

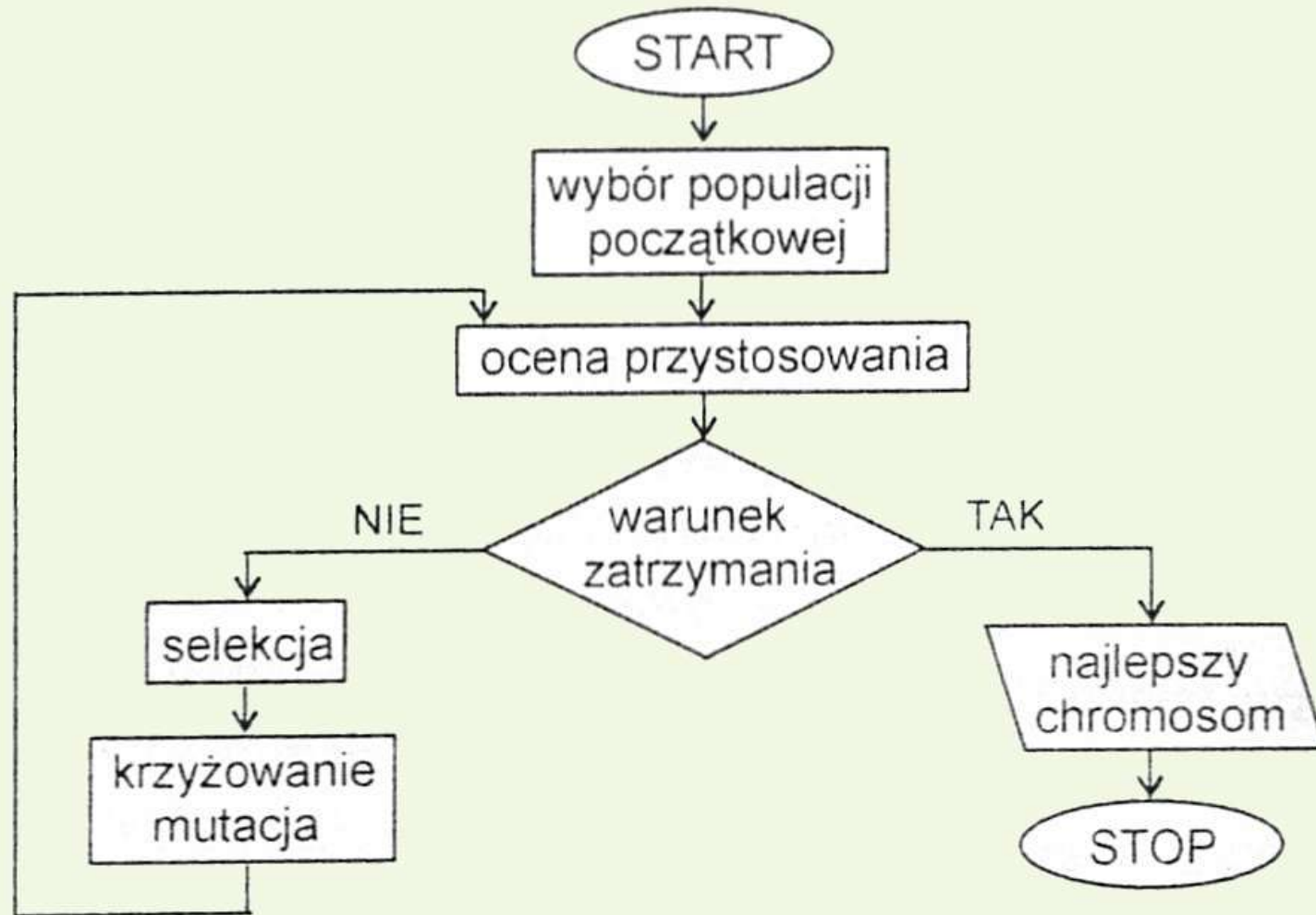
Algorytmy ewolucyjne

wprowadzenie

Historia

- Podstawowy algorytm genetyczny został wprowadzony przez Johna Hollanda (Uniwersytet Michigan) i rozwinięty dzięki jego pracom z lat 60 i 70.
- Celem było stworzenie programu komputerowego rozwiązującego problemy w sposób naśladujący naturalny przebieg ewolucji
- Algorytmy genetyczne wykorzystują mechanizmy naturalnej ewolucji (selekcja, przetrwanie osobników najlepiej przystosowanych, reprodukcja)
- Idea krzyżowania i mutacji chromosomów.
- W symulacjach ewolucji chromosomy reprezentowane są przez ciągi binarne

Podstawowy algorytm genetyczny (PAG)



Schemat blokowy algorytmu genetycznego

Znaczenia, pochodzących z biologii i genetyki, pojęć w algorytmie

chromosom	łańcuch, ciąg uporządkowany genów; zakodowana postać potencjalnego rozwiązania (punktu przestrzeni poszukiwań)
gen	pojedynczy element chromosomu
allele	wartość genu; w PAG równa 0 lub 1
locus	pozycja (miejsce) genu w chromosomie
genotyp	zespół chromosomów danego osobnika; w PAG genotyp stanowi jeden chromosom
osobnik	potencjalne rozwiązanie (punkt przestrzeni poszukiwań); w postaci genotypu lub fenotypu
fenotyp	odpowiednik genotypu, czyli osobnik w formie nie zakodowanej; punkt przestrzeni poszukiwań, którego zakodowaną postacią jest chromosom (genotyp)
populacja	zbiór osobników (chromosomów) o określonej liczebności
przystosowanie	przystosowanie osobników w populacji, oceniane za pomocą tzw. <i>funkcji przystosowania</i> (nazywanej też <i>funkcją dopasowania</i> lub <i>funkcją oceny</i>)

Znaczenia, pochodzących z biologii i genetyki, pojęć w algorytmie

chromosom	łańcuch, ciąg uporządkowany genów; zakodowana postać potencjalnego rozwiązania (punktu przestrzeni poszukiwań)
gen	pojedynczy element chromosomu
allel	wartość genu; w PAG równa 0 lub 1
locus	pozycja (miejsce) genu w chromosomie
genotyp	zespół chromosomów danego osobnika; w PAG genotyp stanowi jeden ci
osobnik	potencjalne roz w postaci genot
fenotyp	odpowiednik ge wanej; punkt pr postacią jest chi
populacja	zbiór osobników
przystosowanie	przystosowanie cą tzw. <i>funkcji p</i> sowania lub <i>fun.</i>

selekcja	wybór, na ogół najlepiej przystosowanych osobników, do tzw. <i>populacji rodzicielskiej</i> , złożonej z chromosomów biorących udział w rekombinacji genów.
krzyżowanie	proces rekombinacji genów, prowadzący do powstania nowych chromosomów (<i>potomków</i>) w wyniku wymiany fragmentów chromosomów <i>rodziców</i>
rodzic	chromosom, kojarzony w parę z drugim chromosomem w celu krzyżowania
potomek	chromosom, otrzymany w wyniku krzyżowania pary rodziców
mutacja	proces zmiany pojedynczego genu w chromosomie; w PAG zamiana wartości wybranego genu z 0 na 1 lub odwrotnie

Przykład:

- Dana jest funkcja:

$$f(x) = 2(x^2+1) \quad x = \{1 \dots 127\}$$

- Cel:

Przeszukać przestrzeń,

znaleźć taki x , dla którego wartość funkcji jest największa

Istota algorytmu genetycznego

- Nie przeszukujemy przestrzeni bezpośrednio
- Wybieramy losowo niewielką populację należących do niej punktów
- Populacja jest modyfikowana zgodnie z zasadami podobnymi do tych, jakie kierują procesem naturalnej ewolucji
- W każdej iteracji algorytmu genetycznego, przetwarzana populacja rozwiązań staje się populacją **coraz lepiej przystosowanych osobników**, reprezentujących rozwiązania **coraz bliższe optymalnemu**

Elementy algorytmu:

1. Wybór (losowy) populacji początkowej
2. Ocena przystosowania
3. Selekcja chromosomów
4. Krzyżowanie
5. Mutacja

Elementy algorytmu:

1. Wybór (losowy) populacji początkowej

- Populacja składa się z n punktów przestrzeni poszukiwań X
- Punkty są zakodowane w postaci ciągów binarnych (nazywamy je **chromosomami**)
- Chromosomy oznaczamy: $ch_1 \dots ch_n$
- Populację początkową oznaczamy
 $P(0) = \{ch_1 \dots ch_n\}$

Elementy algorytmu:

1. Wybór (losowy) populacji początkowej

- Populacja składa się z n punktów przestrzeni poszukiwań X
- Punkty są zakodowane w postaci ciągów binarnych (nazywamy je **chromosomami**)
- Chromosomy oznaczamy: ch_1, \dots, ch_n
- Populację początkową oznaczamy $P(0) = \{ch_1, \dots, ch_n\}$

0100110 (38)

1010001 (81)

0111100 (60)

0101100 (44)

0100101 (37)

0101011 (43)

1011110 (94)

0010111 (23)

Elementy algorytmu:

2. Ocena przystosowania

- W podanym przykładzie, ocena jest dokonywana za pomocą **funkcji przystosowania** ($f(x) = 2(x^2+1)$)
- Ocenia się nie bezpośrednio chromosomy, lecz **fenotypy** (osobnik w formie niezakodowanej – punkt przestrzeni)
- Ocenę (**wartość przystosowania**) przypisuje się następnie odpowiadającym im chromosomom

Elementy algorytmu:

2. Ocena przystosowania

	chromosom	fenotyp	wartość przystosowania
- W podanym przykładzie, ocena jest dokonywana za pomocą funkcji przystosowania ($f(x) = 2(x^2+1)$)	0000110	(6)	74
	1011000	(88)	15490
	0100001	(33)	2180
- Ocenia się nie bezpośrednio chromosomy, lecz fenotypy (osobnik w formie niezakodowanej – punkt przestrzeni)	1001000	(72)	10370
	0001000	(8)	130
	1000111	(71)	10084
- Ocenę (wartość przystosowania) przypisuje się następnie odpowiadającym im chromosomom	0001111	(15)	452
	0110101	(53)	5620

Ocena przystosowania służy do tego, aby:

3. Wyłonić najlepszy chromosom (szczególnie w przypadku, gdy oceniana populacja jest już populacją końcową)
5. Dokonać **selekcji chromosomów** na podstawie ich przystosowania w przypadku gdy oceniana populacja nie jest populacją końcową (populacja bieżąca danej iteracji algorytmu genetycznego)

Elementy algorytmu:

2. Selekcja chromosomów

- Selekcji dokonuje się, gdy warunek zatrzymania nie jest spełniony (populacja nie jest popul. końcową)
- W wyniku selekcji z bieżącej populacji $P(k)$ tworzona jest **populacja rodzicielska** $M(k)$
- W podstawowej wersji algorytmu selekcja chromosomów odbywa się za pomocą metody *koła ruletki*

Elementy algorytmu:

2. Selekcja chromosomów. Metoda koła ruletki.

- Selekcja dokonuje się, poprzez wybór chromosomów, którym na kole (koło ruletki) przydzielono sektory proporcjonalne do wartości przystosowania
- Większa wartość przystosowania = częstszy wybór do populacji rodzicielskiej
- Lepiej przystosowane chromosomy mogą być wybierane wielokrotnie
- Skutek: miejsce „słabszych” zajmują „mocniejsi”

Elementy algorytmu:

2. Selekcja chromosomów. Metoda koła ruletki.

Wielkość sektorów na *kole ruletki* przydzielane są według następujących wzorów:

Wielkość procentowa sektorów:

$$v(ch_i) = p_s(ch_i) \cdot 100\%$$

Prawdopodobieństwo selekcji chromosomu:

$$p_s(ch_i) = \frac{F(ch_i)}{\sum_{i=1}^n F(ch_i)}$$

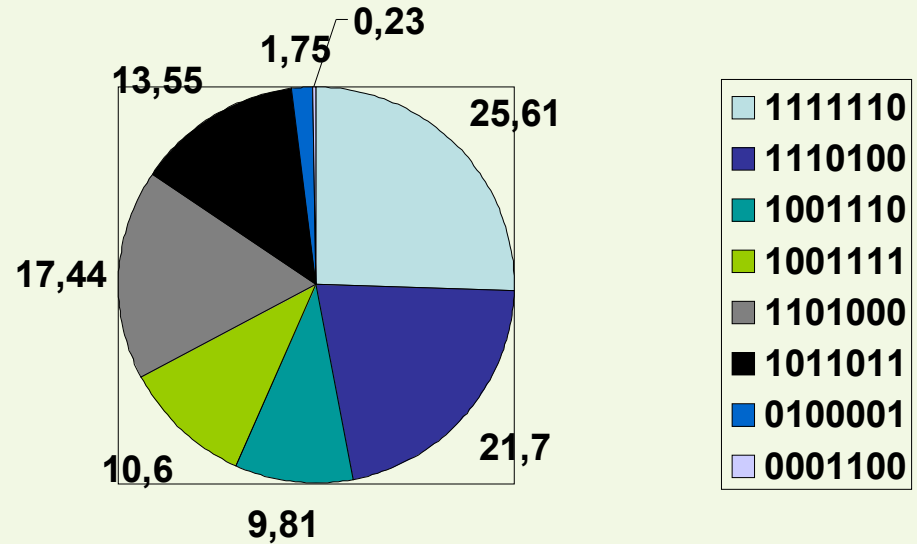
Średnie przystosowanie populacji:

$$F\{P\} = \frac{\sum_{i=1}^n F(ch_i)}{n}$$

Elementy algorytmu:

2. Selekcja chromosomów. Metoda koła ruletki.

1111110 (126)	31754	25,61%
1110100 (116)	26914	21,70%
1001110 (78)	12170	9,81%
1001111 (79)	12484	10,6%
1101000 (104)	21634	17,44%
1011011 (91)	16564	13,35%
0100001 (33)	2180	1,75%
0001100 (12)	290	0,23%



Elementy algorytmu:

3. Operator genetyczny: krzyżowanie

Dwa etapy krzyżowania:

- I. Chromosomy w sposób losowy kojarzone są w pary
- II. Każda para przechodzi krzyżowanie, tworząc chromosomy zwane potomkami

Krzyżowanie par:

1. wybierany jest losowo punkt krzyżowania (**locus**)
2. Wymiana odpowiednich części łańcucha między rodzicami

Elementy algorytmu:

3. Operator genetyczny: krzyżowanie

$$\left. \begin{array}{l} ch_3 = [1 \ 1 \mid 1 \ 0 \ 1] \\ ch_4 = [1 \ 0 \mid 1 \ 0 \ 1] \end{array} \right\} \xrightarrow{\text{krzyżowanie}} \begin{array}{l} ch_3 = [1 \ 1 \ 1 \ 0 \ 1] \\ ch_4 = [1 \ 0 \ 1 \ 0 \ 1] \end{array}$$

locus=2

$$\left. \begin{array}{l} ch_4 = [1 \mid 0 \ 1 \ 0 \ 1] \\ ch_5 = [1 \mid 0 \ 0 \ 1 \ 1] \end{array} \right\} \xrightarrow{\text{krzyżowanie}} \begin{array}{l} ch_6 = [1 \ 0 \ 0 \ 1 \ 1] \\ ch_7 = [1 \ 0 \ 1 \ 0 \ 1] \end{array}$$

locus=1

$$\left. \begin{array}{l} ch_5 = [1 \ 0 \ 0 \mid 1 \ 1] \\ ch_8 = [1 \ 1 \ 1 \mid 0 \ 1] \end{array} \right\} \xrightarrow{\text{krzyżowanie}} \begin{array}{l} [1 \ 0 \ 0 \ 0 \ 1] \\ [1 \ 1 \ 1 \ 1 \ 1] \end{array}$$

locus=3

Elementy algorytmu:

3. Operator genetyczny: mutacja

- Zmienia wartość wybranego losowo genu w chromosomie na przeciwny (1 na 0, 0 na 1)
- Mutacja zachodzi bardzo rzadko – prawdopodobieństwo mutacji jest zwykle dużo mniejsze niż krzyżowania
- **Celem mutacji jest wprowadzenie różnorodności populacji**

$$ch_1 = [1 \ 0 \ 1 \ 0 \ 1] \xrightarrow{\text{mutacja}} ch_1' = [1 \ 0 \ 1 \ 1 \ 1]$$

locus=4

Modyfikacje PAG:

Inne metody selekcji:

- Strategia elitarna
- Selekcja turniejowa
- Selekcja rankingowa

Elementy algorytmu:

1. Wybór populacji początkowej
2. Ocena przystosowania
3. Selekcja
4. Krzyżowanie
5. Mutacja

Inne metody krzyżowania:

- Krzyżowanie dwupunktowe
- Krzyżowanie wielopunktowe
- Krzyżowanie równomierne

Inne metody kodowania:

- Kodowanie rzeczywiste
- Kodowanie logarytmiczne

Inne metody mutacji

Inne modyfikacje PAG

Inne metody selekcji: Strategia elitarna

- Zachowuje najlepszego osobnika w populacji bieżącej i wprowadza go bez zmian do następnej populacji

Zalety:

- Pomaga uniknąć przedwczesnej zbieżności
- Przyspiesza proces poszukiwania optymalnego rozwiązania

Inne metody selekcji: Selekcja turniejowa

- Bardzo często zastępuje metodę ruletki
- Dzielenie populacji na grupy, „rozgrywanie turnieju” pomiędzy osobnikami z poszczególnych grup. Do populacji rodzicielskiej wybierane są najlepsze osobniki z każdej grupy
- ST można dokonywać w sposób deterministyczny (z prawdopodobieństwem równym 1) lub losowo ($p < 1$)
- Populację można dzielić na grupy o różnej liczebności (zwykle po 2 lub 3 osobniki)

Zalety:

- Może być stosowana zarówno do problemów maksymalizacji, jak też minimalizacji
- Doświadczenie wykazuje znacznie lepsze działania AG, w którym zamiast metody ruletki dokonuje się selekcji turniejowej
- Selekcję turniejową można wykorzystywać w zadaniach optymalizacji wielokryterialnej

Inne metody selekcji: Selekcja rankingowa

- Polega na wyborze osobników zgodnie z przypisanymi im rangami
- Osobnikom nadawana jest ranga (zależna od wartości przystosowania)
- Korzystając z takiej „listy rankingowej” definiuje się funkcję (np. liniową) określającą liczbę wybieranych kopii chromosomów w zależności od ich rangi. Na podstawie tej funkcji realizowany jest algorytm selekcji

Zalety:

- Podobnie jak selekcja turniejowa, odpowiednia do problemów maksymalizacji i minimalizacji
- Nie posiada wady metody ruletki dotyczącej konieczności skalowania funkcji przystosowania

Inne metody krzyżowania: krzyżowanie wielopunktowe

- Wybierane są dwa lub więcej punkty krzyżowania chromosomów
- Usprawnia proces krzyżowania w przypadku korzystania z długich chromosomów

$$\begin{array}{c} [1 \ 1 \mid 1 \ 0 \ 1 \mid 1] \\ [1 \ 0 \mid 1 \ 1 \ 0 \mid 0] \end{array} \left. \vphantom{\begin{array}{c} [1 \ 1 \mid 1 \ 0 \ 1 \mid 1] \\ [1 \ 0 \mid 1 \ 1 \ 0 \mid 0] \end{array}} \right\} \begin{array}{c} \xrightarrow{\text{krzyżowanie}} \\ \text{dwupunktowe} \end{array} \begin{array}{c} [1 \ 1 \ 1 \ 1 \ 0 \ 1] \\ [1 \ 0 \ 1 \ 0 \ 1 \ 0] \end{array}$$

locus=2 i 5

Inne metody krzyżowania: krzyżowanie równomierne

- Wylosowanie wzorca, określającego które geny potomków są dziedziczone od każdego z rodziców
- Wzorzec jest łańcuchem binarny, w którym wartości 1 wskazują pozycje (locus) w chromosomie rodzica 1, a wartości 0 odpowiadają pozycjom w chromosomie rodzica 2
- Taka kombinacja tworzy potomka 1 – analogicznie powstaje potomek 2

$$\begin{array}{l} [1 \ 1 \ 1 \ 0 \ 1 \ 1] \\ [1 \ 0 \ 1 \ 1 \ 0 \ 0] \end{array} \left. \vphantom{\begin{array}{l} [1 \ 1 \ 1 \ 0 \ 1 \ 1] \\ [1 \ 0 \ 1 \ 1 \ 0 \ 0] \end{array}} \right\} \begin{array}{c} \xrightarrow{\text{krzyżowanie}} \\ \xrightarrow{\text{równomierne}} \end{array} \begin{array}{l} [1 \ 0 \ 1 \ 0 \ 0 \ 1] \\ [1 \ 1 \ 1 \ 1 \ 1 \ 0] \end{array}$$

locus: 1 2 3 4 5 6

wzorzec: 1 0 0 1 0 1

Inne metody mutacji

- Operatory mutacji, podobnie jak operatory krzyżowania określa się w zależności od rozwiązywanego problemu
- Jeżeli zamiast reprezentacji binarnej stosowane jest kodowanie rzeczywiste – należy zmodyfikować operator mutacji. Zwykle zmienia się wartości rzeczywistą wybranego genu na inną wartość w sposób losowy (np. korzystając z rozkładu normalnego)
- Zupełnie inaczej definiuje się operator mutacji dla zagadnień kombinatorycznych, gdy chromosomy stanowią permutację genów o wartościach rzeczywistych

Inne sposoby kodowania

Dlaczego?

- W przypadku zadania z wieloma zmiennymi, stosując kodowanie standardowe, otrzymujemy bardzo długie chromosomy. Ich długość znacznie się powiększa ze wzrostem wymiarowości problemu.
- Przykład: funkcja 20 zmiennych, każda zmienna reprezentowana przez 20 genów = 400 genów w chromosomie. A to nie jest szczególnie wyśrubowany przykład!
- Przetwarzanie długich chromosomów wydłuża czas działania algorytmu, przez co utrudnia, a czasami nawet uniemożliwia rozwiązanie problemu

Alternatywa?

- Kodowanie **logarytmiczne**, kodowanie **rzeczywiste**

Inne sposoby kodowania: *kodowanie logarytmiczne*

- Pozostaje przy notacji binarnej genów
- Skrócenie długości chromosomów dzięki zastosowaniu funkcji wykładniczej (ekwipotencjalnej)
- W tym kodowaniu: 1-szy gen chromosomu reprezentuje znak liczby wyrażonej za pomocą funkcji wykładniczej, 2-gi – znak wykładnika, a pozostała część stanowi łańcuch kodowy (binarny) wykładnika
- Oznaczmy wymienione części chromosomu przez a,b,bin to chromosom [a b bin] reprezentuje liczbę:

$$(-1)^b e^{(-1)^a [bin]_{10}} \quad \text{gdzie } [bin]_{10} \text{ jest wartością dziesiętną wykładnika}$$

- Stosując tą metodę kodowania można za pomocą chromosomów od dł. 5 zapisać liczbę z przedziału $[-e^7, e^7]$ z rozdzielczością do e^{-7}

Inne sposoby kodowania: *kodowanie logarytmiczne*

- *Przykład 1:*

Chromosom [01101] reprezentuje liczbę:

$$x_1 = (-1)^0 e^{(-1)^1 [101]_{10}} = e^{-5} = 0,0067379$$

- *Przykład 2:*

Chromosom [01011] reprezentuje liczbę

$$x_2 = (-1)^1 e^{(-1)^0 [011]_{10}} = e^{-3} = -20,0855$$

Inne sposoby kodowania: *kodowanie rzeczywiste*

- Geny w chromosomach przyjmują wartości rzeczywiste
- Stosowane często w przypadku wielowymiarowych przestrzeni potencjalnych rozwiązań
- Allele (wartości genów) są wtedy liczbami równymi składowym wektorów odpowiadających punktom tej wielowymiarowej przestrzeni. Zatem geny posiadają postać fenotypów.
- Operator krzyżowania, poprzez wymianę genów między chromosomami, tworzy nowe punkty przestrzeni rozwiązań, czyli generuje nowe osobniki.
- Zaletą jest znaczne skrócenie chromosomów, „wadą” znaczne odejście od idei PAG

Inne modyfikacje

- Algorytmy niszowe, algorytmy do optymalizacji wielokryterialnej, algorytmy do optymalizacji z ograniczeniami
- Techniki stosowane w ewolucyjnym podejściu znacznie odbiegające od PAG (ogólniejsza nazwa: algorytmy ewolucyjne)
- Algorytmy ewolucyjne = wszystkie metody wykorzystujące ideę symulowanej ewolucji, także te nie będące modyfikacją PAG, lecz rozwijane niezależnie.

Inne algorytmy ewolucyjne

Niezależnie od PAG rozwinęły się inne metody symulowanej ewolucji.

Należą do nich:

- **Strategie ewolucyjne**
- **Programowanie ewolucyjne**
- **Programowanie genetyczne**

Strategie ewolucyjne (1+1) ($\mu+\lambda$) (μ,λ)

Strategia (1+1)

- Przetwarzany jest tylko jeden chromosom
- Kluczową rolę odrywa chromosom
- Polega ona na losowej modyfikacji wartości każdego genu chromosomu poprzez dodanie odpowiedniej liczby zgodnie z rozkładem normalnym
- W wyniku tak określonej mutacji (nazwanej tu **perturbacją**) tworzony jest nowy chromosom
- W każdej iteracji algorytmu porównywane są wartości przystosowania obu chromosomów i do następnej generacji wybierany jest ten o większej wartości przystosowania
- Strategie ($\mu+\lambda$) (μ,λ) są uogólnieniem strategii (1+1) – zamiast jednego chromosomu mamy do czynienia z populacją złożoną z μ osobników

Strategie ewolucyjne (1+1) ($\mu+\lambda$) (μ,λ)

Strategia ($\mu+\lambda$)

- Z populacji złożonej z μ osobników tworzy się populację potomków, zawierającą λ osobników.
- Dokonuje się tego poprzez wielokrotne losowanie (ze zwracaniem) osobnika z populacji o liczebności μ
- Kopie wylosowanych osobników umieszcza się w populacji pomocniczej – populacji rodzicielskiej
- Populacja rodzicielska poddawana jest następnie operacjom mutacji i krzyżowania
- W wyniku tej operacji powstaje populacja potomków o liczebności λ
- Tę populację łączy się z poprzednią otrzymując $\mu+\lambda$ osobników
- Nowa populacja w w kolejnej generacji algorytmu zawiera μ najlepszych osobników z populacji o liczebności $\mu+\lambda$

Strategie ewolucyjne (1+1) ($\mu+\lambda$) (μ,λ)

Strategia (μ,λ)

- Strategia (μ,λ) różni się od strategii ($\mu+\lambda$) tylko tym, że nową populację, w kolejnej generacji tworzy się na podstawie populacji potomków liczącej λ osobników, zapominając o μ osobnikach rodzicielskich
- W strategii (μ,λ) każdy osobnik istnieje w populacji tylko podczas jednej generacji, w odróżnieniu od strategii (μ,λ), gdzie osobniki mogą pozostawać w populacji przez wiele generacji algorytmu

Strategie ewolucyjne (1+1) ($\mu+\lambda$) (μ,λ)

Podsumowanie

- W strategiach (μ,λ), ($\mu+\lambda$) oprócz operatora mutacji wprowadzono także operator krzyżowania
- Operatory mutacji i krzyżowania są zdefiniowane inaczej niż w PAG
- Inny jest także sposób reprezentacji osobników w postaci chromosomów
- Strategie (μ,λ), ($\mu+\lambda$) charakteryzują się tzw. mechanizmem adaptacji zasięgu mutacji, czego nie posiada strategia (1+1), ani też jej modyfikacja w postaci strategii (1+ λ), w której tworzonych jest więcej niż jeden potomków.
- W tych wcześniejszych strategiach zasięg mutacji – istotny z punktu widzenia zbieżności algorytmu – był ustalany poprzez dobór wartości odpowiedniego współczynnika, zgodnie z ustaloną regułą
- We wszystkich strategiach ewolucyjnych mutacja jest najważniejszym operatorem genetycznym. Razem z selekcją decyduje o działaniu algorytmu i ma największy wpływ na jego zbieżność

Programowanie ewolucyjne

- Zostało wprowadzone w latach 60 w USA przez Lawrence Fogela
- Fogel stosował tę ewolucyjną metodę w odniesieniu do automatów skończonych
- Działanie automatów było oceniane przez funkcję przystosowania, a poprzez mutację dokonywała się ewolucja osobników
- W latach 90 David Fogel dokonał wielu modyfikacji, upodabniając tę metodę do strategii ewolucyjnych
- W literaturze znajdujemy opis takich zastosowań programowania ewolucyjnego jak modelowanie i identyfikacja systemów, sterowanie, prognozowanie, uczenie sieci neuronowych, rozpoznawanie obrazów, problemy kombinatoryczne

Programowanie genetyczne

- Jest stosunkowo nową odmianą technik ewolucyjnych rozwiniętą na początku lat 90
- Dotyczy ewolucji programów komputerowych pisanych wcześniej w języku LISP, a obecnie również w takich językach jak C.
- Chromosom jest reprezentowany w postaci drzewa, składającego się z węzłów i krawędzi. Za pomocą takiej struktury drzewiastej przedstawia się kod programu
- Występują operatory krzyżowania i mutacji, chromosomy są oceniane na podstawie funkcji przystosowania
- W efekcie otrzymywany jest najlepszy chromosom, reprezentujący najlepszy kod programu
- PG znalazło też zastosowanie w innych problemach, gdzie można wykorzystać drzewiastą strukturę chromosomów, takich jak: synteza drzew decyzyjnych, projektowanie systemów elektronicznych czy poszukiwanie optymalnej struktury sieci neuronowej

Zastosowanie algorytmów genetycznych i ewolucyjnych

- Znajdują zastosowanie w matematyce, medycynie, technice, ekonomii itp.
- Główny, lecz nie jedyny, obszar ich zastosowań stanowi optymalizacja
- Praktyczne zastosowania: wytyczanie trasy połączeń kablowych, sterowania, zadania transportowe, optymalizacja wag sieci neuronowych
- Gry komputerowe. Już w 1967 roku Begley opracował adaptacyjny program do gry w 6 pionków.
- Cavicchio w pracy doktorskiej, z 1970 roku, użył AG do rozwiązania dwóch problemów: wyboru podprogramu i rozpoznawania postaci (ściślej do zaprojektowania zestawu detektorów cech dla urządzenia rozpoznającego)
- System analizy obrazów medycznych za pomocą cyfrowej angiografii różnicowej (Fitzpatrick, Grefenstette, Van Gucht)
- Planowanie optymalnej ścieżki robota z omijaniem przeszkód, konstrukcja ewolucyjnych robotów
- Zagadnienia kombinatoryczne (problem komiwojażera, harmonogramy, planowanie zakupów itp...)