**استفاده از الگوریتم ژنتیک برای حل یک معادله ساده ریاضی**

دنی هرماوانتو

موسسه علوم اندونزی

ایمیل: denny.hermawanto@gmail.com

چكيده

**این مطلب استفاده از الگوریتم ژنتیک را به شکلی ساده به همراه مثال برای افراد مبتدی شرح میدهد. ابتدا فلسفه اساسی و فلوچارت این الگوریتم شرح داده می شود و در مرحله بعد محاسبات قدم به قدم الگوریتم ژنتیک برای حل یک مسائله ساده تساوی ریاضیات بصورت کامل شرح داده می شود.**

**فلسفه پایه الگوریتم های ژنتیک**

توسعه و گسترش یافت که الهام گرفته از تئوری تکامل داروین است. مبنای این نظریه هم بقای گونه ها بر اساس میزان ارزش انها ست هر چه قوی تر احتمل بقا هم بیشتر داروین همچنین معتقد است که بقای یک نسل میتواند بوسیله فرآیند تولید مثل انتخاب، برش و جهش حفظ شود. بعدها از ایده داروین در خصوص تکامل برای بدست آوردن راه حل بهینه در مسائل محاسباتی استفاده شد که یکی از معروف ترین این الگوریتم ها ، الگوریتم ژنتیک است.

راه حتی که توسط الگوریتم های ژنتیک تولید میشود یک کروموزوم نامیده میشود و مجموعه از کروموزم ها ، جمعیت population نامیده می شوند. کروموزم ها از ژنها تشکیل شده اند و مقدار آن میتواند عددی دودویی ، سمبل ها با کاراکترها باشد که وابسته به مسائله است که قصد حل نمودن آن را داریم. شایان ذکر است نحوه نمایش کروموزوم ها و مقادیر آنها یکی از اساسی ترین قسمتهای حل یک مسائله با استفاده از الگوریتم ژنتیک است و دیگر قسمت بسیار مهم در این الگوریتم ها تعریف تابع برازش است که در قسمت بعدی به آن پرداخته می شود.

ارزش و بهای هر کروموزم توسط تابعی تحت نام برازش اندازه گیری میشود. در واقع تابع برازش میزان مفید و مناسب بودن راه حل برای مسائله مورد نظر را اندزه گیری می نماید. پس از هر مرحله ای که الگوریتم ژنتیک تکرار شد برازش هر یک از راه حلهای تولید شده ) کروموزوم ها ) اندزه گیری میشود. برخی از کروموزم ها درون جمعیت تحت پروسه ای به نام crossover با یکدیگر ترکیب میشوند و بدین صورت کروموزم های جدیدی که فرزند نامیده میشود تولید میکنند که ترکیب ژن این کروموزم های جدید ترکیبی از ژنهای والدین آنهاست. در یک نسل تعدای از کروموزومها نیز در ژنهای خود دچار جهش mutation می شوند.

تعداد کروموزومهایی که تحت تأثیر crossover و جهش mutation قرار میگیرند بوسیله نرخ برش و نرخ جهش کنترل میشوند. کروموزومی در جمعیت که برای نسل بعدی حفظ و نگاهداری میشود بوسیله قانون تکامل داروین انتخاب می شود. کروموزمی که مقدار برازش بالاتری دارد از احتمال بالاتری برای انتخاب دوباره در نسل بعدی برخوردار است. بعد از چندین نسل، مقدار کروموزم به یک مقدار خاص که بهترین راه حل برای مسائله است همگرا می شود.

**مراحل الگوریتم ژنتیک**

در الگوریتم ژنتیک فرآیند به شرح ذیل است:

قدم 1 : تعداد کروموزوم ها ، نسلها و نرخ جهش و نرخ برش معیین می شود.

قدم ۲ بر اساس تعداد جمعیت کروموزوم - کروموزوم ایجاد میشود و ژنهای کروموزوم - با یک مقدار تصادفی مقدار دهی اولیه می شوند.

قدم ۳ : قدم های ۷۰۴ را تا زمانی که تعداد نسل ها برآورده شود ) تأمین شود) انجام می شود.

قدم ۴ : مقدار برازش کروموزوم بوسیله محاسبه تابع هدف objective function مورد ارزیابی قرار می گیرد.

قدم ۵ : انتخاب کرموزوم .

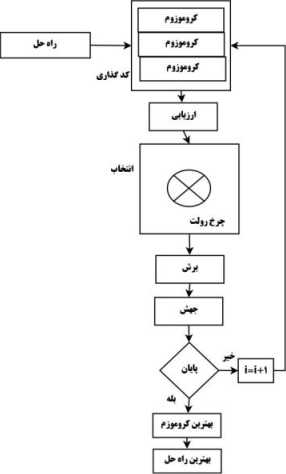
قدم ۵ : برش .

قدم ۶ جهش .

قدم کرموزم جدید (فرزند)

قدم ۸ راه حل بهترین کرموزوم ها

فلوچارت الگوریتم را میتوان در شکل زیر مشاهده نمود :



**مثال عددی**

در اینجا یک مثال از کاربردی که از الگوریتم ژنتیک برای حل نمودن یک مسائله ترکیبی استفاده میکند را توضیح می دهیم فرض کنیم که تساوی زیر را داریم

**a+db+۳c+۴d=۳۰**

از الگوریتم ژنتیک برای بدست آوردن مقادیر تا برای حل تساوی ذکر شده استفاده کنیم.

در ابتدا باید تابع هدف را فرمول کنیم، برای این مسائله هدف مینیمم نمودن مقدار تابع F(X) جایی که:

F(X) = ((a + ۲b + ۳c + ۴d ) – ۳۰)

از آنجا که چهار متغیر در تساوی وجود دارد، یعنیa,b,c,d می‌توانیم کروموزوم ها را به صورت زیر شکل دهیم:

|  |  |  |  |
| --- | --- | --- | --- |
| **D** | **c** | **b** | **۱** |

برای افزودن سرعت محاسبات، می‌توانیم مقدار متغیرها را به اعداد صحیح بین ۰ تا ۳۰ محدود نماییم.

**قدم اول: مقدار دهی اولیه**

برای نمونه تعداد کروموزوم ها درون جمعیت را عدد ۶ تعریف می‌کنیم، سپس مقادیر تصادفی را برای ژنهای که تشکیل دهنده کروموزوم مورد نظر ما هستند تولید می‌کنیم. پس همانطور که ملاحظه می‌شود ۶ کروموزومی را که در اختیار داریم به صورت زیر با اعداد تصادفی مقدار دهی می‌نماییم:

Chromosome[1] = [a ; b ; c ; d] = [۱۲;۰۵;۳۳;۰۸]

Chromosome[2] = [a ; b ; c ; d] = [۰۳ ; ۲۱ ; ۱۸; ۰۳ ]

Chromosome[3] = [a ; b ; c ; d] = [۱۰:۰۴:۱۳:۱۴]

Chromosome[4] = [a ; b ; c ; d] = [۲۰:۰۱:۱۰۰۶]

Chromosome[5] = [a ; b ; c ; d] = [۰۱:۰۴:۱۳:۱۹]

Chromosome[6] = [a ; b ; c ; d] = [۲۰:۰۵:۱۷:۰۱]

**قدم دوم: ارزیابی راه حل ها (کروموزوم ها)**

در این مرحله مقدار تابع هدف برای هر کروموزومی که در مرحله که در مرحله مقدار دهی جمعیت اولیه تولید شده است محاسبه می‌شود:

**F\_obj[1] = Abs(( 12 + 2\*05 + 3\*23 + 4\*08 ) - 30)**

**= Abs((12 + 10 + 69 + 32 ) - 30)**

**= Abs(123 - 30)**

**= 93**

**F\_obj[2] = Abs((02 + 2\*21 + 3\*18 + 4\*03) - 30)**

**= Abs((02 + 42 + 54 + 12) - 30)**

**= Abs(110 - 30)**

**= 80**

**F\_obj[3] = Abs((10 + 2\*04 + 3\*13 + 4\*14) - 30)**

**= Abs((10 + 08 + 39 + 56) - 30)**

**= Abs(113 - 30)**

**= 83**

**F\_obj[4] = Abs((20 + 2\*01 + 3\*10 + 4\*06) - 30)**

**= Abs((20 + 02 + 30 + 24) - 30)**

**= Abs(76 - 30)**

**= 46**

**F\_obj[5] = Abs((01 + 2\*04 + 3\*13 + 4\*19) - 30)**

**= Abs((01 + 08 + 39 + 76) - 30)**

**= Abs(124 - 30)**

**= 94**

**F\_obj[6] = Abs((20 + 2\*05 + 3\*17 + 4\*01) - 30)**

**= Abs((20 + 10 + 51 + 04) - 30)**

**= Abs(85 - 30)**

**= 55**

**مرحله سوم: انتخاب**

با ارزشترین کروموزوم از شانس و احتمال بیشتری برای انتخاب شدن به منظور تولید نسل بعدی دارد. برای محاسبه احتمال برازش باید برازش هر کروموزوم محاسبه شود. برای جلوگیری از مشکل تقسیم بر صفر مقدار تابع هدفی که برای هر کروموزوم در مرحله قبل بدست آمد با عدد ۱ جمع می‌شود:

Fitness[1] = 1 / (1+F\_obj[1])

= 1 / 94

= 0.0106 Fitness[2] = 1 / (1+F\_obj[2])

= 1 / 81 = 0.0123

Fitness[3] = 1 / (1+F\_obj[3])

= 1 / 84

= 0.0119 Fitness[4] = 1 / (1+F\_obj[4])

= 1 / 47 = 0.0213

Fitness[5] = 1 / (1+F\_obj[5])

= 1 / 95

= 0.0105 Fitness[6] = 1 / (1+F\_obj[6])

= 1 / 56 = 0.0179

**Total** = 0.0106 + 0.0123 + 0.0119 + 0.0213 + 0.0105 + 0.0179 = 0.0845

احتمال مربوط به هر کروموزوم را بوسیله فرومول P[i] = Fitness[i]/Total محاسبه می‌نماییم:

P[1] = 0.0106 / 0.0845 = 0.1254

P[2] = 0.0123 / 0.0845 = 0.1456

P[3] = 0.0119 / 0.0845 = 0.1408

P[4] = 0.0213 / 0.0845 = 0.2521

P[5] = 0.0105 / 0.0845 = 0.1243

P[6] = 0.0179 / 0.0845 = 0.2118

با توجه به احتمالاتی که در بالا محاسبه شده است مشاهده می‌شود که کروموزوم ۴ بیشترین برازش را دارا می‌باشد و بنابراین احتمال انتخاب شد

آن برای نسل بعدی کروموزوم ها بسیار بالا است. برای فرایند انتخاب از روش جرخ رولت استفاده می‌کنیم که بدین منظور ابتدا باید مقادیر احتمال تجمعی(cumulative probability) را محاسبه کنیم:

C[1] = 0.1254

C[2] = 0.1254 + 0.1456 = 0.2710

C[3] = 0.1254 + 0.1456 + 0.1408 = 0.4118

C[4] = 0.1254 + 0.1456 + 0.1408 + 0.2521 = 0.6639

C[5] = 0.1254 + 0.1456 + 0.1408 + 0.2521 + 0.1243 = 0.7882

C[6] = 0.1254 + 0.1456 + 0.1408 + 0.2521 + 0.1243 + 0.2118 = 1.0

با داشتن مقادیر احتمال تجمعی استفاده از چرخ رولت در فرآیند انتخاب امکان پذیر می گردد. فرآیند بعدی تولید یک عدد تصادفی در محدوده ۰ تا ۱ است که آن را R می‌نامیم:

**R**[1] = 0.201

**R**[2] = 0.284

**R**[3] = 0.099

**R**[4] = 0.822

**R**[5] = 0.398

**R**[6] = 0.501

اگر برای مثال عدد تصادفی **R**[1] بزرگتر از **C**[1] و کوچکتر از **C**[2]بود در این صورت **Chromosome**[2] به عنوان یک کروموزوم در جمعیت جدید برای تولید نسل بعدی مورد استفاده قرار می‌گیرد. با توجه به مقادیر نمونه های بالا خواهیم دادشت:

NewChromosome[1] = Chromosome[2]

NewChromosome[2] = Chromosome[3]

NewChromosome[3] = Chromosome[1]

NewChromosome[4] = Chromosome[6]

NewChromosome[5] = Chromosome[3]

NewChromosome[6] = Chromosome[4]

و کروموزوم ها در جمعیت به صورت زیر خواهد شد:

Chromosome[1] = [02;21;18;03]

Chromosome[2] = [10;04;13;14]

Chromosome[3] = [12;05;23;08]

Chromosome[4] = [20;05;17;01]

Chromosome[5] = [10;04;13;14]

Chromosome[6] = [20;01;10;06]

**مرحله ۴: برش(crossover)**

در این مثال از برش یک نقطه ای (one cut point) استفاده می‌نماییم. به صورت تصادفی یک محل در کروموزوم والدین انتخاب شده و سپس زیر کروموزوم

هایی که از این نقطه بدست آمده‌‌ اند در دو واحد تعویض می‌شود. کروموزوم های والدینی که با همدیگر جفت شده اند به صورت تصادفی انتخاب می‌شوند و تعداد کروموزوم هایی که جفت می‌شوند توسط پارامتری به نام نرخ برش (crossover rate **ρc)** معین می‌شود. شبه کد فرآیند برش در ذیل آورده شده است:

begin

**k**← 0;

while(**k**<**population**) do

**R**[**k**] = **random**(0-1);

if(**R**[**k**]< **ρc**) then

select **Chromosome**[**k**] as parent;

end;

**k** = **k** + 1;

end;

end;

اگر **R**[**k**]<**ρc** باشد کروموزوم **k** به عنوان یکی از والدین انتخاب می‌شود. فرض کنیم که نرخ برش را ۲۵٪ در نظر گرفته باشیم، در این صورت کروموزم شماره **k** در صورتی برای برش انتخاب می‌شود اگر مقدار عدد تصادفی تولید شده برای کروموزوم **k** کمتر از ۰.۲۵با شد. در فرایند را به صورت زیر ادامه می‌دهیم. ابتدا به تعداد جمعیتی که داریم عدد تصادفی **R** تولید می‌نماییم:

**R**[1] = 0.191

**R**[2] = 0.259

**R**[3] = 0.760

**R**[4] = 0.006

**R**[5] = 0.159

**R**[6] = 0.340

با توجه به اعداد نمونه تصادفی که در بالا آورده شده است، والدین **Chromosome**[1], **Chromosome**[4] و **Chromosome**[5] برای عملگر برش انتخاب می‌شوند. حالا با توجه این والدین ترکیب همه انها را بدست آوریم که بصورت زیر خواهد بود:

**Chromosome**[1] >< **Chromosome**[4]

**Chromosome**[4] >< **Chromosome**[5]

**Chromosome**[5] >< **Chromosome**[1]

بعد از انتخاب کروموزوم ها برای برش، فرآیند بعدی مشخص نمودن محل نقطه برش است. این کار با تولید یک عدد تصادفی بین عدد ۱ تا طول کروموزوم ۱۰ انجام می‌شود. در مثال ما عدد تصادفی باید بین ۱ تا ۳ باشد. بعد از این که نقطه برش را بدست آوردیم کروموزوم های والد در آن نقطه برش خورده و ژن های آنها تعویض می‌شود و یک فرزند جدید تولید می‌شود. برای مثال ما سه عدد تصادفی ذیل را تولید نمودیم:

**C**[1] = 1

**C**[2] = 1

**C**[3] = 2

و کروموزوم ها به صورت زیر تحت تاثیر عملگر برش قرار می‌گیرند. همانگونه که ملاحظه می‌شود ژنهای والدین به ترتیب در ژن های شماره ۱، شماره ۱و شماره ۲ برش داده می‌شوند:

**Chromosome**[1] = **Chromosome**[1] >< **Chromosome**[4]

= [02;21;18;03] >< [20;05;17;01]

= [02;05;17;01]

**Chromosome**[4] = **Chromosome**[4] >< **Chromosome**[5]

= [20;05;17;01] >< [10;04;13;14]

= [20;04;13;14]

**Chromosome**[5] = **Chromosome**[5] >< **Chromosome**[1]

= [10;04;13;14] >< [02;21;18;03]

= [10;04;18;03]

بعد از انجام اولین فرآیند برش در الگوریتم ژنتیک مورد نظرمان جمیعن ما به صورت زیر خواهد بود:

**Chromosome**[1] = [02;05;17;01]

**Chromosome**[2] = [10;04;13;14]

**Chromosome**[3] = [12;05;23;08]

**Chromosome**[4] = [20;04;13;14]

**Chromosome**[5] = [10;04;18;03]

**Chromosome**[6] = [20;01;10;06]

**مرحله ۵: جهش**

تعداد کروموزوم هایی که جمعیت مورد جهش قرار می‌گیرند توسط پارامتر نرخ جهش مشخص می‌شوند با تعویض تصادفی مقدار یک ژن که در یک موقعیت تصادفی قرار دارد با مقدار جدید انجام می‌شود. برای انجام این فرایند مراحل که در ادامه گفته می‌شود را انجام می‌دهیم. ابتدا باید مجموع کل ژن های موجود در جمعیت را محاسبه کنیم.در مثال ما از فرمول زیر بدست می‌آید.

تعداد کل ژنها = تهداد ژنهای هر کروموزوم \* تعداد جمعیت = ۶\*۴=۲۴

فرآیند جهش با تولید یک عدد تصادفی بین ۱ و تعداد کل ژنهای (۱ تا ۲۴ در مثال ما) انجام می‌شود. اگر مقدار عدد تصادفی تولید شده کمتر از نرخ جهش (mutation\_rate ) باشد در این صورت موقعیت این ژن در کروموزوم نشان گذاری می‌شود. فرض کنیم که نرخ جهش را ۱۰٪ در نظر بگیریم، که در این صورت انتظار داریم که ۱۰٪(۰.۱) کل ژنها در جمعیت جهش داده شوند:

تعداد کل جهش ها = ۲۴ \* ۰.۱ = ۲.۳ = ۲

حالا اگر فرض کنیم که اعداد تصادفی که تولید شده است ۱۲ و ۱۸ با‌شند در این صورت کروموزوم های که جهش دارند کروموزوم شماره ۳ در ژن شماره ۴ آن و کروموزوم شماره ۵ رد ژن شماره ۲ آن خواهند بود. مقدار ژنهای جهش یافته در نقطه جهش بوسیله اعداد تصادفی بین اعداد تصادفی بین ۰ تا ۳۰ عوض می‌شوند. اگر اعداد تصادفی تولید شده ۲و ۵ باشند سپس ترکیب کروموزوم ها بعد از جهش به صورت زیر است:

Chromosome[1] = [02;05;17;01]

Chromosome[2] = [10;04;13;14]

Chromosome[3] = [12;05;23;02]

Chromosome[4] = [20;04;13;14]

Chromosome[5] = [10;05;18;03]

Chromosome[6] = [20;01;10;06]

بعد از پایان فرآیند جهش یک تکرار یا یک نسل از الگوریتم ژنتیک داریم که باید این نسل جدید را مجددا توسط تابع هدف مورد ارزیابی قرار دهیم که ارزیابی آن مانند آنچه قبلا گفته شد انجام می‌شود:

**Chromosome**[1] = [02;05;17;01]

**F\_obj**[1] = Abs(( 02 + 2\*05 + 3\*17 + 4\*01 ) - 30)

= Abs((2 + 10 + 51 + 4 ) - 30)

= Abs(67 - 30)

= 37

**Chromosome**[2] = [10;04;13;14]

**F\_obj**[2] = Abs(( 10 + 2\*04 + 3\*13 + 4\*14 ) - 30)

= Abs((10 + 8 + 33 + 56 ) - 30)

= Abs(107 - 30)

= 77

**Chromosome**[3] = [12;05;23;02]

**F\_obj**[3] = Abs(( 12 + 2\*05 + 3\*23 + 4\*02 ) - 30)

= Abs((12 + 10 + 69 + 8 ) - 30)

= Abs(87 - 30)

= 47

**Chromosome**[4] = [20;04;13;14]

**F\_obj**[4] = Abs(( 20 + 2\*04 + 3\*13 + 4\*14 ) - 30)

= Abs((20 + 8 + 39 + 56 ) - 30)

= Abs(123 - 30)

= 93

**Chromosome**[5] = [10;05;18;03]

**F\_obj**[5] = Abs(( 10 + 2\*05 + 3\*18 + 4\*03 ) - 30)

= Abs((10 + 10 + 54 + 12 ) - 30)

= Abs(86 - 30)

= 56

**Chromosome**[6] = [20;01;10;06]

**F\_obj**[6] = Abs(( 20 + 2\*01 + 3\*10 + 4\*06 ) - 30)

= Abs((20 + 2 + 30 + 24 ) - 30)

= Abs(76 - 30)

= 46

همانگونه که از ارزیابی کروموزوم ها در جمعیت جدید مشاهده می‌شود تابع هدف کاهش یافته است که به این معناست که ما کروموزومها یا راه حل های بهتری نسبت به کروموزومهای نسل قبل داریم. بنابراین کروموزومها برای تکرار بعدی به صورت زیر خواهد بود:

**Chromosome**[1] = [02;05;17;01]

**Chromosome**[2] = [10;04;13;14]

**Chromosome**[3] = [12;05;23;02]

**Chromosome**[4] = [20;04;13;14]

**Chromosome**[5] = [10;05;18;03]

**Chromosome**[6] = [20;01;10;06]

کروموزوم های جدید تحت تأثیر فرآیند مشابهی با انچه با نسل قبلی کروموزومها انجام شد قرار می‌گیرند. مثل ارزیابی، انتخاب، برش و جهش و در پایان هر تکرار نیز نسل جدیدی از کروموزومها برای تکرار بعدی تولید می‌شود. این فرآیند تا رسیدن به تعداد تکرار های از پیش تعریف شده با بدست آمدن یک شرط ادامه می‌یابند. در این مثال حدوداً بعد از ۵۰ نسل بهترین کروموزوم بدست می‌آید:

Chromosome = [07; 05; 03; 01]

که بدان معناست که که اگر این اعداد معادله این که در ابتدا داشتیم قرار دهیم خواهیم داشت:

**a** + 2**b** + 3**c** + 4**d** = 30

7 + (2 \* 5) + (3 \* 3) + (4 \* 1) = 30

# در این مثال ساده به خوبی کارایی و قدرت الگوریتم ژنتیک در حل **مسائل** بهینه سازی مشاهده می‌شود.