائی هایی را که معمولاً استفاده می شوند :

1. اعداد صحیح
2. رشته های بیتی
3. اعداد حقیقی در فرم نقطه شناور
4. اعداد حقیقی به فرم رشته های بیتی
5. یک مجموعه از اعداد حقیقی یا صحیح
6. ماشینهای حالت محدود
7. هر فرم دیگری که بتوانیم عملگرهای ژنتیک را بر روی آنها تعریف کنیم

عملگر های الگوریتم ژنتیک

ارزیابی جمعیت Fitness

برای اینکه بتوانیم موجودات بهتر را درون جمعیت تشخیص بدهیم بایستی معیاری را تعریف کنیم که بر اساس آن موجودات بهتر را تشخیص دهیم. به این کار، یعنی تعیین میزان خوبی یک موجود، ارزیابی آن موجود می گویند.

ارزیابی، اینگونه است که بر حسب اینکه موجود چقدر خوب است یک عدد به آن نسبت می دهیم، این عدد که برای موجودات بهتر بزرگتر (یا کوچکتر) است را شایستگی آن موجود می نامیم.

به عنوان مثال در صورتی که به دنبال مینیمم یک تابع هستیم، مقدار شایستگی را می توانیم ورودیهایی که مقادیر تابع برای آنها کمتر است در نظر بگیریم که ورودیهای بهتری هستند.

✓بسته به نوع مساله ما می خواهیم شایستگی را بیشینه و یا کمینه کنیم.

عملگر های الگوریتم ژنتیک

- انتخاب Selection (انتخاب والدین)

✓ سوق دادن جستجو به بخشهایی از فضا که امکان یافتن جوابهای با کیفیت بالاتر وجود دارد.

✓ نسل جدیدی از راه حل ها را با انتخاب والدینی که بالاترین Fitness را دارند تولید می کند.

والدین : در هر نسل تعدادی از عناصر جمعیت این فرصت را پیدا می کنند که تولید مثل کنند. به این عناصر که از میان جمعیت انتخاب می شوند، والدین می گویند.

روشهای مختلفی برای انتخاب والدین وجود دارند.

روش های انتخاب والدین

- انتخاب تمام جمعیت بعنوان والدین:

در واقع هیچگونه انتخابی انجام نمی دهیم (همه عناصر انتخاب می شوند) .

- انتخاب تصادفی:

بصورت تصادفی تعدادی از موجودات جمعیت را بعنوان والدین انتخاب می کنیم، این انتخاب می تواند با جایگذاری یا بدون جایگذاری باشد. در این روشها عناصر با شایستگی بیشتر شانس بیشتری برای انتخاب شدن بعنوان والدین را دارند.

- سایر روشها:

این روشها با استفاده از تکنیک هایی سعی می کنند که انتخاب هایی را ارائه دهند، که هم رسیدن به جواب نهایی را تسریع کنند و هم اینکه کمک می کنند که جواب بهینه تری پیدا شود.

روش های انتخاب

معمول ترین روش های انتخاب

: Elitist Selection

مناسب‌ترین عضو هر اجتماع انتخاب می‌شود.

:Roulette Selection

یک روش انتخاب است که در آن عنصری که عدد برازش (تناسب) بیشتری داشته باشد، انتخاب می‌شود.

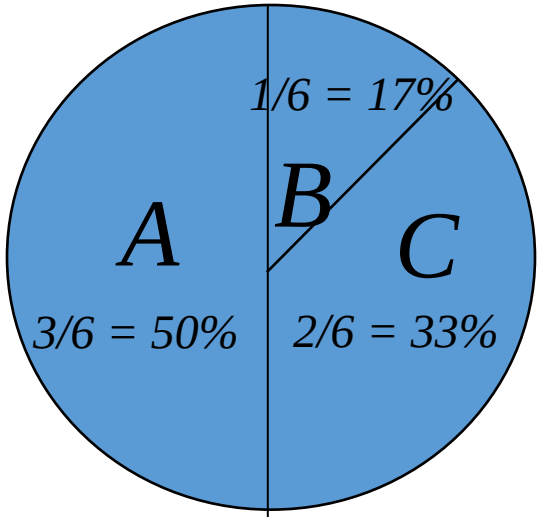
ScalingSelection

به موازات افزایش متوسط عدد برازش جامعه، سنگینی انتخاب هم بیشتر می‌شود و جزئی‌تر. این روش وقتی کاربرد دارد که مجموعه دارای عناصری باشد که عدد برازش بزرگی دارند و فقط تفاوت‌های کوچکی آن‌ها را از هم تفکیک می‌کند.

Tournament Selection

یک زیر مجموعه از صفات یک جامعه انتخاب می‌شوند و اعضای آن مجموعه با هم رقابت می‌کنند و سرانجام فقط یک صفت از هر زیرگروه برای تولید انتخاب می‌شوند.

انتخاب – چرخ رولت



fitness(A) = 3

fitness(B) = 1

fitness(C) = 2

- *Roulette Wheel selection* در روش معرفی شده در الگوریتم ساده *GA* احتمال انتخاب یک فرضیه برای استفاده در جمعیت بعدی بستگی به نسبت *fitness* آن به *fitness* بقیه اعضا دارد. این روش *Roulette Wheel selection* نامیده می‌شود.

$$P(h_i) = Fitness\ (h_i) / \Sigma_j\ Fitness\ (hj)$$

عملگر های الگوریتم ژنتیک

ترکیب مجدد (Recombination/Crossover)

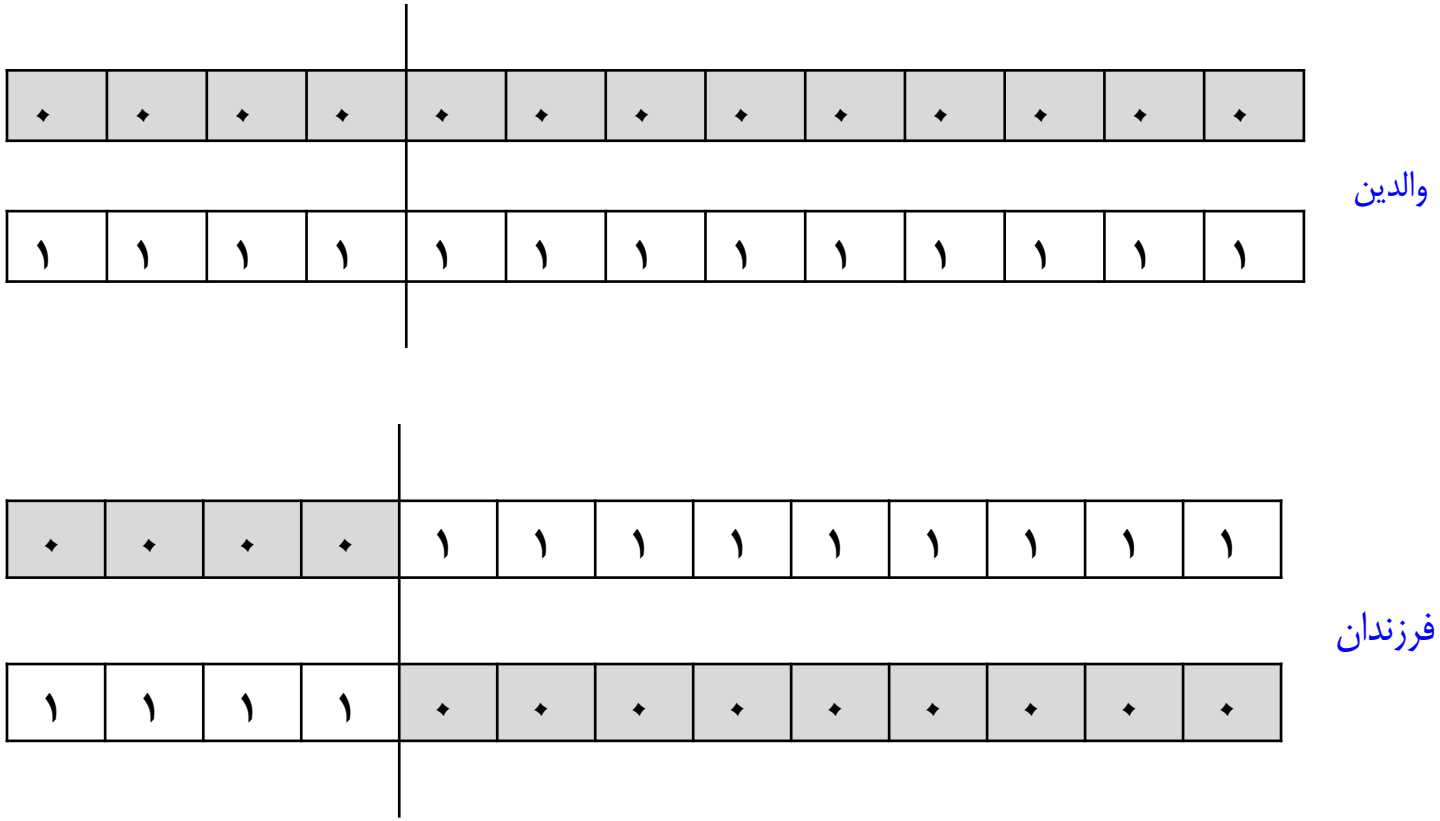
- امکان ترکیب جوابهای جزیی (partial solutions) یافت شده و در نتیجه بدست آوردن جوابهایی با کیفیت بالاتر را فراهم می‌آورد.

در جریان عمل بازترکیبی به صورت اتفاقی بخشهایی از کروموزوم ها با یکدیگر تعویض می شوند.

این موضوع باعث می شود که فرزندان ترکیبی از خصوصیات والدین خود را به همراه داشته باشند و دقیقاً مشابه یکی از والدین نباشند.

هدف تولید فرزند جدید می باشد به این امید که خصوصیات خوب دو موجود در فرزندشان جمع شده و یک موجود بهتری را تولید کند.

روش انجام عمل بازترکیبی



نحوه انجام عملیات بازترکیبی

روش انجام عمل بازترکیبی

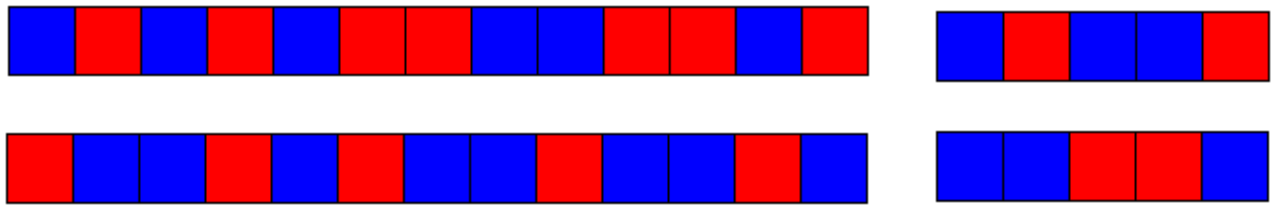
روش کار به صورت زیر است:

بصورت تصادفی یک نقطه از کروموزوم را انتخاب می کنیم
ژن های مابعد آن نقطه از کروموزوم ها را جابجا می کنیم

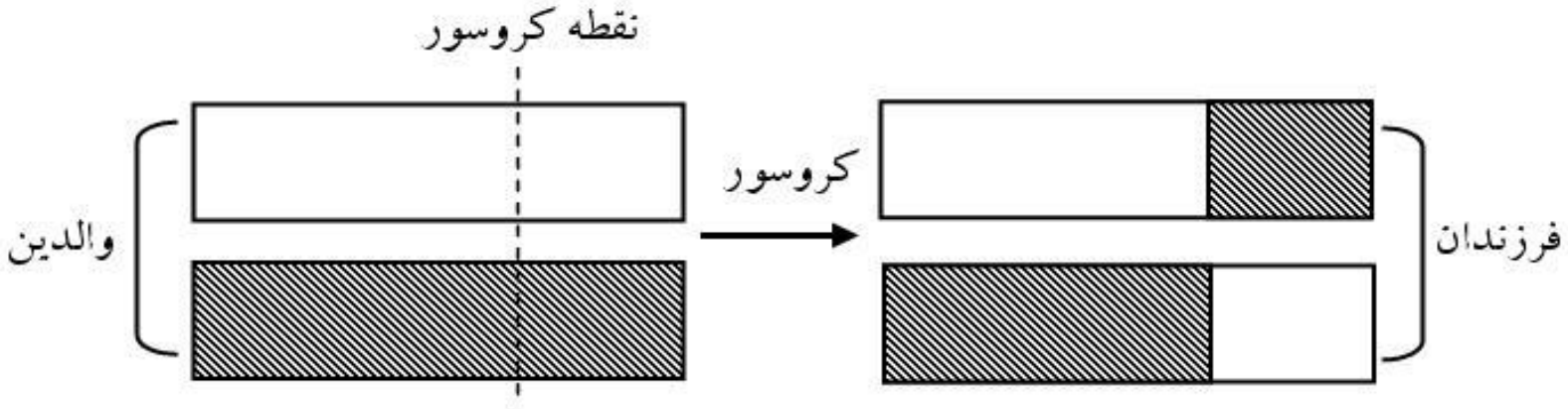
بازترکیبی تک نقطه ای (Single Point Crossover)

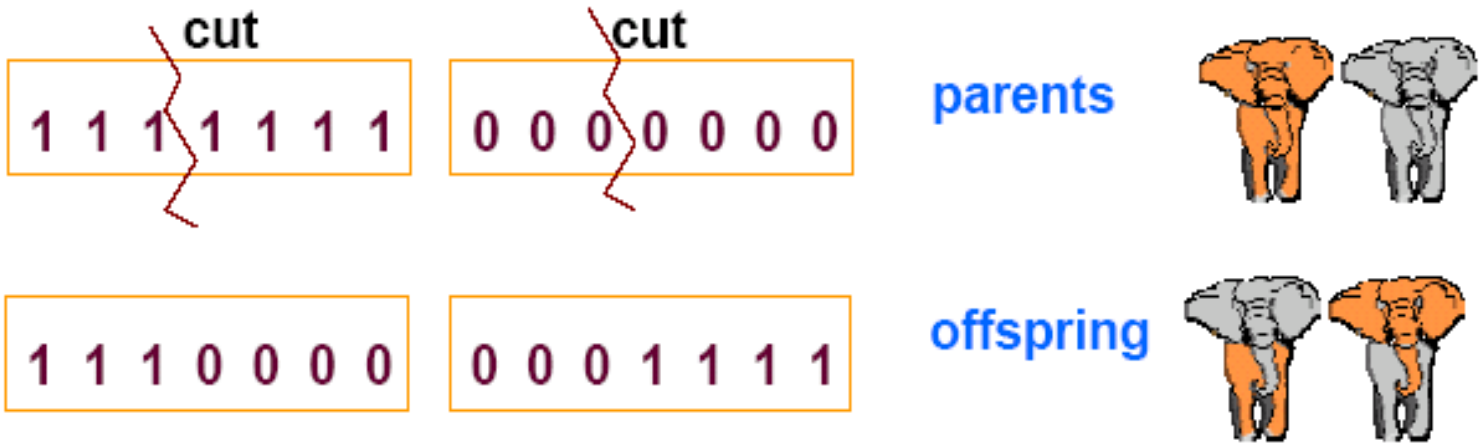
اگر عملیات بازترکیبی را در یک نقطه انجام دهیم به آن بازترکیبی تک نقطه ای می گویند.

ها از این مکان به بعد جابجا می شوند.geneدر این روش یک مکان تصادفی در طول رشته انتخاب می شود و



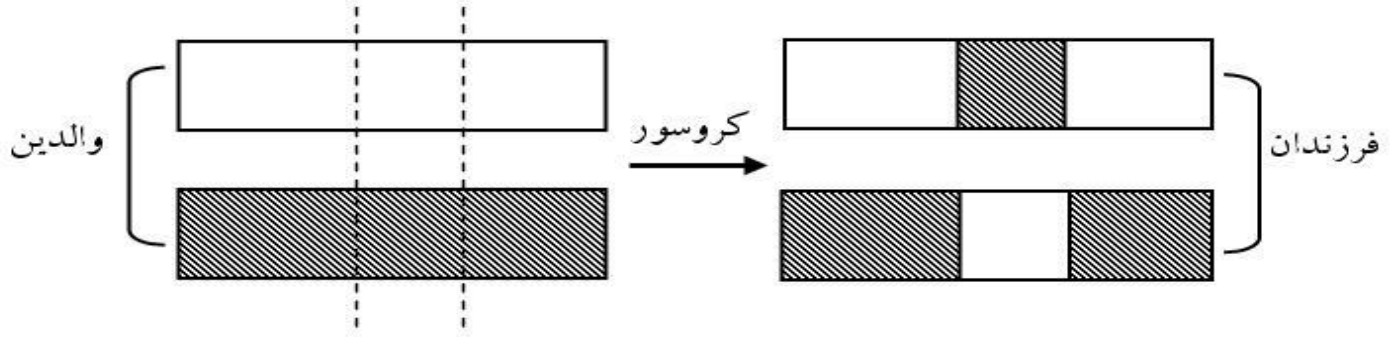
روش انجام عمل بازترکیبی

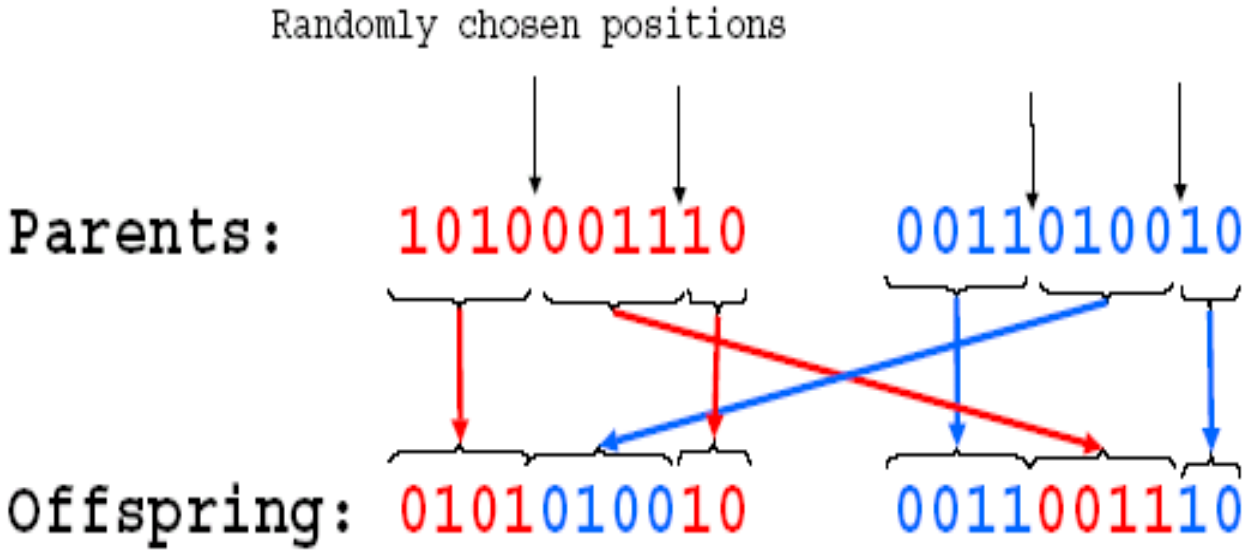


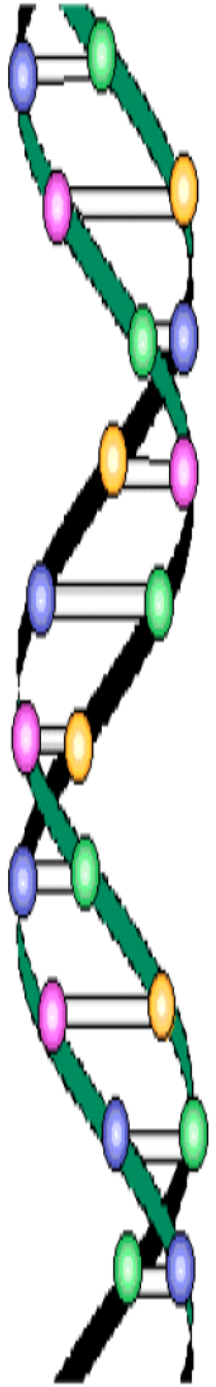


کروسور در رمزگذاری دودویی

- دونقطه ای (Two Point)

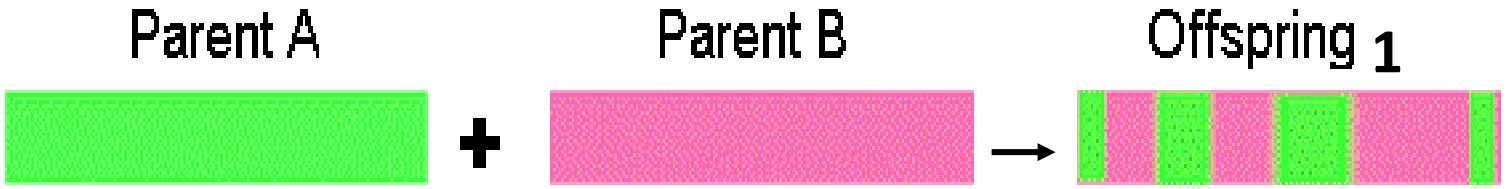






کروسور در رمزگذاری دودویی

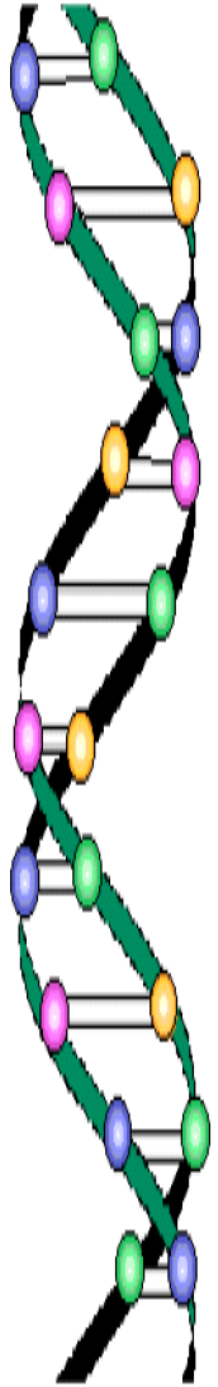
- یکنواخت (Uniform)



Mask: 0110011000 (Randomly generated)

Parents: 1010001110 0011010010

Offspring: 0011001010 1010010110



روش انجام عمل بازترکیبی

ادغام چند نقطه ای (Multipoint Crossover) :

می توانیم این عملیات را در چند نقطه انجام دهیم ، که به آن بازترکیبی چند نقطه ای می گویند .

ادغام جامع (Uniform Crossover)

اگر تمام نقاط کروموزوم را بعنوان نقاط بازترکیبی انتخاب کنیم به آن بازترکیبی جامع می گوئیم.

روش کار را برای این دو مورد اخیر بدین صورت است:

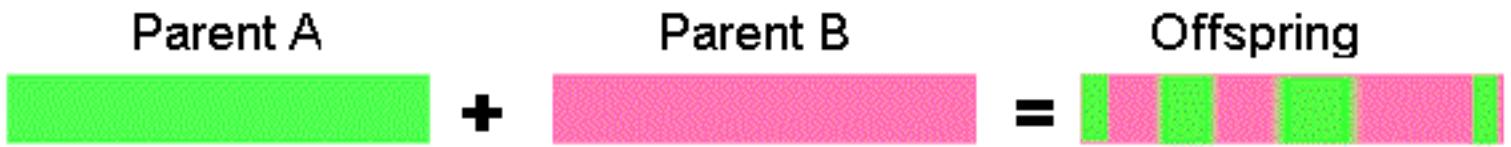
با احتمال ثابتی مثل P_c عمل بازترکیبی را انجام می دهیم

روش کار به صورت زیر است:

- به ازای هر یک از قسمت های کروموزوم:
  - یک عدد تصادفی بین صفر و یک تولید می کنیم
  - اگر این عدد از مقدار ثابتی مثل P_c کوچکتر باشد، ژنهای مابعد آن نقطه از کروموزوم ها را جابجا می کنیم.

یک نکته در مورد عمل بازترکیبی

نمونه ای از ادغام جامع:



❊عملیات بازترکیبی موجودات جدیدی تولید نمی کند و تنها باعث می شود که موجودات موجود بهتر شوند.

❊در صورتی که برای بازنمایی کروموزوم ها از روشهایی غیر از اعداد صحیح و یا رشته های عددی استفاده کرده باشیم، عملیات بازترکیبی را به روشهای دیگری پیاده سازی می کنیم. بعنوان مثال اگر از اعداد حقیقی برای ارائه مدل مساله استفاده کرده باشیم، یک روش اینست که قسمت حقیقی و مانتیس دو عدد را جابجا کنیم.

برای سایر بازنمایی ها نیز روشهای مختلفی برای بازترکیبی ارائه شده است .

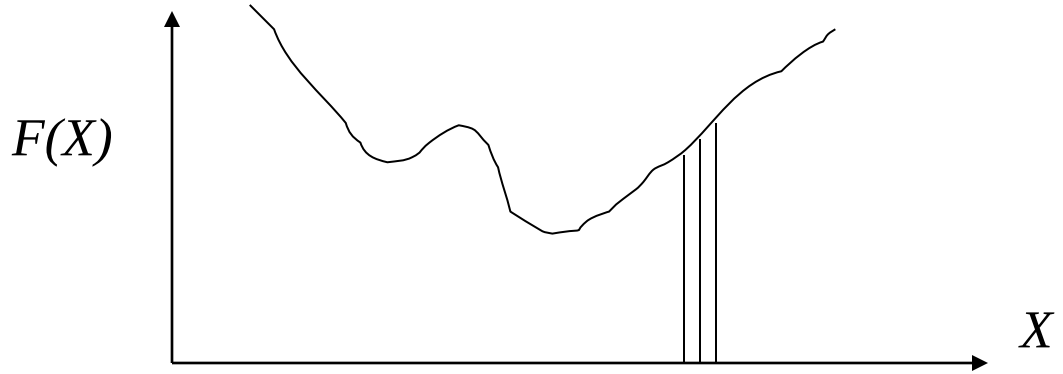
فرضیه فضای در جستجو نحوه

- روش جستجوی GA با روشهای دیگر مثل شبکه‌های عصبی تفاوت دارد:
- در شبکه عصبی روش *Gradient descent* به صورت هموار از فرضیه‌ای به فرضیه مشابه دیگری حرکت می‌کند، در حالیکه GA ممکن است به صورت ناگهانی فرضیه والد را با فرزندی جایگزین نماید که تفاوت اساسی با والد آن داشته باشد. از این رو احتمال گیر افتادن در GA مینیمم محلی کاهش می‌یابد.
- با این وجود GA با مشکل دیگری روبروست که *Crowding* نامیده می‌شود.

Crowding

- *Crowding* پدیده‌ای است که در آن عضوی که سازگاری بسیار بیشتری از بقیه افراد جمعیت دارد به طور مرتب تولید نسل کرده و با تولید اعضای مشابه درصد عمده‌ای از جمعیت را اشغال می‌کند.

- این کار باعث کاهش پراکندگی جمعیت شده و سرعت *GA* را کم می‌کند.



راه‌حل رفع مشکل Crowding

- استفاده از *Ranking* برای انتخاب نمونه‌ها: با اختصاص رتبه به فرضیه‌ها از مقدار مطلق تابع تناسب صرف نظر می‌شود، که این باعث می‌شود نسبت به رولت ویل سهم فرضیه‌های با تطابق بالاتر کاهش یابد.
- *Fitness sharing*: مقدار *Fitness* یک عضو در صورتیکه اعضا مشابهی در جمعیت وجود داشته باشند، کاهش می‌یابد.
- محدود کردن اعضائی که می‌توانند با هم ترکیب شوند: برای مثال اعضا به صورت مکانی توزیع شده و فقط به اعضا شبیه به هم امکان تولید نسل داده می‌شود. این کار به ایجاد گروههائی از اعضا مشابه در داخل جمعیت منجر خواهد شد.

عملگر های الگوریتم ژنتیک – جهش

- جهش (mutation)
  - ویژگی تصادفی بودن و امکان فرار از نقاط بهینه محلی را فراهم می‌آورد.

برای انجام جهش به این صورت عمل می کنیم:

بصورت تصادفی تعدادی از کروموزوم های فرزند را انتخاب می کنیم به صورت تصادفی مقادیر یک یا چند ژن وی را تغییر می دهیم.

همچنین در هنگام پیاده سازی به صورت زیر عمل می کنیم:

به ازای هر کروموزوم اعمال زیر را انجام می دهیم:

یک عدد تصادفی بین صفر و یک تولید می کنیم

اگر عدد تولید شده کوچکتر از P_m بود، به ازای هر ژن اعمال زیر را انجام می دهیم، در غیر اینصورت از جهش دادن کروموزوم صرف نظر می کنیم

یک عدد تصادفی بین صفر و یک تولید می کنیم

اگر عدد تولید شده کوچکتر از P_g بود، ژن مربوطه را جهش می دهیم

عملگر های الگوریتم ژنتیک

بعنوان مثال جهش برای کروموزوم های به فرم باینری به صورت زیر می باشد:

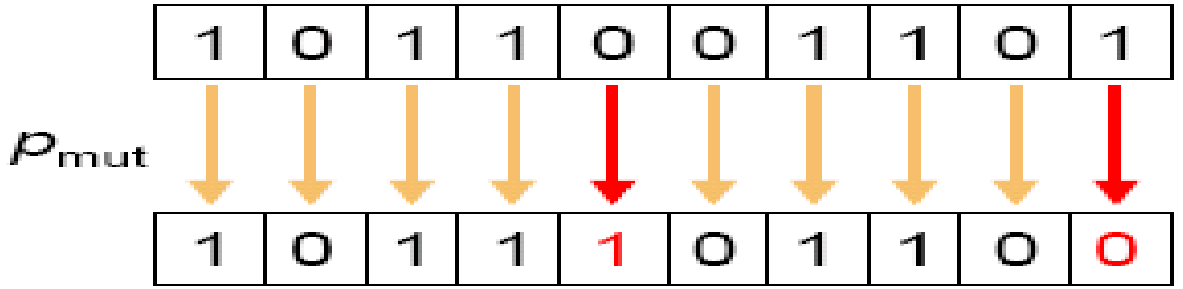
والد	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰
فرزند	۰	۰	۱	۰	۰	۱	۰	۰	۰	۱	۱	۰	۰

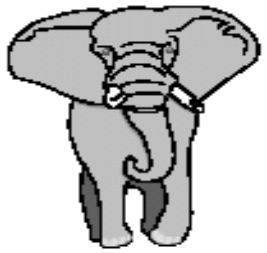
نحوه انجام عملیات جهش

* جهش، برای بازنمایی های که از مقادیر حقیقی استفاده کرده اند، به این صورت پیاده سازی می شود که یک عدد حقیقی بصورت تصادفی در یک محدوده خاص تعیین و جایگزین عدد قبلی می گردد و یا اینکه عدد اصلی با یک مقدار خاص جمع گردد و .... برای سایر مدل های بازنمایی مساله نیز انواع خاصی از جهش پیشنهاد شده است.

یک مثال

جهش بیتی (Bitwise Mutation)





نکاتی در مورد عمل جهش

جهش ابتکاری(Heuristic) :

با توجه به اینکه هدف از جهش بهتر شدن کروموزوم ها و یا اینکه پیدا شدن یک راه حل جدید است، می توانیم به جای تغییر تصادفی کروموزوم ها، تغییرات کروموزوم ها را هدفمند کنیم. برای اینکار، بسته به نوع مساله، بر روی کروموزوم انتخاب شده یکی از روشهای کلاسیک حل مساله را اعمال کرده، و جواب حاصل را بعنوان کروموزوم جدید جایگزین می کنند. استفاده از این روش که از آن با عنوان جهش ابتکاری یاد می شود، بسته به نوع مساله ممکن است دستیابی به راه حل نهایی را سریعتر کنند.

* جهش باعث ایجاد تغییرات ناخواسته در جمعیت شده، و باعث بوجود آمدن موجودات جدید می شود. در واقع برتری جهش نسبت به بازترکیبی نیز همین مطلب می باشد.در صورتی که فقط از جهش استفاده کنیم، ممکن است که بتوانیم جواب بهینه را پیدا کنیم، اما استفاده از بازترکیبی به تنهایی پیدا شدن جواب بهینه را تضمین نمی کند.

شرط خاتمه الگوریتم

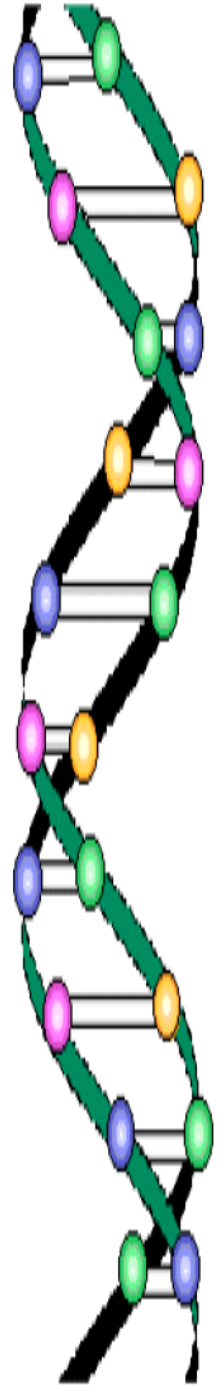
چون که الگوریتم های ژنتیک بر پایه تولید و تست می باشند، جواب مساله مشخص نیست و نمی دانیم که کدامیک از جواب های تولید شده جواب بهینه است تا شرط خاتمه را پیدا شدن جواب در جمعیت تعریف کنیم. به همین دلیل، معیارهای دیگری را برای شرط خاتمه در نظر می گیریم:

1. تعداد مشخصی نسل: می توانیم شرط خاتمه را مثلاً ۱۰۰ دور چرخش حلقه اصلی برنامه قرار دهیم.
2. عدم بهبود در بهترین شایستگی جمعیت در طی چند نسل متوالی
3. واریانس شایستگی جمعیت از یک مقدار مشخصی پائین تر بیاید و یا اینکه در طی چند نسل متوالی مشخص، تغییر نکند.
4. بهترین شایستگی جمعیت از یک حد خاصی کمتر شود.

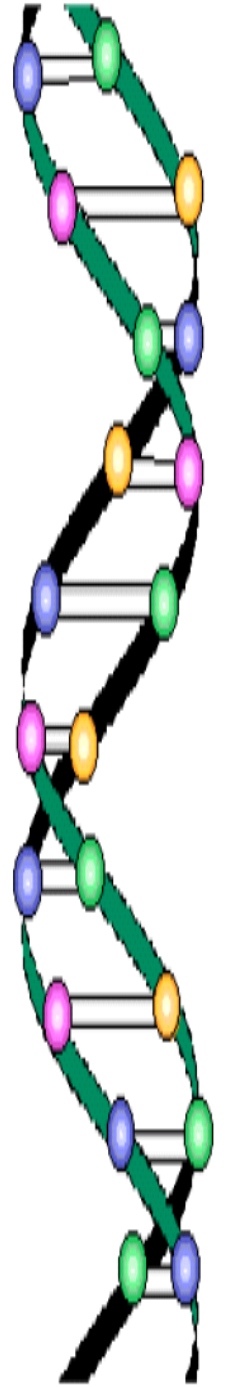
✓شرایط دیگری نیز می توانیم تعریف کنیم و همچنین می توانیم ترکیبی از موارد فوق را به عنوان شرط خاتمه به کار ببندیم.

پارامترهای یک GA

- ✓ طول هرکروموزوم (Strings Length)
- ✓ اندازۀ جمعیت (Population Size)
- ✓ احتمال کروسور (Crossover Probability)
- ✓ احتمال جهش (Mutation Probability)



مثالهایی از بهینه سازی



در این مثالها

- نوع رمزگذاری : دودویی
- تابع رمزگذار : $C_{n,X}$
- تابع رمزگشای : $\tilde{C}_{n,X}$

(برای مثال اول X=[-1,1] و برای مثال دوم X=[-10,10]*[-10,10])

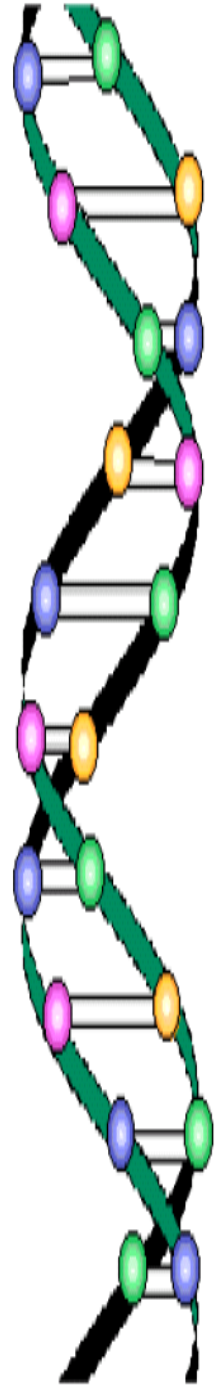
- جمعیت اولیه بر حسب *کروموزوم ها* بصورت *تصادفی* تحت *توزیع برنولی* تولید می شود
  - عملگر انتخاب : چرخ رولت
  - عملگر کروسور : نقطه منفرد
  - عملگر جهش : جهش بیتی

مثالهایی از بهینه سازی

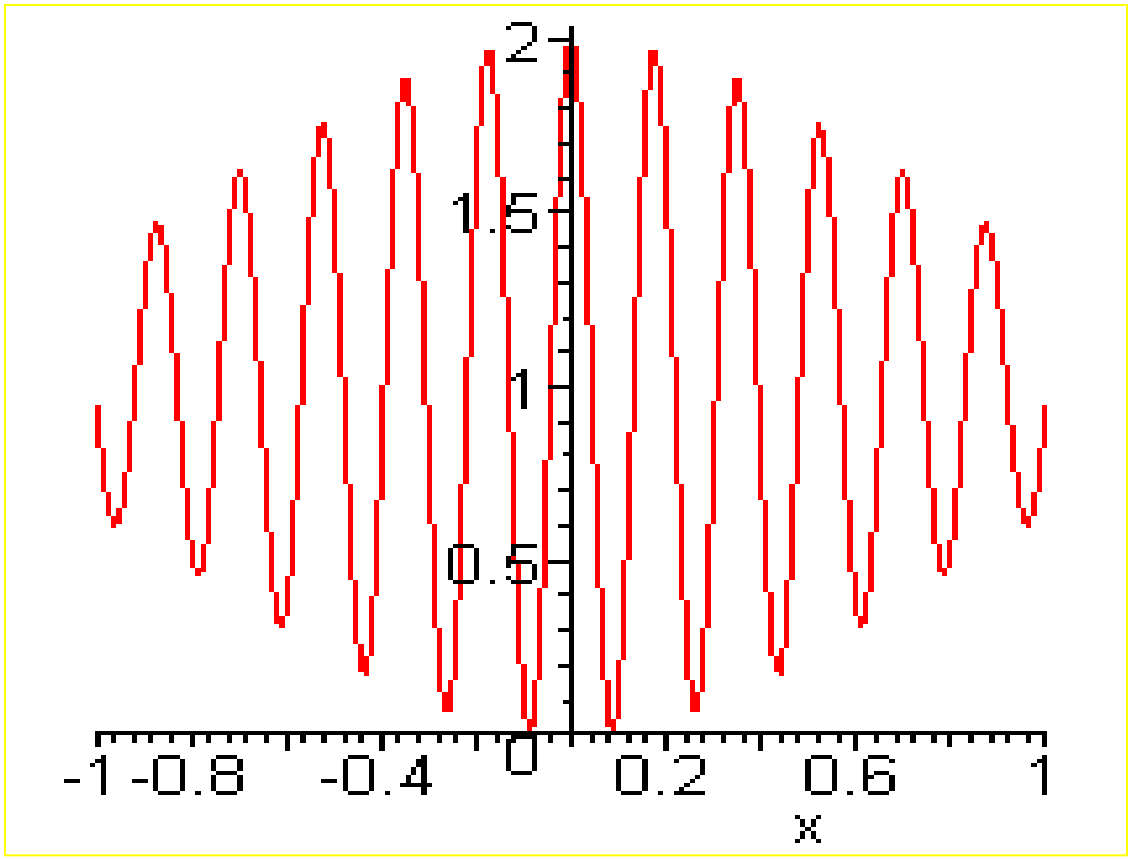
- مثال اول. هدف یافتن ماکزیمم سراسری تابع ذیل است

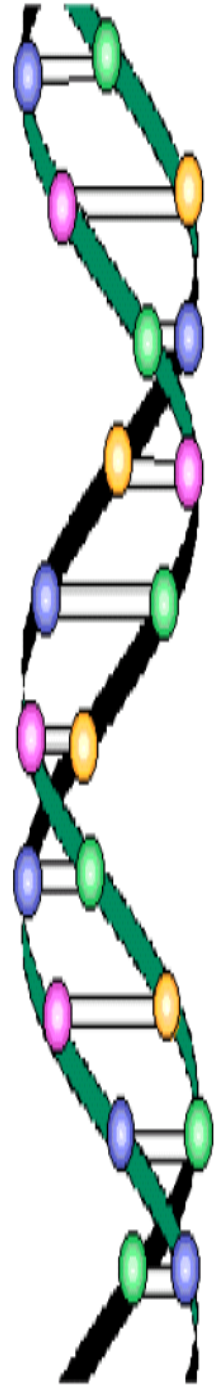
$$f_1 : [-1,1] \rightarrow R$$

$$x \rightarrow 1 + e^{-x^2} \cos(36x)$$



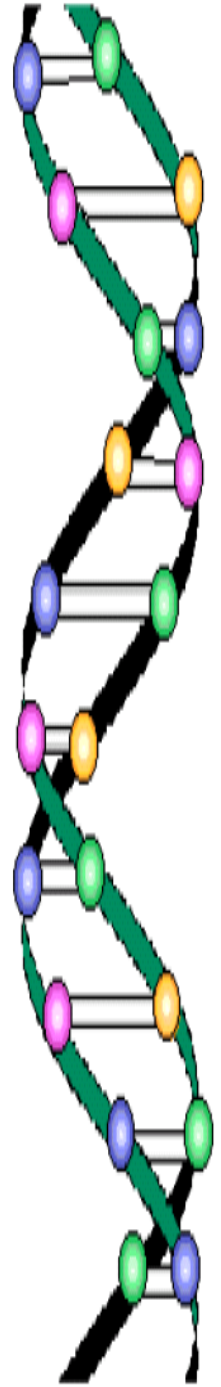
نمودار تابع در مثال اول





برای مثال اولGAپارامترهای

- طول هرکروموزوم : ۲۴
- اندازۀ جمعیت : ۶
- احتمال کروسور : ۱
- احتمال جهش : ۰.۰۴۲
- نخبه گرایی : فعال



نتایج مثال اول

جمعیت اولیه(لیست کروموزوم ها)

0 0 0 1 1 0 0 0 1 0 0 1 1 0 0 1 1 0 1 0 0 0 1 1

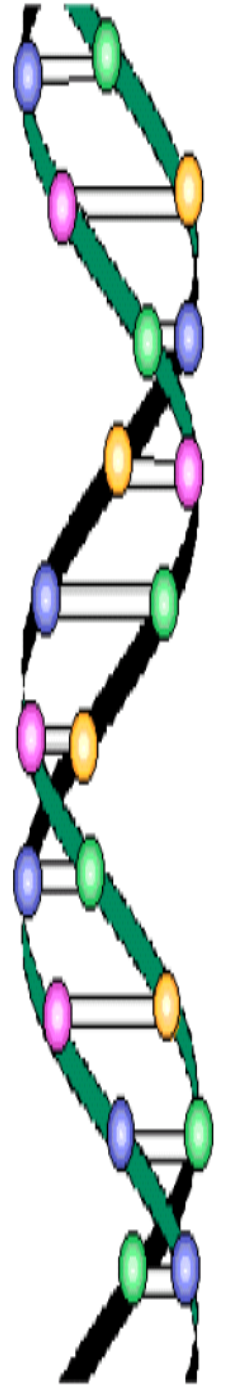
0 1 1 1 0 0 1 0 1 0 1 0 1 0 1 1 0 0 0 0 0 0 0 0

1 1 0 0 1 0 1 0 0 0 1 1 0 0 0 1 0 0 0 1 0 0 1 0

0 0 0 0 0 1 1 0 1 1 0 0 0 0 1 1 0 0 1 1 1 1 0 0

0 0 0 1 0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0

1 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 1 1 1 1



نتایج مثال اول

نسل پنجاه و دوم(لیست کروموزوم ها)

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0

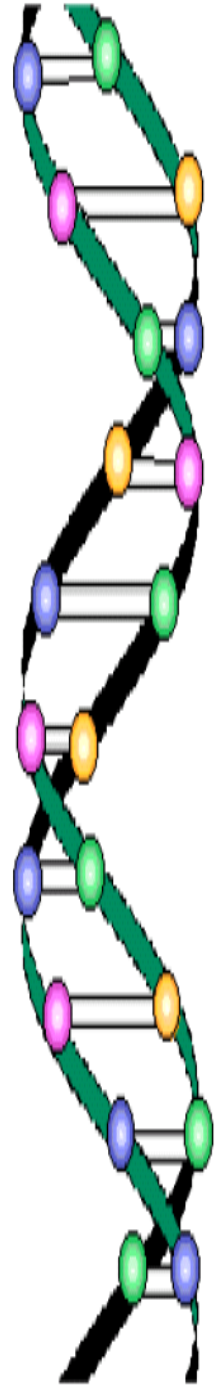
0 1 1 1 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0

0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0

0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0

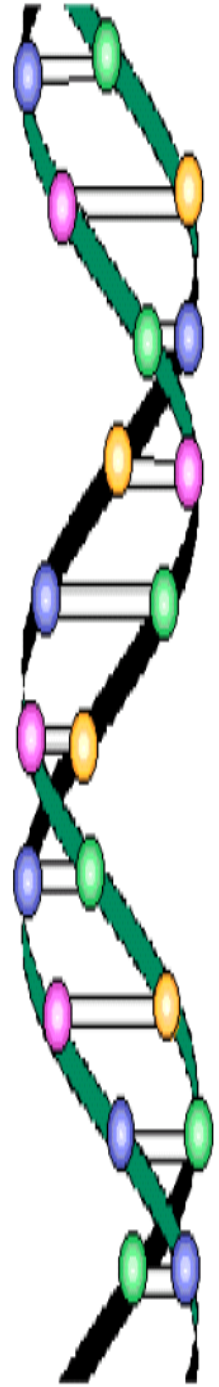
0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0 1 1 0 0

0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0



جمعیت پنجاه و دوم با مقدار مطلوبیت تابع و جواب امکانی متناظر برای مثال اول

$f(x)$	x
9.5292005622E-01	9.9999964221E-01
1.8433600878E+00	-1.5747250144E-02
1.9999999896E+00	-3.9935912355E-06
1.9999999999E+00	-4.1731254896E-07
1.5155558320E+00	-5.0006338958E-01
2.0000000000E+00	-1.7889396986E-07



موفق و پیروز باشید
روانشادنیا

www.irancem.com
www.ravanshadnia.ir