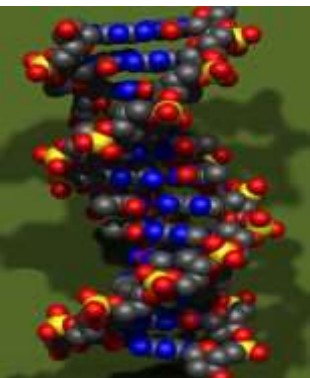




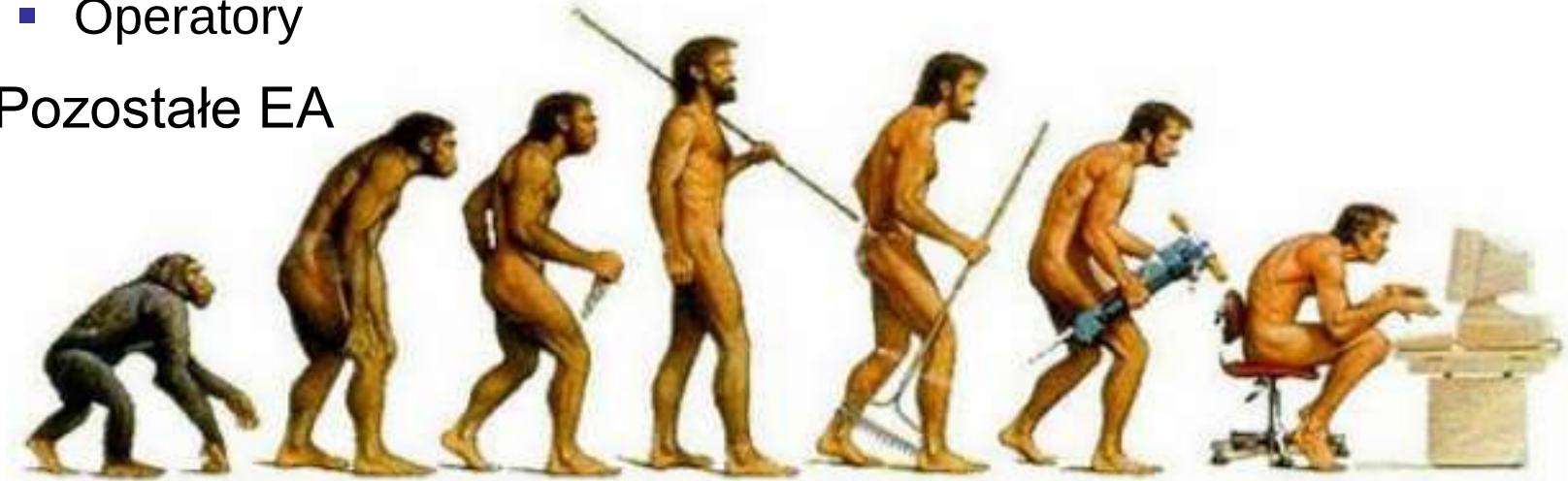
# Algorytmy ewolucyjne

## Część II

Inteligencja obliczeniowa

# Treść wykładu

- ❑ Zastosowania
- ❑ Praktyczne aspekty GA
  - Reprezentacja
  - Funkcja dopasowania
  - Zróżnicowanie dopasowania
  - Selekcja
  - Operatory
- ❑ Pozostałe EA



# EA - zastosowania

## ❑ Ekstrema funkcji numerycznych

Tradycyjne zastosowanie GA, szczególnie dla trudnych, nieciągłych funkcji, o wielu ekstremach.

## ❑ Rozpoznawanie obrazów

W kryminalistyce - do rozpoznawania twarzy przestępców. GA losowo generuje serię twarzy, spośród których świadek wybiera najbardziej podobne do podejrzanych. Poprzez krzyżówki i mutacje otrzymujemy pokolenia twarzy coraz bardziej zbliżone do poszukiwanej. Tutaj rolę funkcji dopasowania spełnia świadek.

# EA - zastosowania

## ❑ Optymalizacja kombinatoryczna

Zagadnienia te obejmują problemy rozmieszczenia dyskretnych obiektów w czasie i/lub przestrzeni.

Najszerzej znanym jest **problem komiwojażera** (TSP).

**Problem załadunku** (BPP) ma duże znaczenie praktyczne.

**Szeregowanie zadań produkcyjnych.**

**Problem transportowy.**

**Kontrola lotów.**

**Planowanie ruchu pacjentów w szpitalu.**

**Podział godzin w szkole.**

Zagadnienia te wymagają stosowania niestandardowych reprezentacji i operatorów.

# EA - zastosowania

## □ Projektowanie

Zagadnienia te można traktować jako połączenie optymalizacji kombinatorycznej i funkcyjnej. Znane zastosowania dotyczą:

- projektowanie mostów,
- optymalizacja dyszy węża ppoż.,
- projektowanie struktury sieci neuronowej,
- projektowanie sieci elektrycznej,
- projektowanie sieci zaopatrzenia w wodę.

# Praktyczne aspekty EA

- ❑ Projektując zastosowanie GA nie wystarczy posłużyć się standardowym schematem. Badania empiryczne dowiodły, że krytyczne są następujące elementy:
  - reprezentacja,
  - postać funkcji dopasowania,
  - zróżnicowanie dopasowania,
  - technika wyboru rodziców,
  - technika wyboru pokolenia potomków.

# Reprezentacja

- ❑ Dostosowana do problemu.
- ❑ Powinna kodować jak najwięcej informacji specyficznych dla problemu.
- ❑ Może wymagać specjalizowanych operatorów.
- ❑ Dopasowanie osobnika jest obliczone na podstawie genotypu.
- ❑ Najczęściej jest to wprost funkcja celu.
- ❑ Często funkcja zawiera ograniczenia nałożone na zmienne np. w postaci funkcji kary.

# Reprezentacja

## Reprezentacja binarna – kod Gray'a

- ❑ Podstawową cechą wyrazów kodu Gray'a jest to, iż każde dwa kolejne wyrazy (również pierwszy i ostatni) różnią się od siebie stanem tylko jednego bitu.
- ❑ Konwersja liczby zapisanej w kodzie binarnym na odpowiednik w kodzie Graya:
  - przesunąć liczbę w postaci binarnej o jeden bit w prawo (podzielić przez 2),
  - wykonać operację XOR na odpowiednich bitach liczby binarnej i wyniku dzielenia tej liczby przez 2.



# Reprezentacja

## Reprezentacja binarna – kod Gray’a

|               | 3 | 2 | 1 | 0 | dziesiętna |
|---------------|---|---|---|---|------------|
| binarna       | 1 | 1 | 0 | 0 | 12         |
| Przes.w prawo | 0 | 1 | 1 | 0 |            |
| Gray          | 1 | 0 | 1 | 0 |            |

|               |   |   |   |   |    |
|---------------|---|---|---|---|----|
| binarna       | 1 | 0 | 1 | 1 | 11 |
| Przes.w prawo | 0 | 1 | 0 | 1 |    |
| Gray          | 1 | 1 | 1 | 0 |    |

|               |   |   |   |   |    |
|---------------|---|---|---|---|----|
| binarna       | 1 | 0 | 1 | 0 | 10 |
| Przes.w prawo | 0 | 1 | 0 | 1 |    |
| Gray          | 1 | 1 | 1 | 1 |    |

# Reprezentacja

## Inne reprezentacje

- Dla prostych problemów kolejnościowych naturalną reprezentacją jest lista uporządkowana:  
1 3 2 4 7 5 6 8
- Podstawowy problem - nie działają klasyczne operatory.
- Czy ciąg liczb rzeczywistych może być reprezentacją dla takich problemów?

Sekwencja wejściowa

| 1        | 2        | 3        | 4       | 5        | 6        | 7        | 8        |
|----------|----------|----------|---------|----------|----------|----------|----------|
| 0,485223 | 0,681597 | 0,489335 | 0,17524 | 0,441197 | 0,699367 | 0,114374 | 0,370359 |

Sekwencja po sortowaniu

| 7        | 4       | 8        | 5        | 1        | 3        | 2        | 6        |
|----------|---------|----------|----------|----------|----------|----------|----------|
| 0,114374 | 0,17524 | 0,370359 | 0,441197 | 0,485223 | 0,489335 | 0,681597 | 0,699367 |

# Reprezentacja

## Inne reprezentacje

Prosta krzyżówka

Rodzice

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 4 | 8 | 5 | 1 | 3 | 2 | 6 |
|---|---|---|---|---|---|---|---|

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 6 | 2 | 3 | 4 | 8 | 7 | 1 | 5 |
|---|---|---|---|---|---|---|---|

Dzieci

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 4 | 8 | 5 | 8 | 7 | 1 | 5 |
|---|---|---|---|---|---|---|---|

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 6 | 2 | 3 | 4 | 1 | 3 | 2 | 6 |
|---|---|---|---|---|---|---|---|

Prosta krzyżówka

Rodzice

|          |          |          |         |          |          |          |          |
|----------|----------|----------|---------|----------|----------|----------|----------|
| 0,485223 | 0,681597 | 0,489335 | 0,17524 | 0,441197 | 0,699367 | 0,114374 | 0,370359 |
|----------|----------|----------|---------|----------|----------|----------|----------|

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| 0,754165 | 0,176504 | 0,329154 | 0,329504 | 0,762581 | 0,006147 | 0,611928 | 0,402365 |
|----------|----------|----------|----------|----------|----------|----------|----------|

Dzieci

|          |          |          |         |          |          |          |          |
|----------|----------|----------|---------|----------|----------|----------|----------|
| 0,485223 | 0,681597 | 0,489335 | 0,17524 | 0,762581 | 0,006147 | 0,611928 | 0,402365 |
|----------|----------|----------|---------|----------|----------|----------|----------|

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| 0,754165 | 0,176504 | 0,329154 | 0,329504 | 0,441197 | 0,699367 | 0,114374 | 0,370359 |
|----------|----------|----------|----------|----------|----------|----------|----------|

|          |         |          |          |          |          |          |          |
|----------|---------|----------|----------|----------|----------|----------|----------|
| 6        | 4       | 8        | 1        | 3        | 7        | 2        | 5        |
| 0,006147 | 0,17524 | 0,402365 | 0,485223 | 0,489335 | 0,611928 | 0,681597 | 0,762581 |

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| 7        | 2        | 3        | 4        | 8        | 5        | 6        | 1        |
| 0,114374 | 0,176504 | 0,329154 | 0,329504 | 0,370359 | 0,441197 | 0,699367 | 0,754165 |

# Reprezentacja

## Inne reprezentacje

- ❑ Dla innych problemów szeregowania - ciąg genów:  
1 1 2 3 4 5 : 3 2 1 4 3 : 2 4 2 1 3
- ❑ Dla problemów grupowania chromosom może składać się z części obiektowej i grupowej:  
1 2 1 3 3 3 1 1 2 : 1 2 3  
część grupowa ma zmienną długość.

# Reprezentacja

## Inne reprezentacje

- Reprezentacja macierzowa:  
np. dla problemu komiwojażera:

0 0 0 1 0

1 0 1 1 1

1 0 0 1 1

0 0 0 0 0

1 0 0 1 0

koduje trasę 2 3 5 1 4

- jest w niej dokładnie  $n(n-1)/2$  jedynek,
  - na przekątnej są zera,
  - jeśli  $m_{ij}=1$  i  $m_{jk}=1$ , to  $m_{ik}=1$ .
- Podobna reprezentacja - dla układania podziału godzin.

# Funkcja dopasowania (FF – Fitness Function)

- ❑ Idealna sytuacja to taka, gdy funkcja dopasowania jest gładka i regularna tj. rozwiązania o podobnym dopasowaniu są położone blisko siebie. Najlepiej, gdy funkcja dopasowania odzwierciedla rzeczywistą wartość chromosomu (punktu w przestrzeni przeszukiwań). Nawet jeśli jej obliczenie nie jest zadaniem trywialnym, wiadomo **co** obliczać. Jednak w wielu przypadkach, szczególnie optymalizacji kombinatorycznej, gdzie jest wiele nierówności, wiele chromosomów reprezentuje rozwiązania niedozwolone, które nie mają praktycznej wartości. Przykład: plan lekcji.

# Funkcja dopasowania

- ❑ By EA był efektywny w takich przypadkach, należy w funkcji dopasowania uwzględnić wskazanie, w jakim stopniu błędny chromosom prowadzi nas w kierunku poprawnych rozwiązań. Dobrym sposobem jest użycie **funkcji kary**, która mówi, jak zły jest chromosom i określenie dopasowania jako (wartość-kara). Dobra funkcja kary powinna mówić, jaki jest koszt przekształcenia złego chromosomu w prawidłowy.
- ❑ Jeśli funkcja dopasowania jest trudna do obliczenia można stosować **przybliżone oszacowanie wartości funkcji**. Takie obliczenie jest znacznie szybsze, co pozwala znacząco zwiększyć przeszukiwaną przestrzeń rozwiązań. Przybliżone oszacowanie 10 punktów przestrzeni jest z reguły lepsze niż dokładne obliczenie jednego.  
EA jest techniką na tyle sprytną, że poradzi sobie z szumem wnoszonym przez oszacowanie.

# Zróżnicowanie dopasowania

## ❑ Przedwczesna zbieżność

Dominacja dobrych (ale nie optymalnych) kilku chromosomów powoduje utknięcie w lokalnym ekstremum. Gdyby populacja była nieskończona nie byłoby problemu, w rzeczywistych zastosowaniach należy zmodyfikować sposób wyboru rodziców do reprodukcji. Wyjściem jest kontrola liczby szans na reprodukcję poszczególnych osobników, tak aby była ona ani za duża, ani za mała.

## ❑ Powolne finiszowanie

Odwrotne zagadnienie; po wielu generacjach populacja jest w większości zbieżna, ale globalne maximum nie jest precyzyjnie zlokalizowane. Średnie dopasowanie jest wysokie, a różnica między najlepszym i najgorszym osobnikiem - niewielka. W konsekwencji brak impulsu do skierowania EA ku maksimum. Wyjściem jest rozszerzenie efektywnego zróżnicowania wartości funkcji dopasowania przez jej skalowanie.



# Techniki wyboru rodziców

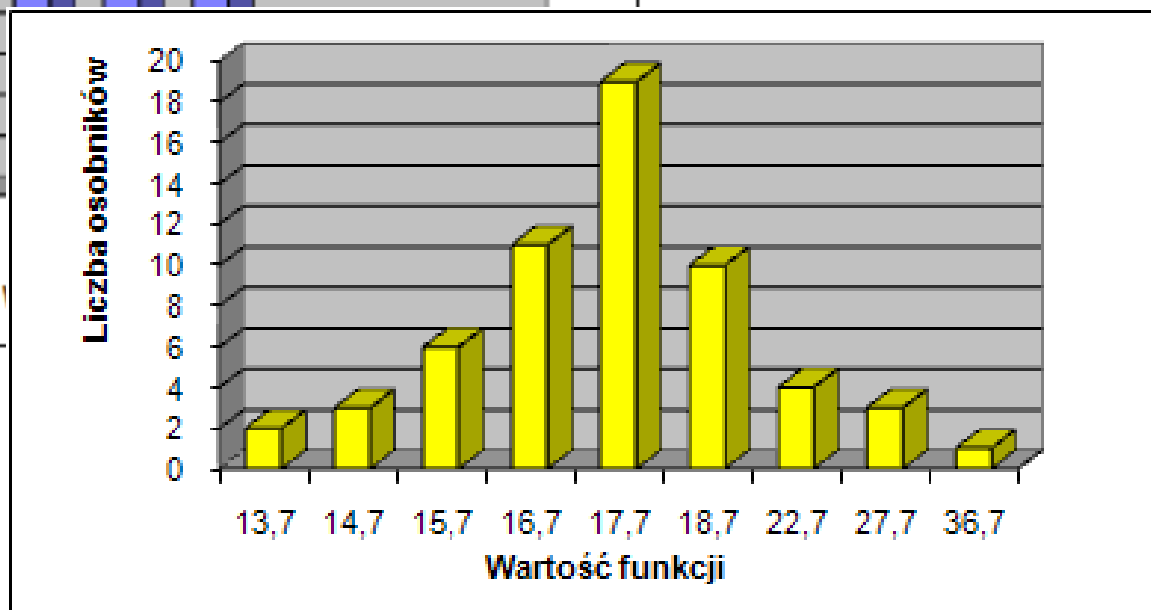
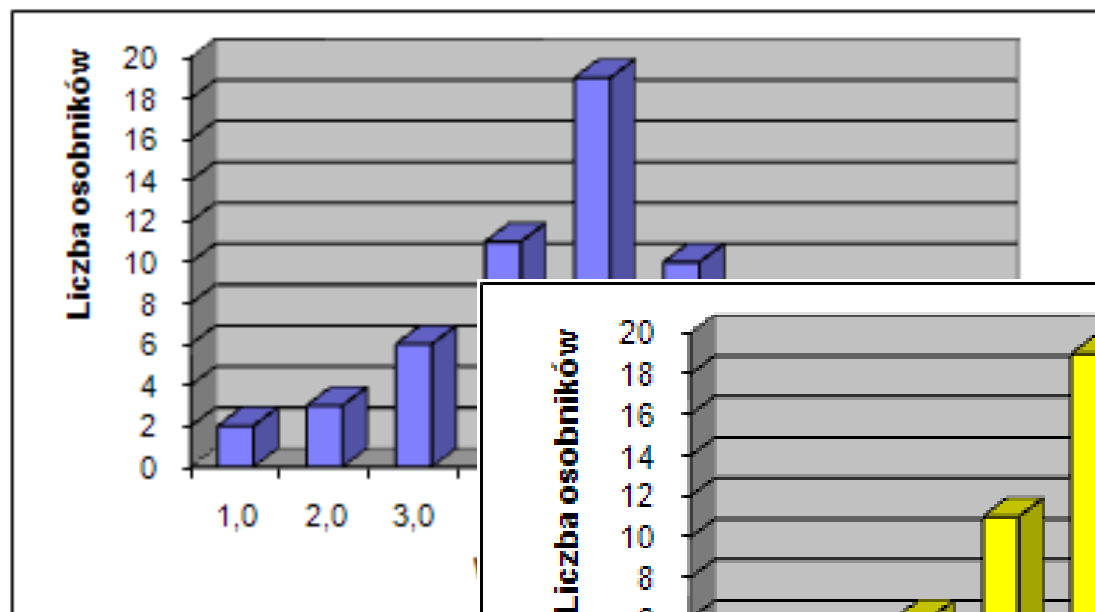
- Standardowo, osobniki są wybierane do krzyżowania w taki sposób, że najlepsze osobniki mają dużą szansę uczestniczyć w nim wielokrotnie, a najgorsze – małą albo zerową. Okazało się, że schemat, w którym wybór jest wprost proporcjonalny do czystej wartości dopasowania, nie zawsze zdaje egzamin.

- **Skalowanie dopasowania**

W technice tej przyjmuje się, że stosunek szans na krzyżowanie najlepszego osobnika do średniej w populacji powinien wynosić 2:1. Osiągamy to przez odjęcie liczby  $= 2 * \text{średnia-max}$  od wartości funkcji dopasowania wszystkich osobników. Taka kompresja zróżnicowania dopasowania zmniejsza przedwczesną zbieżność, ale w przypadku, gdy mamy do czynienia tylko z jednym wybitnym osobnikiem, skalowanie zbyt spłaszcza zmodyfikowaną funkcję dopasowania, co pogarsza efektywność algorytmu.

# Techniki wyboru rodziców

## Skalowanie dopasowania

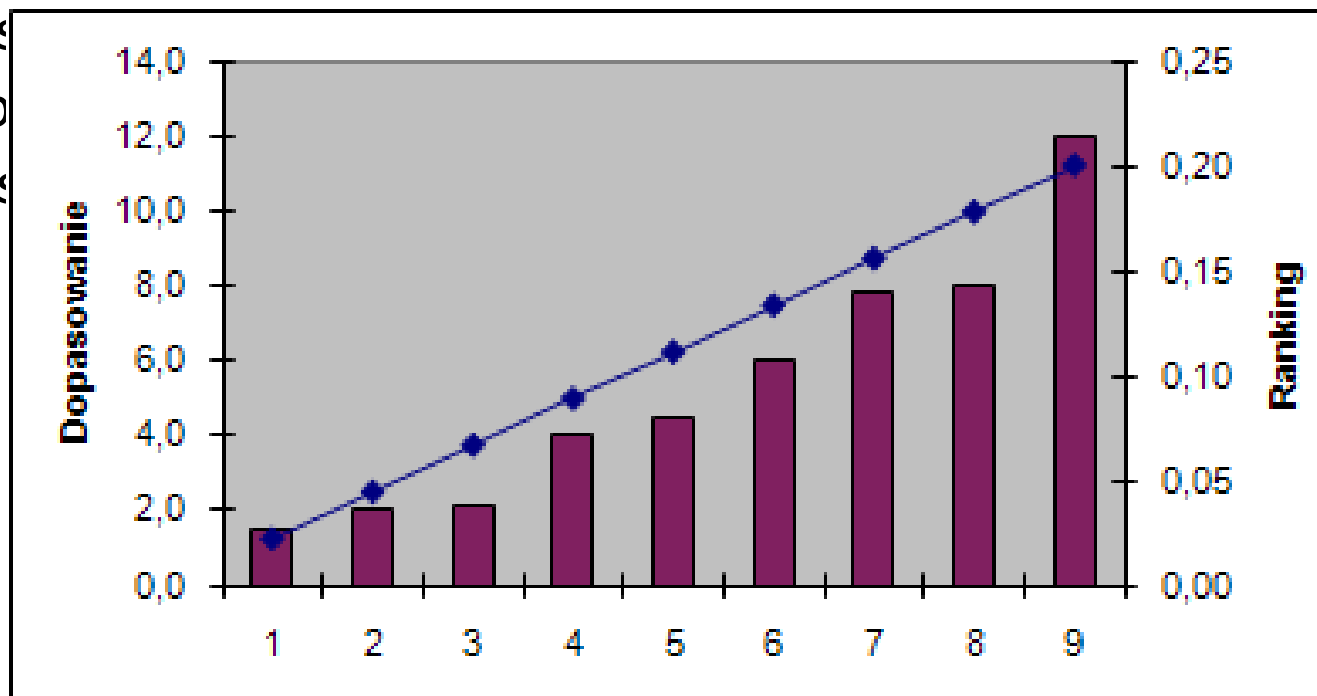


# Techniki wyboru rodziców

## Ranking dopasowania

Osobniki są sortowane wg czystych wartości funkcji, po czym reprodukcyjne dopasowania są przyporządkowywane zgodnie z tym rankingiem. Można to robić liniowo bądź

eksponencyjnie. W tym przykładzie wykorzystano ranking liniowy.



# Techniki wyboru rodziców

## ❑ Selekcja na zasadzie turnieju

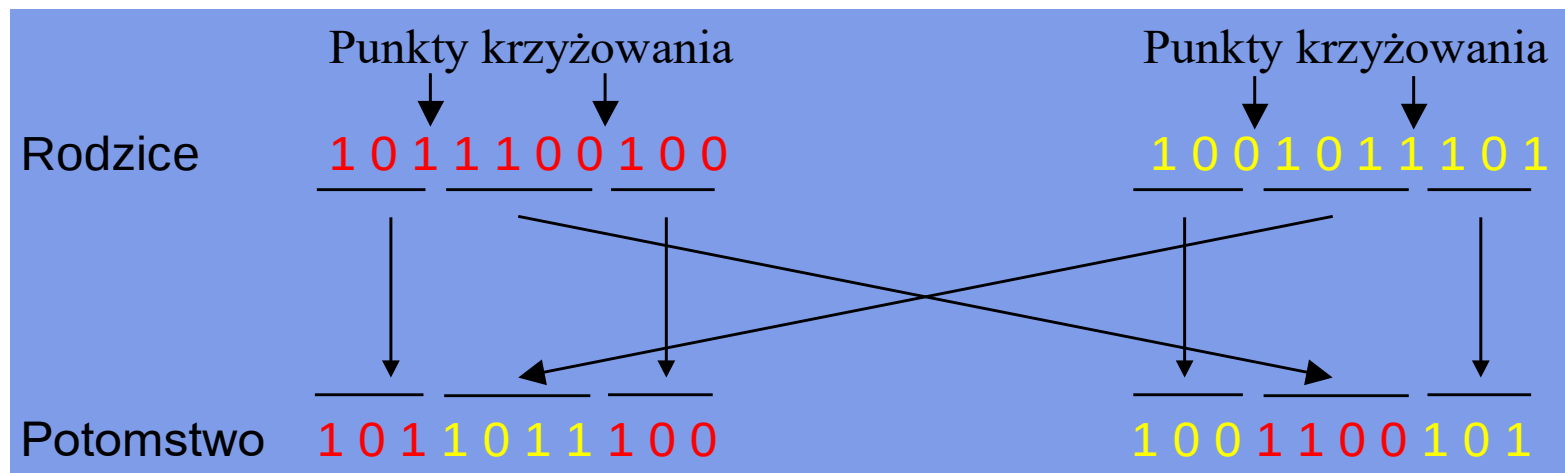
W najprostszej wersji para osobników jest losowana z populacji; osobnik o lepszym dopasowaniu z tej dwójki jest wybierany do krzyżowania. Turniej trwa aż do stworzenia wszystkich par. Możliwa jest również wersja, w której „ściera się” ze sobą  $n$  losowo wybranych osobników. W turnieju probabilistycznym, z pary (lub  $n$ ) wylosowanych osobników wygrywa lepszy z prawdopodobieństwem  $0,5 < p < 1,0$ . Selekcja jest dużo łagodniejsza.

# Technika wyboru potomstwa

- Standardowo, cała populacja rodziców jest wymieniana przez potomstwo. Badania wykazują, że lepiej wymienić tylko część populacji. Ma to uzasadnienie w naturze. Sposób wyboru osobników, które „mają pecha” i zostaną usunięte z populacji może być różny: najczęściej stosuje się wybór losowy lub deterministyczny najgorszych osobników.

# Techniki krzyżowania

- Wielopunktowe krzyżówki służą intensywniejszemu przeszukiwaniu przestrzeni rozwiązań. Szczególnie dobrze działa krzyżówka dwupunktowa.



# Techniki krzyżowania

- ❑ **Krzyżówka ujednolicająca** (ang. uniform crossover) tworzy potomków przez skopiowanie odpowiednich genów z jednego lub drugiego rodzica zgodnie z tworzoną losowo dla każdej pary maską.

|           |                                       |   |   |   |   |   |   |   |   |   |  |  |  |   |   |   |
|-----------|---------------------------------------|---|---|---|---|---|---|---|---|---|--|--|--|---|---|---|
|           | Maska krzyżowania 0 0 1 1 0 0 1 1 0 1 |   |   |   |   |   |   |   |   |   |  |  |  |   |   |   |
| Rodzice   | 1                                     | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |  |  |  | 1 | 0 | 0 |
| Potomstwo | 1                                     | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |  |  |  | 1 | 0 | 0 |

# Techniki krzyżowania

- ❑ Dla niektórych problemów opisane wyżej krzyżówki nie mogą być stosowane. Należą do nich problemy kolejnościowe, gdzie wartości genów z reguły nie są bitami, a wartość funkcji dopasowania zależy od kolejności, w której występują.
- ❑ **Operator PMX** (Partially Matched Crossover) został opracowany dla problemu komiwojażera (Goldberg i Lingle, 1985). Jego działanie polega na wykorzystaniu mapy wymiany, która zawiera pary odpowiadających sobie elementów w sekcji wymiany rodziców.



# Techniki krzyżowania - PMX

- Rozważmy parę rodziców:

$$R1 = (1,2,3 \mid 4,5,6 \mid 7,8,9) \quad \text{i}$$

$$R2 = (2,4,1 \mid 7,8,3 \mid 9,5,6),$$

gdzie geny umieszczone między znakami | oznaczają sekcję kojarzenia, czyli ciąg genów wymienianych między rodzicami w procesie krzyżowania. Mapa wymiany ma postać:

$$M = (4-7, 5-8, 6-3).$$

- Po wymianie sekcji kojarzenia otrzymujemy parę potomków:

$$P1 = (x,x,x \mid 7,8,3 \mid x,x,x) \quad \text{i}$$

$$P2 = (x,x,x \mid 4,5,6 \mid x,x,x).$$

## Techniki krzyżowania - PMX

- Pozostałe geny są uzupełniane z odpowiedniego rodzica, a w przypadku wartości powtarzających się - geny są wstawiane zgodnie z mapą wymiany. W rezultacie otrzymujemy parę potomków:

$P1 = (1, 2, 6, 7, 8, 3, 4, 5, 9) \quad i$

$P2 = (2, 7, 1, 4, 5, 6, 8, 3, 9).$

---

$R1 = (1, 2, 3 \mid 4, 5, 6 \mid 7, 8, 9) \quad i$

$R2 = (2, 4, 1 \mid 7, 8, 3 \mid 9, 5, 6),$

$M = (4-7, 5-8, 6-3).$

# Techniki krzyżowania - OX

- ❑ **Krzyżówka OX (Order Crossover)**. Zaproponował ją Davis (1999); buduje potomstwo przez wybór podciągu z uszeregowania i zachowanie względnej kolejności zadań z drugiego rodzica.

- ❑ Np. dwoje rodziców:

R1 = ( 1 2 3 | 4 5 6 7 | 8 9 )

R2 = ( 4 5 2 | 1 8 7 6 | 9 3 )

Na początku segmenty pomiędzy punktami cięcia są kopiowane do potomków:

P1 = ( x x x | 4 5 6 7 | x x )

P2 = ( x x x | 1 8 7 6 | x x )

## Techniki krzyżowania - OX

- Następnie poczynając od drugiego punktu cięcia pierwszego rodzica, zadania z drugiego rodzica są kopiowane w tej samej kolejności, pomijając symbole już istniejące. Po osiągnięciu końca ciągu kontynuujemy od pierwszej pozycji ciągu.

P1 = ( 2 1 8 | 4 5 6 7 | 9 3 )

P2 = ( 3 4 5 | 1 8 7 6 | 9 2 )

---

R1 = ( 1 2 3 | 4 5 6 7 | 8 9 )

R2 = ( 4 5 2 | 1 8 7 6 | 9 3 )

# Techniki krzyżowania

- ❑ **Operator losowy z uzupełnianiem** MkX (Masked Crossover - Louis i Rawlins, 1991) działający w ten sposób, że losowane są pozycje zadań przenoszonych bezpośrednio z rodziców do potomków, a reszta - jest uzupełniana z drugiego rodzica (z pominięciem zadań powtarzających się). Rozważmy np. parę rodziców:

R1 = (1,2,3,4,5,6,7,8,9)

i

R2 = (2,4,1,7,8,3,9,5,6)

# Techniki krzyżowania

- Niech dla pierwszego rodzica wylosowano niezmiennione pozycje nr 2,4,5,8, a dla drugiego - 1,2,6,9. Zadania na tych pozycjach są przepisywane do potomków:

P1 = (x, 2, x, 4, 5, x, x, 8, x)

R1 = (1, 2, 3, 4, 5, 6, 7, 8, 9)

i

R2 = (2, 4, 1, 7, 8, 3, 9, 5, 6)

P2 = (2, 4, x, x, x, 3, x, x, 6).

- Reszta jest przenoszona z drugiego rodzica dając w rezultacie parę:

P1 = (1, 2, 7, 4, 5, 3, 9, 8, 6)

i

P2 = (2, 4, 1, 5, 7, 3, 8, 9, 6).

# Algorytm genetyczny – przykład TSP

- Znaleźć najkrótszą trasę dla  $n=7$  miast; odległości między miastami wynoszą:

|   | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 |   | 4 | 2 | 5 | 2 | 2 | 1 |
| 2 | 4 |   | 3 | 7 | 4 | 4 | 5 |
| 3 | 2 | 3 |   | 1 | 8 | 5 | 3 |
| 4 | 5 | 7 | 1 |   | 6 | 1 | 4 |
| 5 | 2 | 4 | 8 | 6 |   | 5 | 1 |
| 6 | 2 | 4 | 5 | 1 | 5 |   | 3 |
| 7 | 1 | 5 | 3 | 4 | 1 | 3 |   |

- Założenia:
  - rozmiar populacji  $P = 4$ ;
  - krzyżówka losowa z uzupełnianiem;
  - dobór rodziców zgodnie z funkcją dopasowania (ruletka);
  - prawd. zastosowania operatora krzyżowania  $pc = 0,8$ ;
  - prawd. zastosowania operatora mutacji typu swap  $pm = 0,05$ ;
  - selekcja następnego pokolenia – 50% najlepszych dzieci;
  - warunek zatrzymania.

# Przykład GA – TSP dla $n = 7$ miast

- ❑ Krok 0 – wylosowanie populacji początkowej.
- ❑ Pętla 1 – krok 1: obliczenie funkcji dopasowania osobników.

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) |
|---------|----|----|----|----|----|----|----|-------------|-------------|
| 1       | 4  | 2  | 5  | 3  | 7  | 1  | 6  | 26          | 15          |
| 2       | 7  | 3  | 6  | 1  | 2  | 5  | 4  | 28          | 13          |
| 3       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          |
| 4       | 6  | 7  | 1  | 5  | 3  | 4  | 2  | 26          | 15          |
| suma    |    |    |    |    |    |    |    | 103,0       | 61,0        |
| średnia |    |    |    |    |    |    |    | 25,8        | 15,3        |

- ❑ Pętla 1 – krok 2: wybór rodziców zgodnie z funkcją dopasowania.

| DT( $x_i$ ) | FD( $x_i$ ) | Prawd. | Dystr. |
|-------------|-------------|--------|--------|
| 26          | 15          | 0,246  | 0,246  |
| 28          | 13          | 0,213  | 0,459  |
| 23          | 18          | 0,295  | 0,754  |
| 26          | 15          | 0,246  | 1,000  |
| 103,0       | 61,0        | 1,000  |        |
| 25,8        | 15,3        |        |        |

| L.los | 1. rodzic | L.los | 2. rodzic |
|-------|-----------|-------|-----------|
| 0,176 | 1         | 0,738 | 3         |
| 0,278 | 2         | 0,552 | 3         |
| 0,637 | 3         | 0,883 | 4         |
| 0,924 | 4         | 0,220 | 1         |



# Przykład GA – TSP dla $n = 7$ miast

- ❑ Pętla 1 – krok 3: krzyżówka.
- ❑ Pętla 1 – krok 4: ewaluacja populacji.

| L.los | 1. rodzic | L.los | 2. rodzic |
|-------|-----------|-------|-----------|
| 0,176 | 1         | 0,738 | 3         |
| 0,278 | 2         | 0,552 | 3         |
| 0,637 | 3         | 0,883 | 4         |
| 0,924 | 4         | 0,220 | 1         |

## Rodzice

|       |   |   |   |   |   |   |   |   |
|-------|---|---|---|---|---|---|---|---|
| 0,642 | 1 | 4 | 2 | 5 | 3 | 7 | 1 | 6 |
| 0,117 | 2 | 7 | 3 | 6 | 1 | 2 | 5 | 4 |
| 0,987 | 3 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0,694 | 4 | 6 | 7 | 1 | 5 | 3 | 4 | 2 |

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 3 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 3 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 4 | 6 | 7 | 1 | 5 | 3 | 4 | 2 |
| 1 | 4 | 2 | 5 | 3 | 7 | 1 | 6 |

## Dzieci

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 4 | 1 | 5 | 2 | 7 | 3 | 6 |
| 3 | 2 | 3 | 4 | 1 | 6 | 5 | 7 |
| 5 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7 | 6 | 4 | 1 | 5 | 3 | 7 | 2 |

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 2 | 1 | 4 | 3 | 2 | 5 | 6 | 7 |
| 4 | 7 | 2 | 3 | 4 | 1 | 6 | 5 |
| 6 | 6 | 7 | 1 | 5 | 3 | 4 | 2 |
| 8 | 4 | 1 | 5 | 3 | 7 | 2 | 6 |

# Przykład GA – TSP dla $n = 7$ miast

- ❑ Pętla 1 – krok 3: krzyżówka.
- ❑ Pętla 1 – krok 4: ewaluacja populacji.

| L.los | 1. rodzic | L.los | 2. rodzic |
|-------|-----------|-------|-----------|
| 0,176 | 1         | 0,738 | 3         |
| 0,278 | 2         | 0,552 | 3         |
| 0,637 | 3         | 0,883 | 4         |
| 0,924 | 4         | 0,220 | 1         |

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) |
|---------|----|----|----|----|----|----|----|-------------|-------------|
| 1       | 4  | 1  | 5  | 2  | 7  | 3  | 6  | 25          | 16          |
| 2       | 1  | 4  | 3  | 2  | 5  | 6  | 7  | 22          | 19          |
| 3       | 2  | 3  | 4  | 1  | 6  | 5  | 7  | 22          | 19          |
| 4       | 7  | 2  | 3  | 4  | 1  | 6  | 5  | 22          | 19          |
| 5       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          |
| 6       | 6  | 7  | 1  | 5  | 3  | 4  | 2  | 26          | 15          |
| 7       | 6  | 4  | 1  | 5  | 3  | 7  | 2  | 28          | 13          |
| 8       | 4  | 1  | 5  | 3  | 7  | 2  | 6  | 28          | 13          |
| suma    |    |    |    |    |    |    |    | 196,0       | 132,0       |
| średnia |    |    |    |    |    |    |    | 24,5        | 16,5        |

# Przykład GA – TSP dla $n = 7$ miast

□ Pętla 1 – krok 5: mutacja.

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) | L. losowa |
|---------|----|----|----|----|----|----|----|-------------|-------------|-----------|
| 1       | 4  | 1  | 5  | 2  | 7  | 3  | 6  | 25          | 16          | 0,848     |
| 2       | 1  | 4  | 3  | 2  | 5  | 6  | 7  | 22          | 19          | 0,158     |
| 3       | 2  | 3  | 4  | 1  | 6  | 5  | 7  | 22          | 19          | 0,596     |
| 4       | 7  | 2  | 3  | 4  | 1  | 6  | 5  | 22          | 19          | 0,824     |
| 5       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          | 0,843     |
| 6       | 6  | 7  | 1  | 5  | 3  | 4  | 2  | 26          | 15          | 0,883     |
| 7       | 6  | 4  | 1  | 5  | 3  | 7  | 2  | 28          | 13          | 0,715     |
| 8       | 4  | 1  | 5  | 3  | 7  | 2  | 6  | 28          | 13          | 0,015     |
| suma    |    |    |    |    |    |    |    | 196,0       | 132,0       |           |
| średnia |    |    |    |    |    |    |    | 24,5        | 16,5        |           |

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) |
|---------|----|----|----|----|----|----|----|-------------|-------------|
| 1       | 4  | 1  | 5  | 2  | 7  | 3  | 6  | 25          | 16          |
| 2       | 1  | 4  | 3  | 2  | 5  | 6  | 7  | 22          | 19          |
| 3       | 2  | 3  | 4  | 1  | 6  | 5  | 7  | 22          | 19          |
| 4       | 7  | 2  | 3  | 4  | 1  | 6  | 5  | 22          | 19          |
| 5       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          |
| 6       | 6  | 7  | 1  | 5  | 3  | 4  | 2  | 26          | 15          |
| 7       | 6  | 4  | 1  | 5  | 3  | 7  | 2  | 28          | 13          |
| 8       | 4  | 2  | 5  | 3  | 7  | 1  | 6  | 26          | 15          |
| suma    |    |    |    |    |    |    |    | 194,0       | 134         |
| średnia |    |    |    |    |    |    |    | 24,3        | 16,75       |

# Przykład GA – TSP dla $n = 7$ miast

- Pętla 1 – krok 6: selekcja następnego pokolenia.

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) |
|---------|----|----|----|----|----|----|----|-------------|-------------|
| 2       | 1  | 4  | 3  | 2  | 5  | 6  | 7  | 22          | 19          |
| 3       | 2  | 3  | 4  | 1  | 6  | 5  | 7  | 22          | 19          |
| 4       | 7  | 2  | 3  | 4  | 1  | 6  | 5  | 22          | 19          |
| 5       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          |
| 1       | 4  | 1  | 5  | 2  | 7  | 3  | 6  | 25          | 16          |
| 6       | 6  | 7  | 1  | 5  | 3  | 4  | 2  |             |             |
| 8       | 4  | 2  | 5  | 3  | 7  | 1  | 6  |             |             |
| 7       | 6  | 4  | 1  | 5  | 3  | 7  | 2  |             |             |

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) |
|---------|----|----|----|----|----|----|----|-------------|-------------|
| 1       | 4  | 2  | 5  | 3  | 7  | 1  | 6  | 26          | 15          |
| 2       | 7  | 3  | 6  | 1  | 2  | 5  | 4  | 28          | 13          |
| 3       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          |
| 4       | 6  | 7  | 1  | 5  | 3  | 4  | 2  | 26          | 15          |
| suma    |    |    |    |    |    |    |    | 103,0       | 61,0        |
| średnia |    |    |    |    |    |    |    | 25,8        | 15,3        |

| osobnik | m1 | m2 | m3 | m4 | m5 | m6 | m7 | DT( $x_i$ ) | FD( $x_i$ ) |
|---------|----|----|----|----|----|----|----|-------------|-------------|
| 2       | 1  | 4  | 3  | 2  | 5  | 6  | 7  | 22          | 19          |
| 3       | 2  | 3  | 4  | 1  | 6  | 5  | 7  | 22          | 19          |
| 4       | 7  | 2  | 3  | 4  | 1  | 6  | 5  | 22          | 19          |
| 5       | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 23          | 18          |
| suma    |    |    |    |    |    |    |    | 89,0        | 75          |
| średnia |    |    |    |    |    |    |    | 22,3        | 18,75       |

Opt.     3   2   5   7   1   6   4     DT=13

# Zmiana prawdopodobieństwa operacji

- ❑ Podczas rozwoju kolejnych populacji raz zadane na początku prawdopodobieństwo krzyżowania i mutacji może nie być optymalne. Dobre rezultaty dało liniowe zmniejszanie prawd. krzyżowania i zwiększanie prawd. mutacji (Davis, Syswerda).
- ❑ Jeśli mamy więcej operatorów, każdy z nich może mieć swoje dynamicznie zmieniające się prawdopodobieństwo zastosowania. Davis opisał technikę samoadaptacyjnej zmiany tych prawdopodobieństw.

# Zmiana prawdopodobieństwa operacji

- ❑ Premiowany jest każdy operator dający lepsze rozwiązanie niż istniejące w populacji. Waga operatora powstaje przez zsumowanie jego sukcesów w ostatnich 50 generacjach. Dla każdego zdarzenia operator jest wybierany losowo zgodnie z tabelą wag. Wielką zaletą takiej techniki jest wyeliminowanie uciążliwego doboru parametrów podczas wprowadzania nowych operatorów.
- ❑ Inne próby dotyczyły eksplotencjalnego zmniejszania prawd. mutacji (jak w SA).

# Strategie ewolucyjne

- ❑ Strategie ewolucyjne (ES, Evolutionary Strategy) to metoda stworzona przez niemieckich naukowców, Ingo Rechenberga i Hansa-Paula Schwefela, do optymalizacji budowy urządzeń mechanicznych. U podstaw leży zdroworozsądkowa zasada: „mała przyczyna powoduje mały efekt”.
- ❑ Strategie ewolucyjne różnią się znacznie od algorytmów genetycznych działających na reprezentacjach binarnych. ES działają na liczbach zmiennoprzecinkowych. Kolejna różnica tkwi w procesie selekcji. W algorytmie genetycznym lepiej przystosowane osobniki mogą być wybierane kilka razy. W ES osobniki wybierane są bez powtórzeń, a selekcja jest deterministyczna (słabsze osobniki nie mają szans na zasilenie nowej populacji bazowej). O wiele też większe znaczenie ma mutacja.

# Strategie ewolucyjne

- Są trzy podstawowe schematy ES:  $(1+1)$ ,  $(\mu + \lambda)$ ,  $(\mu, \lambda)$ , gdzie 1. element oznacza liczbę rodziców, 2. – sposób selekcji, 3. – liczbę potomków.

## BEGIN

$t := 0$ ;

inicjalizacja populacji początkowej  $P(t)$ ;

badanie dopasowania  $P(t)$ ;

## REPEAT

wybór  $\mu$  osobników spośród  $(\mu + \lambda) | (\lambda)$ , zgodnie z ich wartością funkcji dopasowania;

$t := t+1$ ;

utworzenie  $P(t)$  zawierającej  $\lambda$  dzieci drogą <krzyżowania i > mutacji;

badanie dopasowania  $P(t)$ ;

**UNTIL** (spełniony warunek zatrzymania)

**END.**



# Strategie ewolucyjne

- Osobnik w ES składa się z dwóch chromosomów. Pierwszym z nich jest wektor  $X$ , którego elementami są wartości zmiennych niezależnych. Drugi chromosom zawiera wartości standardowych odchyłeń wykorzystywanych podczas mutacji. Operacjom genetycznym ulegają obydwa chromosomy.
- Mutacja polega na wykonaniu operacji losowania z rozkładu normalnego dla dwóch chromosomów:

$$\sigma'_i = \sigma_i \exp(\tau' N(0,1) + \tau N_i(0,1)) \quad x'_i = x_i + \sigma'_i N_i(0,1)$$

$$x = (8, 12, 31, \dots, 5) \quad \delta = (.1, .3, .2, \dots, .5)$$

$$x_{mutowane} = (8.1, 11.8, 31.4, \dots, 5.9)$$

# Strategie ewolucyjne

- ❑ Operator krzyżowania (nazywany w ES - rekombinacją) najczęściej polega na wymianie lub uśrednianiu wartości wektorów  $X$  i  $\delta$ .
- ❑ Równe prawdopodobieństwo otrzymania parametru z dwóch rodziców:

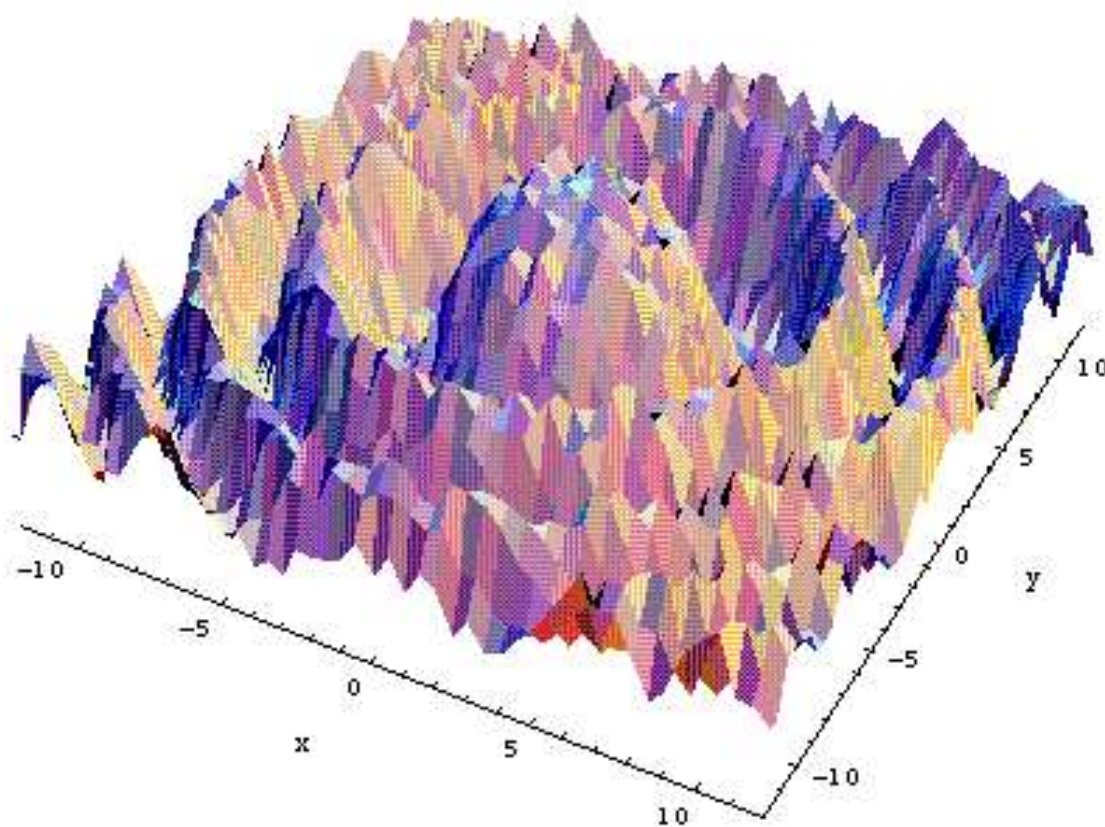
$$(8, 12, 31, \dots, 5) (2, 5, 23, \dots, 14) \\ (2, 12, 31, \dots, 14)$$

- ❑ Parametry dziecka są średnią wartością rodziców:

$$(8, 12, 31, \dots, 5) (2, 5, 23, \dots, 14) \\ (5, 8.5, 27, \dots, 9.5)$$

# Strategie ewolucyjne

- To niekoniecznie jest łatwe



# Programowanie ewolucyjne

- ❑ Programowanie ewolucyjne (EP, Evolutionary Programming) podobne jest do strategii ES ( $\mu + \lambda$ ), z mutacją jako jedynym operatorem. Twórcą jest Lawrence Fogel; wg zamierzeń EP miało służyć do modelowania rozwoju sztucznej inteligencji. EP nie nakłada żadnych ograniczeń na postać reprezentacji problemu, wychodząc z założenia, że kodowanie powinno w naturalny sposób wynikać z przedmiotu optymalizacji.
- ❑ W standardowym programowaniu ewolucyjnym potomek jest tworzony przez skopiowanie (zduplikowanie) rodzica, po czym jest on poddawany mutacji podobnej jak w ES. Różnica polega na tym, że odchylenie standardowe nie jest elementem reprezentacji rozwiązania, lecz jest obliczane na podstawie funkcji dopasowania rodzica. Selekcja oparta jest na stochastycznym procesie współzawodnictwa potomków i rodziców dającym w rezultacie ranking osobników.

# Programowanie ewolucyjne

**BEGIN**

$t := 0;$

inicjalizacja populacji początkowej  $P(t)$  składającej się z  $m$  osobników;

badanie dopasowania  $P(t)$ ;

**REPEAT**

$t := t+1;$

$P_m(t) = P(t)$ ; {populacja tymczasowa równa się populacji rodziców}

**FOR**  $i:=1$  **to**  $m$  **DO**

**BEGIN**

zduplikowanie  $i$ -tego osobnika i utworzenie  $i$ -tego potomka;

mutacja  $i$ -tego potomka;

wyliczenie funkcji dopasowania  $i$ -tego potomka;

dodanie  $i$ -tego potomka do populacji  $P_m(t)$ ;

**END;**

**FOR**  $j:=1$  **to**  $2m$  **DO**

porównanie  $j$ -tego osobnika z  $k$  losowo wybranymi konkurentami z populacji tymczasowej;

ranking osobników wg liczby zwycięstw w powyższym współzawodnictwie;

selekcja  $m$  najwyższej sklasyfikowanych osobników tworzących nową populację  $P(t)$ ;

**UNTIL** (spełniony warunek zatrzymania)

**END.**

# Programowanie genetyczne

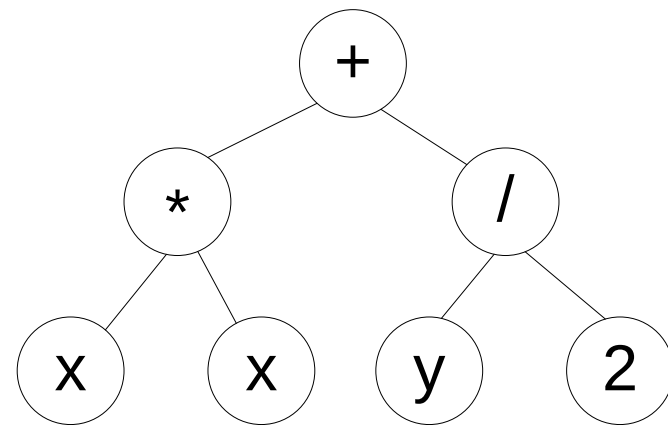
- ❑ Algorytm programowania genetycznego (GP, Genetic Programming) stworzony przez J.R. Koza (1990) wykorzystuje proces ewolucyjny do tworzenia najlepszego programu komputerowego.
- ❑ Programowanie genetyczne to ogólna nazwa dla klasy ewolucyjnych metod, które służą do rozwiązywania problemów, nie wymagając przy tym od użytkownika a priori wiedzy lub znajomości formuły czy struktury rozwiązania. W tym sensie jest podobne do sztucznych sieci neuronowych, jednak dzięki reprezentacji rozwiązań używanych w GP, może być stosowane do znacznie szerszego zakresu problemów, w tym problemów nie- i liniowych, o strukturze drzewa lub grafu. Ponadto, w przeciwieństwie do SSN, dają czytelne dla człowieka modele, które mogą być dalej modyfikowane lub interpretowane w celu lepszego opisu konkretnych zjawisk.

# Programowanie genetyczne

- 1: Randomly create an initial population of programs from the available primitives.
- 2: repeat
  - 3: Execute each program and ascertain its fitness.
  - 4: Select one or two program(s) from the population with a probability based on fitness to participate in genetic operations.
  - 5: Create new individual program(s) by applying genetic operations with specified probabilities.
- 6: until an acceptable solution is found or some other stopping condition is met (e.g., a maximum number of generations is reached).
- 7: return the best-so-far individual.

# Programowanie genetyczne

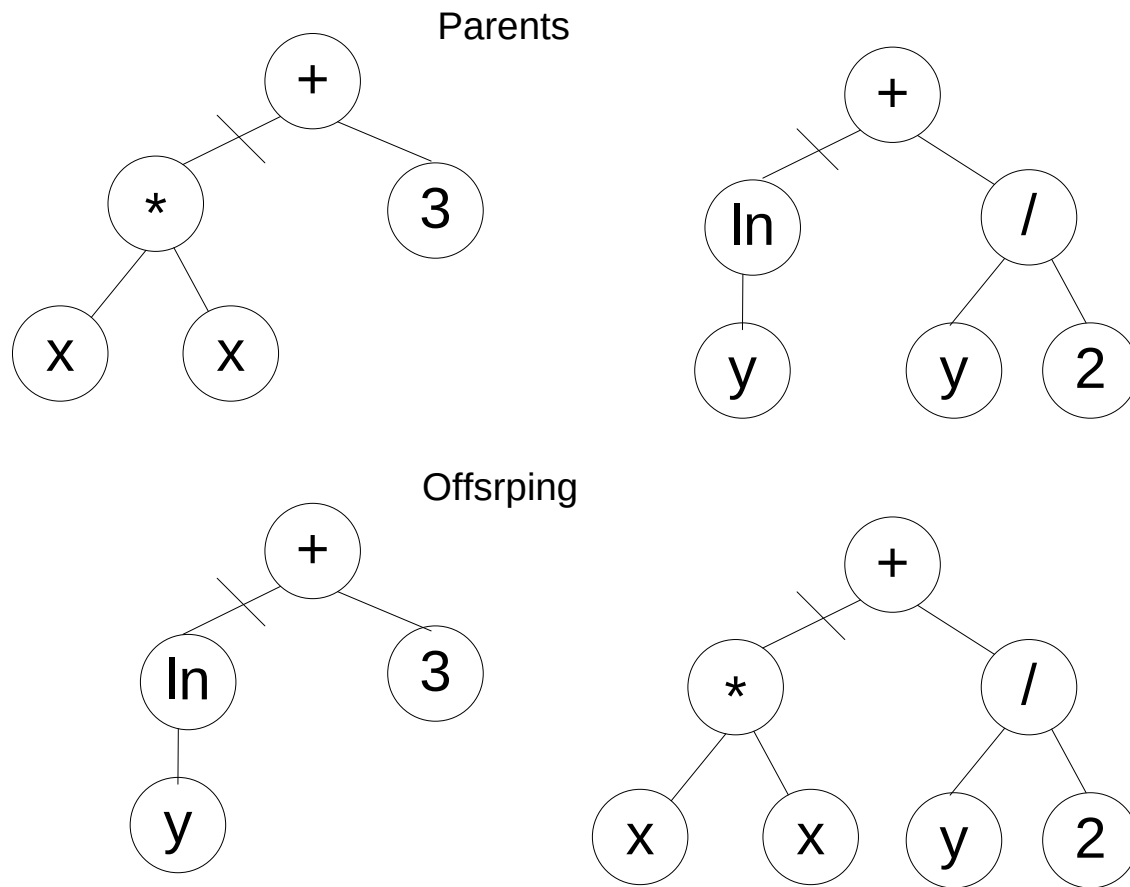
- ❑ Osobnik jest zbudowany w formie drzewa, które zawiera sekwencję elementów funkcjonalnych w takim sensie, że korzeń drzewa reprezentuje wyjście programu komputerowego, węzły reprezentują funkcje/operators, ich argumenty są umieszczone w podrzędnych węzłach, którymi również są funkcje lub symbole np. nazwa zmiennych lub stałych. Takie rozwiązanie może być łatwo przetłumaczone na wyrażenie LISP, zaproponowane przez Koza.
- ❑ Obok pokazano proste wyrażenie, które w LISP jest zapisane jako  $(+ (* x x) (/ y 2))$ , i łatwo je zapisać w standardowej formie jako  $f(x,y) = x*x + y/2$ .





# Programowanie genetyczne

## Krzyżówka



# Programowanie genetyczne

## Mutacja

Niektóre geny potomka mogą być zmieniane z określonym prawdopodobieństwem za pomocą dwóch rodzajów operatorów mutacji. **Mutacja punktowa** zmienia losowo wybraną funkcję lub symbol na inny z zestawu możliwych symboli.

W **mutacji poddrzewa**, poddrzewo poniżej punktu mutacji jest zmieniane na inne - generowane losowo - poddrzewo:

