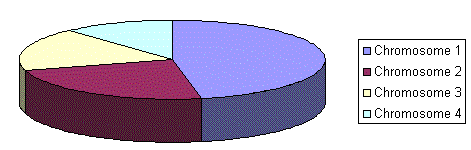
Генетични алгоритми. Кръстосване и мутации. Сиво кодиране. Пример с Ханойските кули

При [скицирането на GA](https://www.obitko.com/tutorials/genetic-algorithms/bulgarian/ga-basic-description.php#outline), хромозомите се избират от популацията за кръстосване. Проблема е в това как да бъдат подбрани тези хромозоми. Съгласно еволюционната теория на Дарвин най-добрите т рябва да оцелеят и да създадат ново потомство. Има много методи за избор на най-добрите хромозоми, примерно селекция по кръгова рулетка, селекция на Boltzman, състезателна селекция, селекция по ранг, селекция на устойчивите състояния и някои други.

**Селекция по Кръгова Рулетка**

Родителите се избират според тяхната жизнеспособност. Колкото по-добри са хромозомите, толкова по-добър шанс имат да бъдат избрани. Представете си колело на рулетка където са поставени всички хромозоми от популацията, всеки има собствено място с г олемина съответна на функцията му за жизнеспособност, както на следната фигура.



Тогава топчето се хвърля и избира хромозома. Хромозома с по-голяма жизнеспособност ще бъде избирана повече пъти.

Това може да бъде симулирано чрез следния алгоритъм.

1. **[Сума]** Изчислява се сумата от жизнезпособността на всички хромозоми в популацията - сума ***S***.
2. **[Избор]** Генериране на случайно число в интервала ***(0,S)*** - ***r***.
3. **[Цикъл]** Обхожда се популацията и се сумира жизнеспособностите ***0*** - сума ***s***. Когато сумата ***s*** е по голяма от ***r***, се спира и се връща хромозомата която се разглежда в момента.

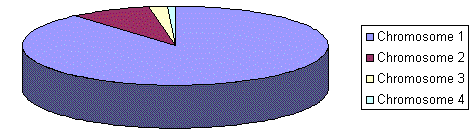
Разбира се, стъпка **1** се извършва само веднъж за всяка популация.

**Селекция по Ранг**

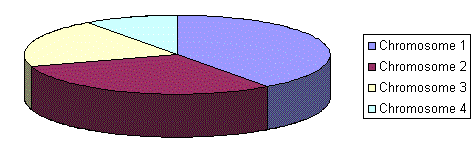
Предишния метод за селекция ще има проблем когато жизнеспособността се различава много. Примерно, ако жизнеспособността на най-добрата хромозома е 90% от цялата кръгова рулетка тогава останалите хромозоми ще имат много малки шансове да бъдат избрани.

Селекцията по ранг първо подрежда популацията и тогава всяка хромозома получава жизнеспособност спрямо тази подредба. Най-лошата ще има жизнеспособност 1, втората ложа 2 т.н. и най-добрата ще има жизнеспособност N (броя на хромозомите в популацията).

На следващата фигура може да се види, как се променя ситуацията след промяна жизнеспособността при пореден номер.



*Ситуацията преди подредбата (графика на жизнеспособност)*



*Ситуацията след подредбата (графика на поредните номера)*

След всичко това хромозомите имат шанс да бъдат избрани.Но този метод може да доведе до бавна сходимост, защото най-добрите хромозоми не се различават толкова много от останалите.

**Селекция на Устойчивите-Състояние**

Това не е особен метод за избор на родители. Основната идея на тази селекция е че голяма част от хромозомите би трябвало да оцелеят до следващото поколени.

GA тогава работи по следния начин. Във всяко поколение се избират няколко (добри - с висока жизнеспособност) хромозоми за създаването на новото потомство. След това няколко (лоши - с ниска жизнеспособност) хромозоми се премахват и новото потомство се пос тавя на тяхно място. Останалата част от популацията оцелява и в новото поколение.

**Прехвърляне**

Идеята на прехвърлянето бе представена вече. Когато се създава нова популация чрез кръстосване и мутация, има голям шанс, да бъде изгубена най-добрата хромозома.

Прехвърляне е името на метод, който първо копира най-добрата хромозома (или няколко най-добри хромозоми) в новата популация. Останалите се формират по класическия начин. Прехвърлянето може много бързо да увеличи изпълнението на GA, защото предпазва загуб ата на най-доброто открито решение.

**Кръстосването и мутацията** са двете основни операции на GA. Изпълнението на GA много зависи от тях. Типа и изпълнението на операциите зависи от кодирането и от проблема също.

**Двоично Кодиране**

**Кръстосване**

Единична точка на кръстосване - избира се една точка на кръстосване, двоичния низ от началото на хромозомата до точката на кръстосване е копие от единия родител, останалата е копие от втория родител

**11001**011+11011**111** = **11001111**

Кръстосване в две точки - избирате се две точки на кръстосване, двоичния низ от началото на хромозомата до първата точка на кръстосване е копие от първия родител, частта от първата точка на кръстосване до втората точка на кръстосване е копие от втория родител и останалото е копие от първия родител

**11**0010**11** + 11**0111**11 = **11011111**

Еднородно кръстосване - битове произволно се копират от първия и от втория родител

1**10**010**11** + **1**10**111**01 = 11011111

**Аритметично кръстосване** - някоя аритметична операция се извършва за получаване на новото потомство

11001011 + 11011111 = 11001001 (AND)

**Мутация**

Инвертиране на бит - избрания бит се инвертира

1**1**001001 =>  1**0**001001

**Кодиране на Пермутации**

**Кръстосване**

Кръстосване в една точка - избира се една точка на кръстосване, one crossover point is selected, до тази точка пермутацията се копира от първия родител, след това втория родител се сканира и ако номерата все още не са потомството се добавят

*Забележка: има повече начини за представяне останалото след точката на кръстосване*

(**1 2 3 4 5** 6 7 8 9) + (**4 5 3 6 8** 9 7 2 1) = (**1 2 3 4 5** 6 8 9 7)

**Мутация**

**Смяна на реда** - два числа се избират и се разменят

(1 **2** 3 4 5 6 **8** 9 7) => (1 **8** 3 4 5 6 **2** 9 7)

**Кодиране по Стойност**

**Кръстосване**

Всички кръстосвания от [двоично кодиране](https://www.obitko.com/tutorials/genetic-algorithms/bulgarian/crossover-mutation.php#binary) може да бъдат използвани

**Мутация**

Добавяне на малки числа (за кодиране на реални стойности) - към избраната стойност се добавя (или изважда) малко число

(1.29  5.68  **2.86**  **4.11**  5.55) => (1.29 5.68  **2.73**  **4.22**  5.55)

**Кодиране в Дърво**

**Кръстосване**

Кръстосване на дърво - и в двата родителя се избира една точка на кръстосване, родителите се разделят в тази точка на кръстосване и си разменят частта под точката за кръстосване за да представят новото потомство

**Двоично Кодиране**

**Кръстосване**

Единична точка на кръстосване - избира се една точка на кръстосване, двоичния низ от началото на хромозомата до точката на кръстосване е копие от единия родител, останалата е копие от втория родител

**11001**011+11011**111** = **11001111**

Кръстосване в две точки - избирате се две точки на кръстосване, двоичния низ от началото на хромозомата до първата точка на кръстосване е копие от първия родител, частта от първата точка на кръстосване до втората точка на кръстосване е копие от втория родител и останалото е копие от първия родител

**11**0010**11** + 11**0111**11 = **11011111**

Еднородно кръстосване - битове произволно се копират от първия и от втория родител

1**10**010**11** + **1**10**111**01 = 11011111

Аритметично кръстосване - някоя аритметична операция се извършва за получаване на новото потомство

11001011 + 11011111 = 11001001 (AND)

**Мутация**

**Инвертиране на бит** - избрания бит се инвертира

1**1**001001 =>  1**0**001001

**Кодиране на Пермутации**

**Кръстосване**

Кръстосване в една точка - избира се една точка на кръстосване, one crossover point is selected, до тази точка пермутацията се копира от първия родител, след това втория родител се сканира и ако номерата все още не са в потомството се добавят

*Забележка: има повече начини за представяне останалото след точката на кръстосване*

(**1 2 3 4 5** 6 7 8 9) + (**4 5 3 6 8** 9 7 2 1) = (**1 2 3 4 5** 6 8 9 7)

**Мутация**

**Смяна на реда** - два числа се избират и се разменят

(1 **2** 3 4 5 6 **8** 9 7) => (1 **8** 3 4 5 6 **2** 9 7)

**Кодиране по Стойност**

**Кръстосване**

Всички кръстосвания от [двоично кодиране](https://www.obitko.com/tutorials/genetic-algorithms/bulgarian/crossover-mutation.php#binary) може да бъдат използвани

**Мутация**

Добавяне на малки числа (за кодиране на реални стойности) - към избраната стойност се добавя (или изважда) малко число

(1.29  5.68  **2.86**  **4.11**  5.55) => (1.29 5.68  **2.73**  **4.22**  5.55)

**Кодиране в Дърво**

**Кръстосване**

Кръстосване на дърво - и в двата родителя се избира една точка на кръстосване, родителите се разделят в тази точка на кръстосване и си разменят частта под точката за кръстосване за да представят новото потомство

**Двоично Кодиране**

**Кръстосване**

Единична точка на кръстосване - избира се една точка на кръстосване, двоичния низ от началото на хромозомата до точката на кръстосване е копие от единия родител, останалата е копие от втория родител

**11001**011+11011**111** = **11001111**

Кръстосване в две точки - избирате се две точки на кръстосване, двоичния низ от началото на хромозомата до първата точка на кръстосване е копие от първия родител, частта от първата точка на кръстосване до втората точка на кръстосване е копие от втория родител и останалото е копие от първия родител

**11**0010**11** + 11**0111**11 = **11011111**

Еднородно кръстосване - битове произволно се копират от първия и от втория родител

1**10**010**11** + **1**10**111**01 = 11011111

Аритметично кръстосване - някоя аритметична операция се извършва за получаване на новото потомство

11001011 + 11011111 = 11001001 (AND)

**Мутация**

**Инвертиране на бит** - избрания бит се инвертира

1**1**001001 =>  1**0**001001

**Кодиране на Пермутации**

**Кръстосване**

Кръстосване в една точка - избира се една точка на кръстосване, one crossover point is selected, до тази точка пермутацията се копира от първия родител, след това втория родител се сканира и ако номерата все още не са в потомството се добавят

(**1 2 3 4 5** 6 7 8 9) + (**4 5 3 6 8** 9 7 2 1) = (**1 2 3 4 5** 6 8 9 7)

**Мутация**

**Смяна на реда** - два числа се избират и се разменят

(1 **2** 3 4 5 6 **8** 9 7) => (1 **8** 3 4 5 6 **2** 9 7)

**Кодиране по Стойност**

**Кръстосване**

Всички кръстосвания от [**двоично** кодиране](https://www.obitko.com/tutorials/genetic-algorithms/bulgarian/crossover-mutation.php#binary) може да бъдат използвани

**Мутация**

**Добавяне** на малки числа (за кодиране на реални стойности) - към избраната стойност се добавя (или изважда) малко число

(1.29  5.68  **2.86**  **4.11**  5.55) => (1.29 5.68  **2.73**  **4.22**  5.55)

**Кодиране в Дърво**

**Кръстосване**

**Кръстосване на дърво**- и в двата родителя се избира една точка на кръстосване, родителите се разделят в тази точка на кръстосване и си разменят частта под точката за кръстосване за да представят новото потомство

**Мутация**

**Смяна номера на операция**- избраните възли се сменят

**Мутация**

**Смяна номера на операция**- избраните възли се сменят

**Мутация**

**Смяна номера на операция**- избраните възли се сменят

https://www.obitko.com/tutorials/genetic-algorithms/bulgarian/crossover-mutation.php