

الگوریتم ژنتیک در مسیریابی وسیله نقلیه

فهرست مطالب

- معرفي الگوریتم ژنتیک و مفاهیم اولیه آن
- بررسی مسئله مسیریابی وسیله نقلیه با الگوریتم ژنتیک

- محققان با مشاهده خصوصیات سیستم های طبیعی و تامل در این که طبیعت چگونه مشکلات خود را حل می کند به فکر تقلید از روش های طبیعی در حل مسایل پیچیده و طراحی سیستمها افتادند.
- ایده اصلی الگوریتم ژنتیک بر پایه نظریه تکامل داروین در سال 1859 شکل گرفت.



- در سال 1975 دانشمندی در دانشگاه میشیگان به نام John Holland ایده استفاده از الگوریتم ژنتیک را در بهینه سازی مهندسی مطرح کرد.



مفاهیم اولیه

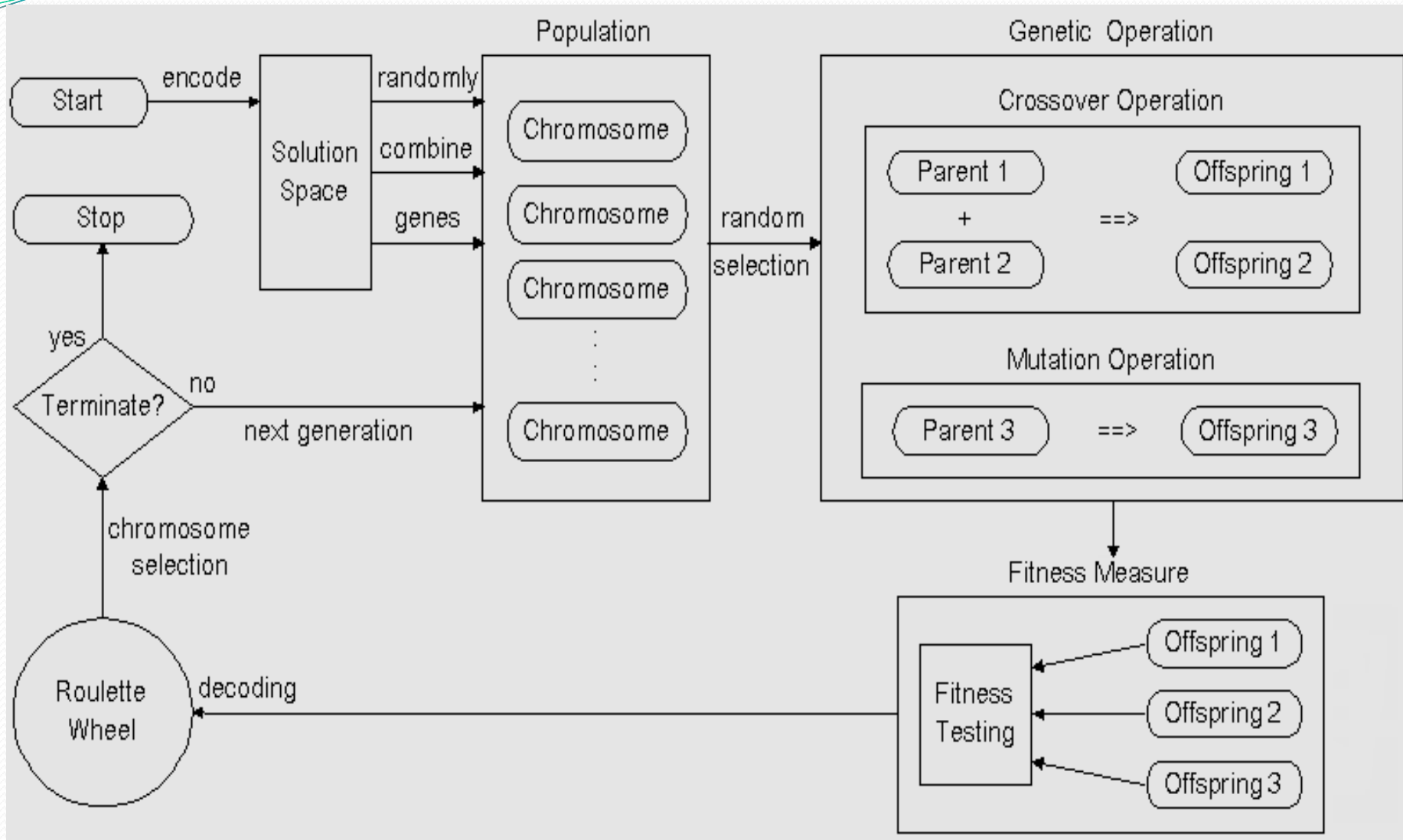
- ژن: عناصر تشکیل دهنده يك کروموزوم که معمولا عدد یا کلمه یا حتی دستورات برنامه نویسی می باشد. ژن یکی از متغیرهای تصمیم یا یک قسمتی از متغیرهای تصمیم است.
- کروموزوم: به رمز درآمده یا کد شده يك جواب یا قسمتی از جواب را کروموزوم می گویند. به عنوان مثال، يك کروموزوم با n ژن می تواند به شکل زیر باشد:

$$b1 \ b2 \ b3 \ ... \ bn$$

- مکان: موقعیت هر ژن در کروموزوم
- جامعه (population): به مجموعه ای از کروموزوم ها که از بین آنها selection صورت می گیرد جامعه گفته می شود.
- نسل (generation): هر جایگزینی از جامعه قدیمی با جامعه جدید (هر تکرار الگوریتم)

مراحل الگوریتم ژنتیک

1. تعیین روش کدگذاری (encoding) توسط برنامه نویس
2. تولید جامعه اولیه به اندازه N
3. محاسبه ی تابع برازندگی (fitness function) هر کروموزوم در جامعه
4. ایجاد جامعه جدید (عمل تولید نسل ((reproduction): N بار قدم های زیر را انجام دهید:
 - انتخاب دو کروموزوم والد (یا بیشتر) از جامعه فعلی با استفاده از عملگر انتخاب (selection)
 - اعمال عملگر تقاطع (crossover) روی والدین با احتمال وقوع pc (probability of crossover) و ایجاد فرزند
 - اعمال عملگر جهش (mutation) روی فرزندان با احتمال وقوع pm (probability of mutation)
 - اضافه کردن فرزند جدید به جامعه
5. انتخاب جامعه جدید و جایگذاری جامعه جدید با جامعه فعلی با استفاده از عملگر انتخاب
6. آزمودن شرایط توقف و برگرداندن بهترین جواب در صورت توقف
7. بازگشت به قدم 3



شمای کلی از الگوریتم ژنتیک

رایج ترین حالت الگوریتم ژنتیک

ایجاد جمعیت توسط گروهی از کروموزوم ها

ارزیابی کروموزوم ها

انتخاب دو کروموزوم بر اساس مقدار برازندگی

تولیدمثل یک فرزند یا بیشتر

جهش دادن فرزندان به صورت تصادفی

تکرار این مراحل تا رسیدن به معیار توقف (بسته به خواست برنامه نویس)

کد گذاری Encoding

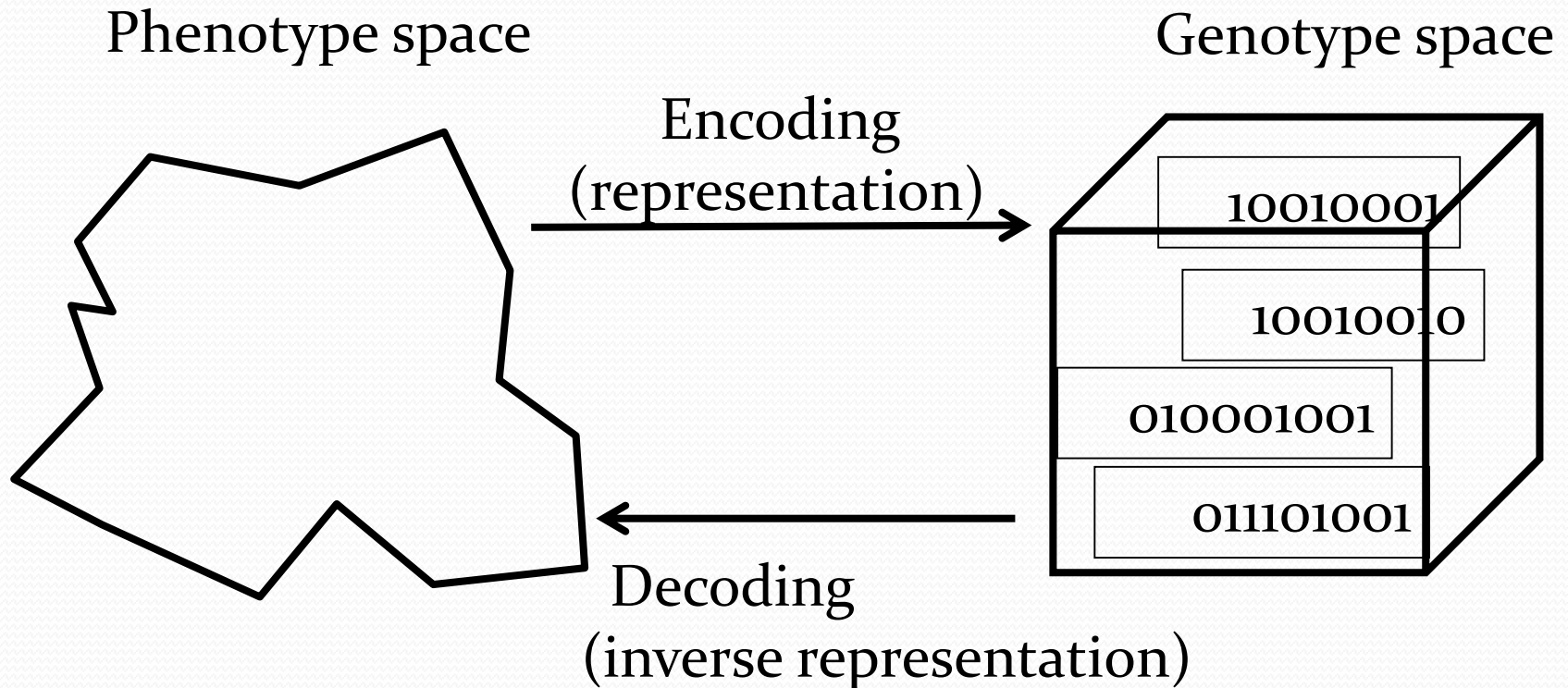
الگوریتم ژنتیک به جاي کار بر روي متغیرهاي مساله، با کد شده جواب ها يعني کروموزوم ها سر و کار دارد. ساير گام ها شديدا تحت تاثير گام کدگذاری مي باشند.

انواع روشهای نشان دادن کروموزوم ها:

- (1) رشته های بیتی (باینری) (bit strings) ← 0101...1100
- (2) اعداد حقیقی (real numbers) ← 43.2 -33.1 ... 0.0 89.2
- (3) جایگشت های المان ها (permutations of element) ←

E11 E3 ... E1 E15

مفهوم Representation (تبدیل متغیرها به کروموزوم ها)



روش های ایجاد جامعه اولیه

- روند کاملاً تصادفی برای ایجاد جامعه اولیه
- روش های ابتکاری

عملگرهای الگوریتم ژنتیک (GA operators):

1. عملگر انتخاب (selection): انتخاب کروموزوم های والد برای ادغام
2. عملگر تقاطع (crossover): ادغام کروموزوم های انتخاب شده
3. عملگر جهش (mutation): تغییر ژنتیکی فرزندان

عملگر انتخاب (selection)

- مطابق نظریه تکاملی داروین، جامعه بیشتر به داشتن موجوداتی تمایل دارد که با تغییر کروموزوم های جامعه، ساختار و رفتار آن تکامل یابد.
- در الگوریتم ژنتیک نیز وظیفه ی اصلی عملگر انتخاب، هدایت الگوریتم به نواحی امیدبخش فضای جواب است.
- چون در الگوریتم ژنتیک اندازه جمعیت ثابت است عملگر انتخاب با شناسایی جواب های خوب، ساختن کپی از آنها و حذف جوابهای بد، علاوه بر حفظ اندازه جمعیت، باعث زیاد شدن جوابهای خوب در جمعیت نیز می شود.

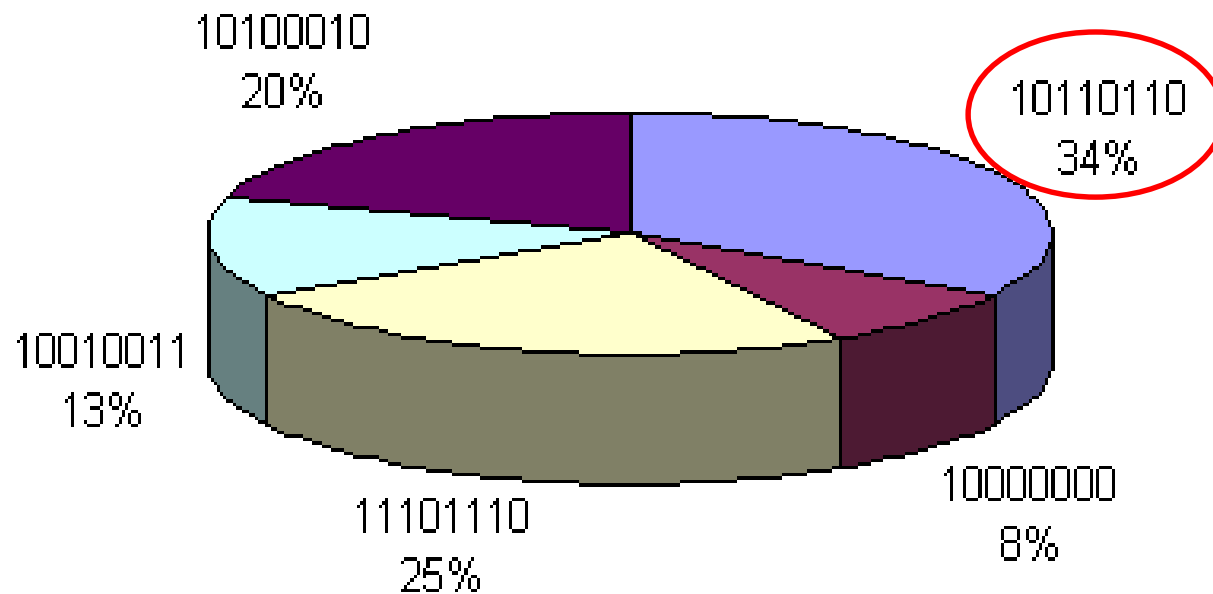
برخی از انواع عملگرهای انتخاب

I. انتخاب برترین ها (Elitist): مناسب ترین کروموزوم(های) هر جامعه انتخاب می شود.

II. انتخاب چرخ رولت (Roulette wheel): کروموزومی که مقدار تابع برازندگی بیشتری داشته باشد شانس بیشتری برای انتخاب شدن دارد.

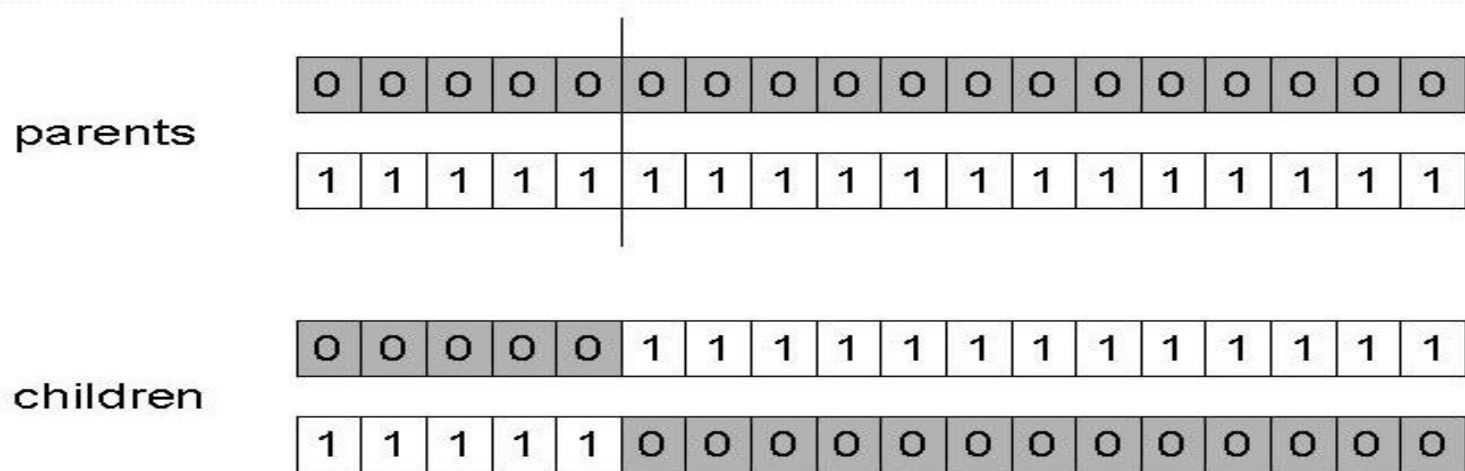
III. انتخاب رقابتی (Tournament): زیر مجموعه هایی از یک جامعه انتخاب می شود و اعضای آن زیر مجموعه ها با هم رقابت می کنند و سرانجام فقط یک عضو از هر زیر گروه برای تولید انتخاب می شود.

Chromosome Fitness on a Roulette Wheel

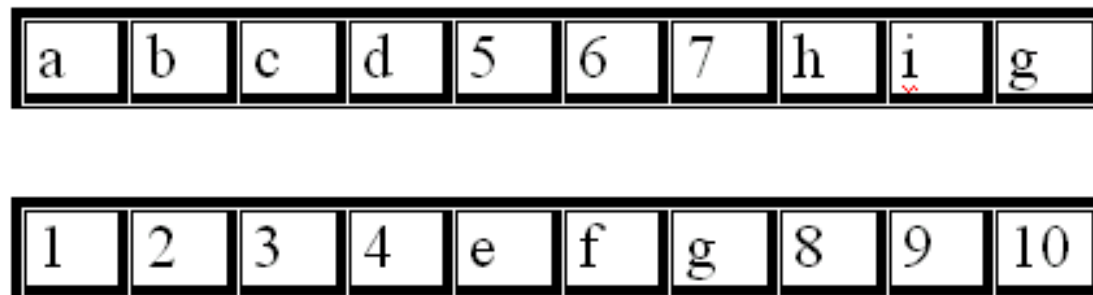
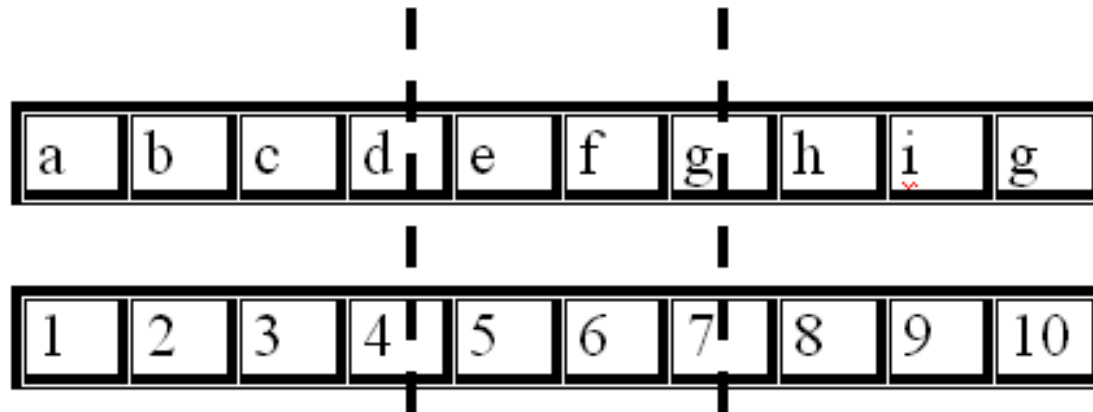


عملگر تقاطع يك نقطه برش 1-point crossover

- انتخاب تصادفي يا غير تصادفي يك نقطه روي کروموزوم والدين
- شکستن والدين از اين نقطه
- فرزندان، سر را از يك والد و دم را از ديگري به ارث مي برند.



عملگر تقاطع با دو نقطه برش 2-point crossover



فرزندان، تنه را از یک والد و سر و دم را از والد دیگر می گیرند.

تقاطع مطابقت جزئی (PMX) Partially Matched Crossover

- جهت حل مساله فروشنده دوره گرد با کدگذاری ترتیبي مناسب است.

اگر دو رشته مقابل را داشته باشیم :

8521|364|7 و 1234|567|8

و یک تقاطع دو نقطه ای انجام دهیم خواهیم داشت:

$V1=1234|364|8$ و $V2=8521|567|7$

که بطور بديهی غیر مجاز (not legal) هستند چون $V1$ شهر 5 یا 7 را بازدید

نمی کند و شهرهای 3 و 4 را دوبار بازدید می کند. بطور مشابه $V2$ شهرهای 3

و 4 را نمی بیند و شهرهای 5 و 7 را دوبار می بیند.

ادامه تقاطع مطابقت جزئی

8521|364|7 و 1234|567|8

با استفاده از PMX رشته های زیر ساخته می شوند:

8 3 2 1 5 6 7 4 و 1 2 5 7 3 6 4 8

ژن های 3 و 4 و 5 و 6 و 7 بین دو نقطه تقاطع واقع شده اند.

انتخاب دوتایی ژن ها بر مبنای صعودی بودن شماره و عدم واقع شدن هر دو ژن در یکی از سه منطقه برش داده شده است.

جهش

- هر فرزند ممکن است دچار جهش ژنتیکی شود.
- جهش ژنتیکی با احتمال مساوي روي تمام ژن هاي هر کروموزوم نوزاد رخ مي دهد.
- احتمال وقوع جهش روي يك ژن را نرخ جهش نامند.
- نرخ جهش را با pm نشان مي دهند كه در بازه $[0.01, 0.2]$ قرار مي گيرد.

parent	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
child	0	1	0	0	1	0	1	1	0	0	0	1	0	1	1	0	0	1

معیارهای مختلف توقف الگوریتم ژنتیک

- 1 – بهترین جواب را بعد از اجرای تعداد مشخصی تکرار تغییر ندهد .
 - 2 – میانگین برازندگی جوابهای موجود در جمعیت جاری همان برازندگی بهترین جواب یا بسیار نزدیک به آن باشد.
 - 3 – می توانیم از پیش قرارداد کنیم که الگوریتم به تعداد بار مشخصی اجرا شود.
- نکته: معمولا روتین های توقف مختلف است و بستگی به پیچیدگی و چگونگی مساله دارد.

نقاط قوت الگوریتم ژنتیک

- موازی بودن:

از آنجایی که الگوریتم ژنتیک **چندین نقطه شروع** دارد، در یک لحظه می تواند فضایی مسئله را از چند جهت مختلف جستجو کند. اگر یکی به نتیجه نرسید سایر راه ها را ادامه می دهیم.

- الگوریتم های دقیق در حل مسایل بزرگ بسیار کند عمل می کنند.

- در مسایل NP-hard گاه نمی توان از راه های دقیق به جواب بهینه رسید.

نقاط ضعف الگوریتم ژنتیک

- چگونگی نوشتن تابع برازندگی

تابع برازندگی باید منجر به بهترین راه حل برای مسئله شود.

برای انتخاب تابع مناسب برای برازندگی پارامترهای زیر دخیلند:

(1) اندازه جامعه

(2) نرخ جهش و تقاطع

(3) قدرت و نوع انتخاب

نارس بودن

- اگر یک کروموزوم فاصله اش با سایر کروموزوم های نسل اش زیاد باشد (خیلی بهتر از بقیه باشد) و خیلی زود دیده شود (ایجاد شود) ممکن است محدودیت ایجاد کند و راه حل را به سوي جواب بهینه محلي سوق دهد.
- این اتفاق معمولاً در جمعیت های کم اتفاق می افتد.
- روش انتخاب رقابتي بر این مشکل غلبه می کند.

الگوریتم ژنتیک در مسیریابی وسیله نقلیه

- الگوریتم ژنتیک به عنوان یکی از پرکاربردترین روشهای حل VRP مطرح بوده است.
- اولین کار صورت گرفته در زمینه بررسی VRP با الگوریتم GA توسط تانگیا و همکاران (1991) صورت گرفت که مسئله VRPTW را با الگوریتم GA بررسی کردند. مدل ارائه شده آنها GIDEON نام داشت.
- برگر و همکاران (2003)، GA را برای حل VRPTW به کار بردند. این الگوریتم در مقایسه با بهترین حالت الگوریتم Tabu search قابل رقابت است.

- حالتی از الگوریتم ژنتیک که در VRP کاربرد دارد الگوریتم ژنتیک مبتنی بر ترتیب است که تمامی کروموزوم ها، جایگشت یک لیست هستند؛ یعنی جوابها به صورت توالی کدبندی می شوند.
- برخی از مسائل بررسی شده با الگوریتم ژنتیک مبتنی بر ترتیب:

i. VRPTW

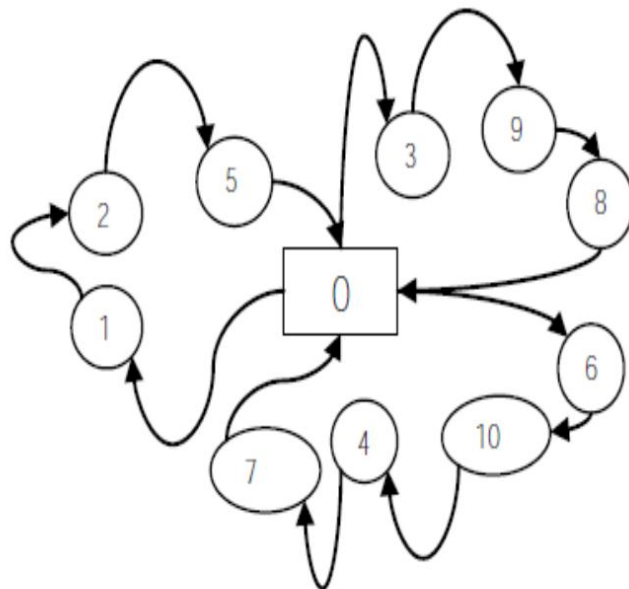
ii. TSP

iii. Bin Packing

iv. مسائل جایابی

یک کروموزوم (جواب) در VRP

- تعداد وسایل نقلیه
- مشتریانی که هر مسیر شامل می شود
- ترتیب ملاقات برای هر مسیر را مشخص کند.



Route 1

6	10	4	7
---	----	---	---

Route 2

1	2	5
---	---	---

Route 3

3	9	8
---	---	---

یک مثال برای تابع برازندگی

$$Eval_i = d_i / \max(d_m, d_i)$$

d_i فاصله سفر کرده کل کروموزوم i
 d_m میانگین فاصله سفر کرده کل کروموزوم های تشکیل دهنده جامعه اولیه
در اینجا مقدار برازندگی هرچه کمتر باشد بهتر است.

عملگرهای تقاطع سنتی برای الگوریتم ژنتیک مبتنی بر ترتیب

ویتلی و همکاران (1990) و استارکودر و همکاران (1991)

- ترکیب مجدد یال Edge Recombination

- تقاطع ترتیب 1 Order Crossover 1

- تقاطع ترتیب 2 Order Crossover 2

- PMX

- تقاطع حلقه Cycle Crossover

- تقاطع موقعیت Position Crossover

تقاطع ترتیب 1

Chromosome A: 2 5 6 1 0 7 3 8 4 9

Chromosome B: 4 1 6 9 3 8 2 0 5 7

Early Time Ordering: 0 1 2 3 4 5 6 7 8 9
(Global Precedence Vector)

Chromosome A: 2 5 6 1 0 7 3 8 4 9

Chromosome B: 2 1 6 9 3 8 4 0 5 7

Child AB: 2 X X X X X X X X X

Chromosome A: 2 1 6 5 0 7 3 8 4 9

Chromosome B: 2 1 6 9 3 8 4 0 5 7

Child AB: 2 1 X X X X X X X X

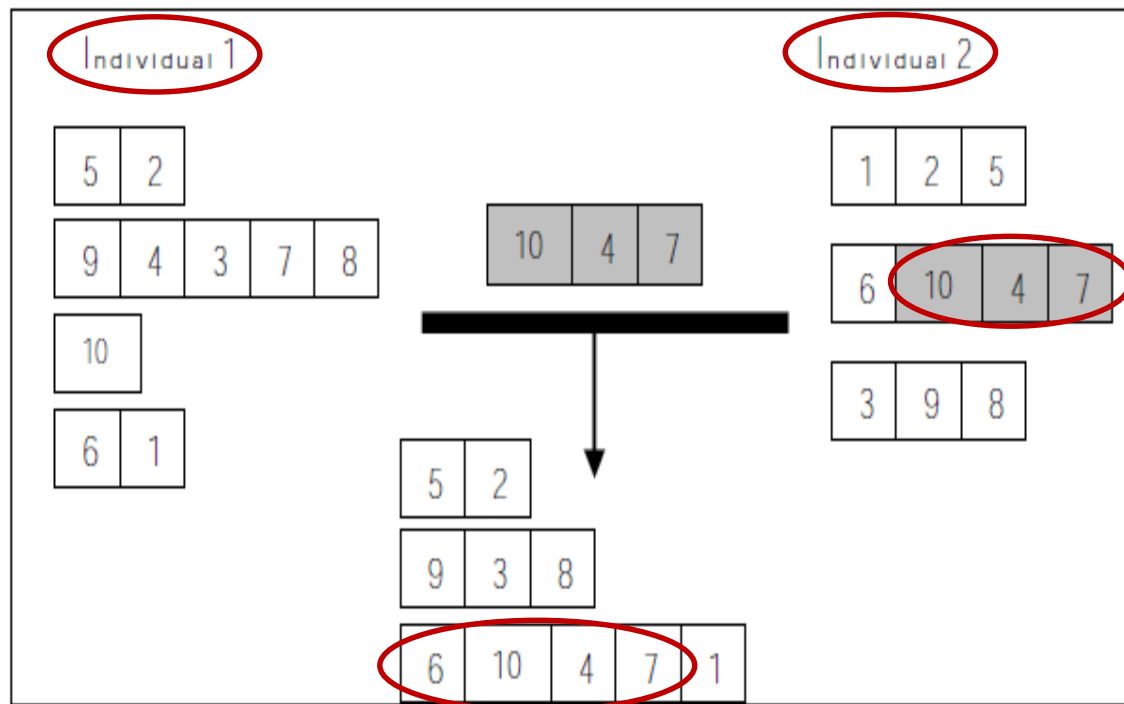
Child AB: 2 1 6 5 0 7 3 4 8 9

یک الگوریتم عملگر تقاطع جدید

- برای هر فرد I_1 از مجموعه انتخاب شده S تکرار کنید:
- به صورت تصادفی فرد دیگری از S (I_2) انتخاب کنید.
- از I_2 به صورت تصادفی، زیرمسیر SR را انتخاب کنید: $\{a_1, a_2, \dots, a_n\}$
- مشتری c را که به SR متعلق نیست و از لحاظ جغرافیایی به a_1 نزدیکتر است بیابید.
- SR را در I_1 به گونه ای جای دهید که a_1 درست بعد c قرار گیرد.
- از I_1 اصلی تمامی مشتریان تکراری را که در SR نیز ظاهر شده اند حذف کنید که نتیجتاً فرزند D به عمل می آید.

ادامه عملگر تقاطع جدید

- فرض کنید مشتری 6 نزدیکترین مشتری به مشتری 10 است. به این جهت $\{10, 4, 7\}$ جدا می شود و پس از مشتری 6 وارد یکی از مسیرهایی که به I1 تعلق دارد می شود.



ویژگی های عملگر تقاطع در VRP

- قسمتی از اطلاعات والدین را به فرزندان منتقل می کند. توجه شود که این فرایند تعداد وسایل نقلیه را زیاد نمی کند بلکه ممکن است چندین مسیر را حذف کند.

یا

- نه تنها مشتری های یک مسیر را با مسیر دیگر معاوضه می کنند بلکه تعداد وسایل نقلیه را هم بهبود می دهد (بهینه می کند).

عملگر جهش

- فرزندان به عمل آمده از تقاطع می توانند مورد عمل جهش واقع شوند. 4 عملگر جهش مطرح است:

(1) وارون سازی (inversion)

(2) جابجایی (displacement)

(3) درج (insertion)

(4) معاوضه (swap)

تمامی عملگرهای جهش ذکر شده، احتمال کاربرد خاص خود را دارند.

- وارون سازی یک زیرمسیر در نظر می گیرد و ترتیب مشتریان ملاقات شده ای که به آن زیرمسیر تعلق دارند را وارون می کند.

$$12345 \rightarrow 12543$$

- جابجایی یک زیرمسیر انتخاب می کند و آنرا به مکان دیگری که به صورت درونی (intra) یا بینی (inter) می تواند جابجا کند وارد می کند (قسمت انتخاب شده می تواند هم در همان مسیر و هم در مسیر دیگری وارد شود).

$$1234567 \rightarrow 1263457$$

- عملگر درج یک مشتری انتخاب می کند و آنرا به مکان دیگری وارد می کند. مسیر درج شونده به صورت تصادفی انتخاب می شود و نیز امکان دارد این مشتری واحد یک مسیر جدید ایجاد کند.

$$\begin{array}{cc} 12345 & \xrightarrow{\quad} & 1245 \\ 6789 & & 67389 \end{array} \quad \text{or} \quad \begin{array}{c} 1245 \\ 3 \end{array}$$

- عملگر معاوضه دو مشتری انتخاب می کند و جای آنها را با هم تعویض می کند. این دو مشتری می توانند متعلق به یک مسیر یا مسیرهای متفاوت باشند. همانند عملگر قبلی ممکن است مسیر با توالی جدیدی ایجاد شود (احتمال این رخداد به همانگونه محاسبه می شود).

$$\begin{array}{cc} 01234 & \xrightarrow{\quad} & 01734 \\ 56789 & & 56289 \end{array}$$

معاوضه و وارون سازی توانایی حذف و اضافه کردن تعداد وسایل نقلیه برای یک کروموزوم را ندارند. از سوی دیگر، درج و جابجایی قادر به بهبود مسیرها هستند.

Comparison of CPU times with published results

Cities	Vehicles	Osman ^a (SA)	Osman ^b (TS)	Gendreau et al. ^c (TABUROUTE)	Pure ^d GA	Hybrid ^e GA
50	5	167.4	114.0/67.2	360	213	23
75	10	6434.3	178.7/70.8	3228	765	617
100	8	9334.0	1543.0/675.0	1104	1148	717
150	12	5012.3	3560.0/3075.0	3528	2475	1961
199	17	2318.1	3246.0/1972.7	5454	3999	5261
50	6	3410.2	173.0/140.2	810	217	429
75	11	626.5	1056.7/203.0	3276	786	449
100	9	957.2	2998.0/1200.0	1536	1134	1904
150	14	84301.2	4755.8/2443.6	4260	2258	2242
199	18	5708.0	4561.0/3310.1	5988	3687	6433
120	7	315.8	1445.4/1398.4	1332	1633	1483
100	10	632.0	892.2/407.5	960	1160	1285
120	11	7622.5	2834.0/1343.0	3552	1694	1063
100	11	305.2	1175.9/5579.0	3942	1197	585
Average		9081.8	2038.1/1563.3	2809.3	1597.6	1746.6

^aSimulated annealing by Osman, seconds using VAX 8600 (listed in [33] as 0.48 Mflop/s).

^bTabu search + first-best-admissible/best-admissible by Osman, seconds using VAX 8600 (listed in [32] as 0.48 Mflop/s).

^cTabu search by Gendreau et al., seconds using Silicon Graphics work station (5.7 Mflop/s).

^dPure genetic algorithm, seconds using Pentium 266 MHz (83.7 Mflop/s).

^eGenetic algorithm hybridised with neighbourhood search, seconds using Pentium 266 MHz (83.7 Mflop/s).