



Тема 1 Генетични алгоритми

Генетичният алгоритъм (ГА) е стохастичен метод за решаване на оптимизационни задачи с и без ограничения, основан на естествената селекция, процесът определящ биологичната еволюция. ГА променя на всяка стъпка популацията от индивиди, като избира тези от текущото поколение, които ще продължат развитието си, т.е. ще се използват за следващата генерация. Селекцията на най-добрите индивиди става въз основа на функционал или функционали (целеви функции), даващи оценка за близостта на индивида с желаното решение. Правилата, по които става изборът на индивиди от текущата популация са:

- Правила за селекция – избират се най-добрите индивиди, които да участват в създаването на следващото поколение или да бъдат прехвърлени без промяна в следващото поколение;

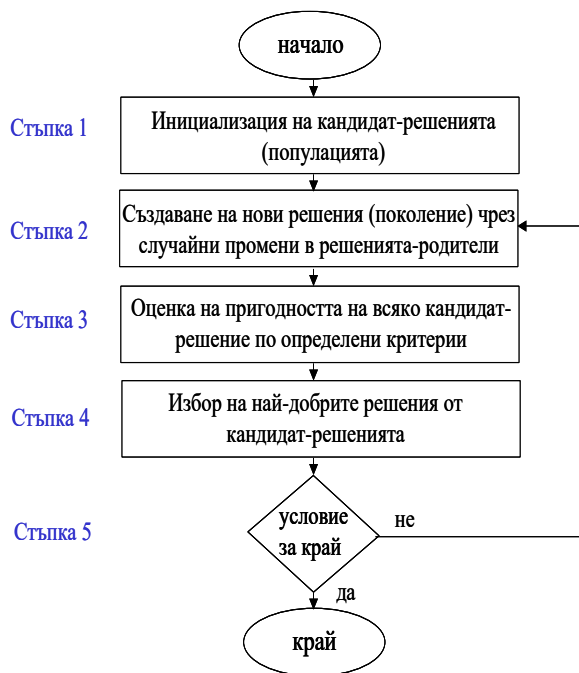
- Правила за кръстосване – избраните чрез селекция индивиди се кръстосват, като се цели да се получат индивиди, които да наследят най-добрите характеристики на родителите си;

- Правила за мутация – чрез случайна промяна на някои от гените се гарантира, дори и нито един от индивидите в текущото поколение да не съдържа необходимия ген, пак да се достигне екстремум [4].

Генетични алгоритми

Генетичните алгоритми представляват метод за търсене с налучкване, при който основната идея е да се симулират генетичните и еволюционните процеси в природата. Така, за дадена оптимизационна задача, първоначално се построяват няколко произволни неоптимални решения. След което следва запазване на най-сполучливите от тях и на тяхната база се построяват нови с очакването, че те ще бъдат още по-добри. Неоптималните и безперспективни решения се изолират от по-нататъшно разглеждане. Очевидна е аналогия с процеса на еволюция на видовете в природата (“оцеляване на най-доброто”).

----- www.eufunds.bg -----



Фиг.1 Блок схема на протичане на подбора на най- добро решение

Дефиниция на генетичен алгоритъм

Вариант на стохастично търсене в лъч, при което новите състояния се генерират чрез комбиниране на двойки родителски състояния вместо чрез модифициране на текущото състояние.

Основни принципи

- Състоянията се представят като низове над дадена крайна азбука (често като низове от нули и единици).
- Оценяваща функция (fitness function): оценява пригодността (близостта до целта) на съответното състояние. Има по-големи стойности за по-добрите състояния.
- Алгоритъмът започва работа с множество (популация) от k случайно генерирани състояния (поколение 0).
- Принципи на получаването на състоянията от следващите поколения: селекция, кръстосване, мутация.

Основни термини

----- www.eufunds.bg -----

Проект BG05M2OP001-2.016-0003, „Модернизация на Национален военен университет "В. Левски" - Велико Търново и Софийски университет "Св. Климент Охридски" - гр. София, в професионално направление 5.3 Компютърна и комуникационна техника“, финансиран от Оперативна програма образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез

гр.
направление
„Наука и



В литературните и електронни информационни източници, касаещи теорията на генетичните алгоритми, са въведени следните основни базови понятия [1,2,3]:

- **ген** (*gene*) – минимален набор от организирани символи, съдържащи информация със значение за решаваната задача, най-често един бит;
- **хромозома** (*chromosome*) – набор от гени с определена дължина (брой елементи), които формират някакво допустимо решение на поставената задача, най-често последователност от битове;
- **генотип** (*genotype*) – определя типа на информацията в гените (най-често бинарни стойности, но могат да бъдат и цели числа, символи и др.) и дължината на хромозомите;
- **фенотип** (*phenotype*) – съвкупността от особености, проявени в резултат на избора на определен генотип;
- **геном** (*genome*) – понякога е възможно да се търси комплексно решение, съставено от решения на отделни задачи – в този случай набора от хромозоми формират така наречения геном, т.е. съвкупност от допустими частни решения, които формират допустимо комплексно решение; в някои източници терминът „геном“ се използва равностойно на „хромозома“.
- **индивид** (*individual*) – условен притежател на съответната хромозома (геном) по аналогия с природата макар, че при организмите броят на хромозомите винаги е повече от един – например, при хората всяка клетка съдържа 46 хромозома, групирани в 23 двойки;
- **популация** (*population*) – набор от n на брой индивиди, с което се формира моментно множество от възможни допустими решения на поставената задача; в някои източници, равностойно на „популация“, се използва също термина **поколение** (*generation*);
- **фитнес функция** (*fitness function*) – или функция на сближаване; това е функцията, за която се търси оптимално решение; нарича се още обектна функция (*objective function*).

----- www.eufunds.bg -----

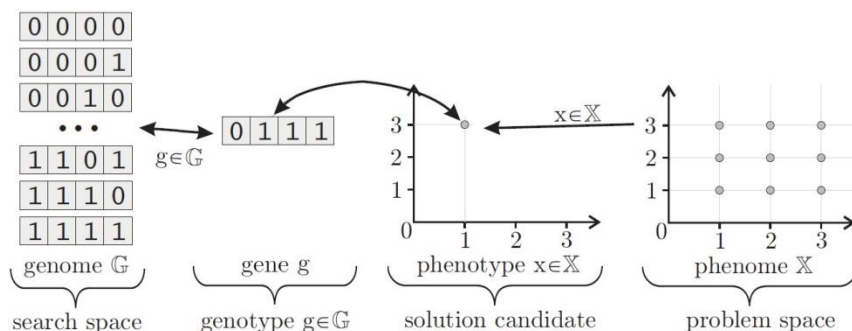


Алгоритмична последователност.

Най-общо казано, генетичните алгоритми търсят решение на оптимизационната задача като на всяка итерация се генерира нова популация (набор от решения), която се очаква да е по-добра от предходната по предварително избран критерий и този процес продължава до достигане на целевата функция с необходимото зададено приближение [1, 4].

Развитието на един генетичен алгоритъм протича в следната последователност от действия.

1) **Избор на генотип** (*Genome Definition*) – преди да се стартира алгоритъма, следва да се дефинира множеството от възможни търсени решения, в зависимост от спецификата на оптимизационната задача и наличните логически и физически ограничителни условия – представения на фиг. 2 примерен избор на генотип, пояснява графично действията на този етап при конкретна задача.



Фиг. 2. Избор на генотип за множеството от възможни търсени решения

2) **Селекция.** В генерирането на състоянията от следващото поколение участват някои от най-добрите представители на текущото поколение (съгласно оценяващата функция), избрани на случаен принцип.

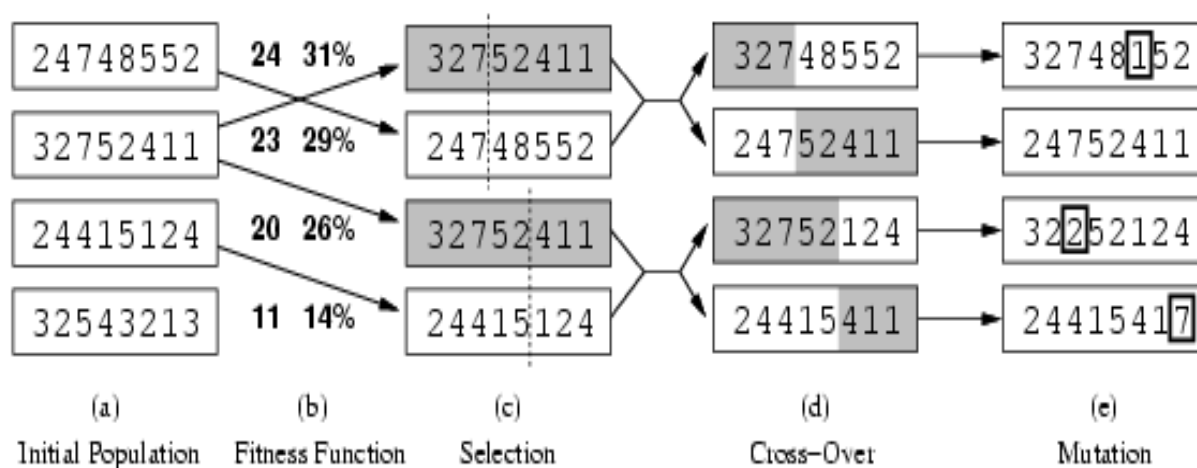


3) Кръстосване. Избират се двойка “родителски” състояния и се определя т.нар. точка на кръстосването им (позиция в двата низа). Състоянието – наследник се получава чрез конкатенация на началната част на първия и крайната част на втория родител. Възможно е да се получи и друг наследник, в конструирането на който участват неизползваните части на двамата родители.

4) Мутация. Извършване на случайни промени в случайно избрана малка част от новата популация с цел да се осигури възможност за достигане на всяка точка от пространството на състоянията и да се избегне опасността от попадане в локален екстремум.

Пример: задача за 8-те царици. Да се намери такова разположение на 8 царици върху шахматна дъска, че никоя двойка от тях да не се бият взаимно.

- Представяне на състоянията: всяко състояние се представя чрез 8-компонентен вектор, i -тият компонент на който представя вертикалната позиция p_i на i -тата царица(виж фиг.3).



Фиг.3. чрез 8-компонентни вектори[1]

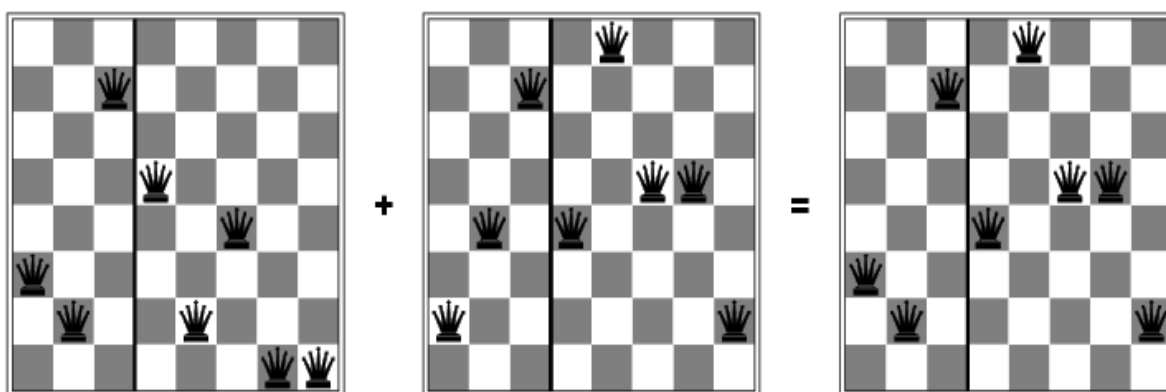


• **Оценяваща функция:** за всяко състояние връща като резултат броя на двойките царици, които не се бият взаимно ($\min = 0$, $\max = 8 \times 7/2 = 28$)

• $24/(24+23+20+11) = 31\%$

• $23/(24+23+20+11) = 29\%$ и т.н

На фиг. 3. е представен пример за кръстосване на двойки царици.



Фиг.4. Възможности за кръстосване

• **Кръстосване в единична точка** Избира се една точка на кръстосване. Низът – резултат от началото си до точката на кръстосване е копие на началната част на единия родител, останалата му част е копие на съответната част на втория родител.

Parent A

8691247536

+

Parent 2

1234567892

=

Offspring

8691567892

Кръстосване в две точки

Избират се две точки на кръстосване. Низът – резултат от началото си до първата точка на кръстосване е копие на съответната част от първия

----- www.eufunds.bg -----

Проект BG05M2OP001-2.016-0003 „Модернизация на Национален военен университет "В. Левски" - гр. Велико Търново и Софийски университет "Св. Климент Охридски" - гр. София, в професионално направление 5.3 Компютърна и комуникационна техника“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез



родител, частта на резултата от първата точка на кръстосване до втората точка на кръстосване е копие на съответната част на втория родител и останалото е копие на оставащата след втората точка на кръстосване част на първия родител.

Parent A		Parent 2		Offspring
8691247536	+	1234567892	=	8691567536

Аритметично кръстосване

Извършва се определена операция (аритметична, логическа и т.н.) между двамата родители и в резултат се получава новото потомство.

Parent A		Parent 2		Offspring
1101011010	+	1101111101	=	1101011000

двоично кодиране, операция AND между съответните битове

Характерни приложения на генетичните алгоритми

- Решаване на оптимизационни задачи (задача за търговския пътник, задача за раницата и др.)
- Решаване на задачи за удовлетворяване на ограничения
- Избор на стратегия при игри за двама (по-общо, N) играчи
- Самообучение на невронни мрежи (уточняване на теглата на връзките между елементите в невронна мрежа с определена архитектура)
- Генетично програмиране

Пример: задача за търговския пътник. Дадени са група градове и разстоянията между тях. Търговският пътник трябва да посети всички градове, като минимизира общата дължина на пътя. Да се намери такъв

----- www.eufunds.bg -----

Проект BG05M2OP001-2.016-0003 „Модернизация на Национален военен университет "В. Левски" - гр. Велико Търново и Софийски университет "Св. Климент Охридски" - гр. София, в професионално направление 5.3 Компютърна и комуникационна техника“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез



маршрут на търговския пътник, който започва от даден град, завършва в същия град, включва всички останали градове точно по веднъж и образува път с възможно най-малка обща дължина

Кодиране (представяне на състоянията)

Състоянията се представят като поредици от означенията на съответните градове в реда, в който търговският пътник ще ги посети.

Пример: състоянието, представено чрез низа "153264798", означава маршрут на търговския пътник, съставен от следната поредица от градове (предполага се, че градовете са означени с цифрите от 1 до 9): 1-5-3-2-6-4-7-9-8-1.

Кръстосване

Може да бъде както в единична точка, така и в две точки. И в двата случая са необходими допълнителни действия, които да осигурят получаването на коректно състояние като резултат от кръстосването.

Пример: кръстосване в две точки с получаване на поколение от двама наследници

Ако са две родителски състояния, то при кръстосването им ще се получат

$$\begin{array}{l} p_1 = \text{"1234 | 567 | 8"} \\ p_2 = \text{"8521 | 364 | 7"} \end{array} \quad \text{и} \quad \begin{array}{l} c_1 = \text{"1234 | 364 | 8"} \\ c_2 = \text{"8521 | 567 | 7"} \end{array}$$

са две родителски състояния, то при кръстосването им ще се получат

Очевидно c_1 и c_2 представят некоректни състояния, затова се прибягва до следната допълнителна операция: размените $3 \leftrightarrow 5$, $6 \leftrightarrow 6$ и $4 \leftrightarrow 7$, които са извършени при кръстосването на родителските състояния в областта между двете точки на кръстосване, се повтарят в областите извън точките на кръстосване.



Така окончателно се получава потомство

$$\begin{aligned} d_1 &= \text{"1257 | 364 | 8"} \\ d_2 &= \text{"8321 | 567 | 4"} \end{aligned}$$

Мутация

Размяна на два случайно избрани знака от съответния низ

“153264798” \rightarrow “153864792”

Използвана литература:

1. Holland, John, „Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence“, University of Michigan Press, 1975.
2. Holland, John, „Genetic Algorithms and Adaptation“, Adaptive Control of Ill-Defined Systems, pp 317-333, Department of Computer and Communication Sciences, University of Michigan, US, 1984
3. Минчев, Ч., „Изкуствен интелект – теория и приложение при разпознаване на радиолокационни изображения“- Монография, ISBN 978-619-7531-15-2, Шумен, 2021
4. Иванова Д., М. Попова, Н. Вълков, Приложение на генетичен алгоритъм за настройка на ПИД регулатори, 2009, НАУЧНИ ТРУДОВЕ НА РУСЕНСКИЯ УНИВЕРСИТЕТ - 2009, том 48, серия 3.1

----- www.eufunds.bg -----

Проект BG05M2OP001-2.016-0003 „Модернизация на Национален военен университет "В. Левски" - гр. Велико Търново и Софийски университет "Св. Климент Охридски" - гр. София, в професионално направление 5.3 Компютърна и комуникационна техника“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез

Европейските структурни и инвестиционни фондове.