Algorytmy ewolucyjne

materiały wykładowe

(c) 1998–2000 Marek Kisiel-Dorohinicki

Literatura

- David E. Goldberg „Algorytmy genetyczne i ich zastosowania''
- Zbigniew Michalewicz „Algorytmy genetyczne + struktury danych = programy ewolucyjne''
- http://galaxy.uci.agh.edu.pl/~doroh/genetic_info.html

Wprowadzenie

Obliczenia ewolucyjne (ang. *evolutionary computation*)

- heurystyczne techniki optymalizacji:
  - kombinatorycznej – poszukiwania najlepszej z punktu widzenia pewnego wskaźnika użyteczności kombinacji elementów (decyzji) z określonego zbioru (dziedzina dyskretna i nieuporządkowana)
  - funkcji – poszukiwania wartości parametrów minimalizujących (maksymalizujących) określoną funkcję celu (dziedzina ciągła)
- opierają się na analogiach do naturalnej ewolucji organizmów, modelowanej na różnych poziomach: genetycznym, cech osobniczych i cech gatunkowych
- najbardziej znane: algorytmy genetyczne

Ewolucja według Darwina

- zjawiska rozmnażania się i śmiertelności same z siebie nie tłumaczą pokoleniowej zmienności gatunkowej
- zasada doboru naturalnego (selekcji naturalnej): w środowisku o ograniczonych zasobach największe szanse na przeżycie będą miały osobniki nalepiej przystosowane, czyli najlepiej wykorzystujące dostępne zasoby
- występowanie u potomstwa niewielkich, losowych zmian cech osobniczych – mutacji

Syntetyczna teoria ewolucji

- jednostki informacji genetycznej: nukleotyd, gen, chromosom, genom
- ontogeneza: genotyp $\rightarrow$ fenotyp
- ewolucja a dziedziczenie
- rekombinacja w organizmach haplo- i diploidalnych
- mutacje somatyczne i pokoleniowe

Część I

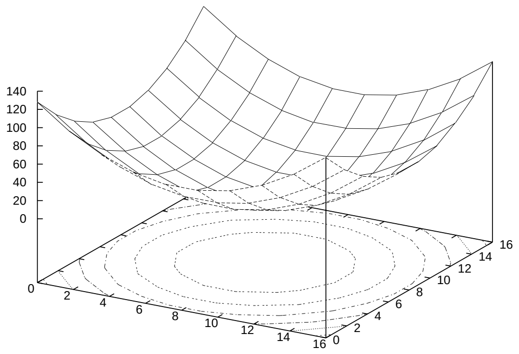
Algorytmy genetyczne

Simple Genetic Algorithm

1. Losowo wygeneruj populację początkową.
2. Zdekoduj i oceń każdego osobnika.
3. Utwórz pulę rodzicielską (ang. *mating pool*) z osobników wylosowanych z prawdopodobieństwem proporcjonalnym do ich jakości.
4. Utwórz kolejne pokolenie dokonając klonowania lub krzyżowania oraz mutacji osobników z puli rodzicielskiej.
5. Jeżeli nie spełniony warunek zakończenia, to przejdź do 2.

SGA – przykład działania

$$f(x,y) = (x-8)^2 + (y-8)^2$$



SGA – reprezentacja

$\vec{a} \in \{0,1\}^8$ – osobniki są ciągami 8-bitowymi kodującymi parametry x_i, y_i:

$$h(x,y) = \vec{a} \quad ; \quad h'(\vec{a}) = (x,y)$$

zatem np.: $\vec{a}_i = \underbrace{0110}_{x_i=6}\underbrace{1110}_{y_i=7}$

a wtedy: $f(h'(\vec{a}_i)) = f(x_i, y_i) = 5$

SGA – parametry

- wielkość populacji: $N = 4$
- prawdopodobieństwo krzyżowania: $p_c = 1$
- prawdopodobieństwo mutacji: $p_m = 0$

SGA – symulacja odręczna 1

populacja początkowa:

$$P[0] = \{01101110, 01010011, 10000110, 10100001\}$$

selekcja (utworzenie puli rodzicielskiej)

osobnik $\vec{a}$	parametry (x_i, y_i)	fitness $f(x_i, y_i)$	prawdop. $f_i / \sum f$	liczba kopii N_i
01101110	(6,7)	5	0,057	0
01010011	(10,12)	20	0,230	1
10000110	(1,6)	53	0,609	2
10100001	(5,8)	9	0,103	1

SGA – symulacja odręczna 2

reprodukcja (utworzenie nowej populacji)

osobnik $\vec{a}_i^*$	nowy osobnik $\vec{a}_i$	parametry (x_i, y_i)	fitness $f(x_i, y_i)$
10000|110	10000011	(1,12)	65
01010|011	01010110	(10,6)	8
100|00110	10000001	(1,8)	49
101|00001	10100110	(5,6)	13

Algorytmy genetyczne

reprezentacja: dyskretna (binarna)

selekcja: probabilistyczna

operatory genetyczne (wariacji):

- głównie krzyżowanie: wymiana fragmentów łańcuchów kodowych
- drugorzędna mutacja: zmiana (odwrócenie) wartości elementu łańcucha kodowego (bitu)

model ewolucji: na poziomie genetycznym

Reprezentacja binarna

$$\vec{a} \in \{0,1\}^{\ell}, \quad \ell = \sum_{i=1}^{n} \ell_i \text{ – ustalona długość chromosomu}$$

parametry dyskretne: $x_i \in \Omega_i$, $\operatorname{card} \Omega_i = L_i$
kodowane w segmentach o długości $\ell_i = \lceil \log_2 L_i \rceil$

liczby całkowite (wartości uporządkowane): $x_i = 0, \ldots, L_i - 1$

- kod uzupełniający do 2 (zmiana na 1 pozycji reprezentacji różnie wpływa na zmianę wartości)
- kod Greya (reprezentacje kolejnych wartości różnią się na 1 pozycji)

Odchylenie reprezentacji (ang. *representational bias*)

gdy $L_i < 2^{\ell_i}$

- obcinanie (ang. *clipping*): $L_i - 1$ wartości reprezentowanych jest przez $L_i - 1$ różnych kombinacji bitów, ostatnia wartość reprezentowana przez pozostałych $2^{\ell_i} - L_i + 1$ kombinacji
- skalowanie (ang. *scaling*): $2^{\ell_i} - L_i$ wybranych wartości reprezentowanych jest przez dwie kobinacje bitów, np.:

$$a_i = \left\lfloor \frac{x_i}{L_i - 1}(2^{\ell_i} - 1) \right\rfloor$$

Kodowanie parametrów rzeczywistych

parametry $x_i \in [u_i, v_i] \subset \boldsymbol{R}$

kodowane są przy pomocy segmentów bitowych o długości ℓ_i reprezentujących liczby całkowite z_i, odwzorowywane liniowo na przedział $[u_i, v_i]$:

$$x_i = u_i + \frac{z_i}{2^{\ell_i} - 1}(v_i - u_i)$$

długość segmentu ℓ_i jest kompromisem między

dokładnością przybliżenia

oraz

szybkością poszukiwania

Dynamiczne kodowanie parametrów (ang. *dynamic parameter encoding*)

polega na sukcesywnym ograniczaniu zasięgu poszukiwań w trakcie działania algorytmu

operator powiększenia (ang. *zoom*) zawęża przedziały poszukiwanych wartości poszczególnych parametrów:

$$[u_i, v_i] \to [u_i', v_i'] \subset [u_i, v_i]$$

uwaga! operacja powiększenia jest **nieodwracalna** i musi być ostrożnie stosowana by nie doprowadzić do przedwczesnej zbieżności algorytmu

Reprezentacja – inne możliwości

- kodowanie wektorów parametrów:
  - najczęściej **konkatenacja** (istotna kolejność!)
  - można także **z przeplotem**
- w uzasadnionych przypadkach do reprezentacji użyć można **alfabetu niebinarnego**, częściej stosowana jest **reprezentacja zmiennopozycyjna**
- **kodowanie adaptatywne** – zmienia się znaczenie pozycji i/lub długość ciągu kodowego

Selekcja w algorytmach genetycznych

- wybór osobników do puli rodzicielskiej (dokonywana przed zastosowaniem operatorów wariacji)
- ściśle związana z miarą przystosowania, ale probabilistyczna (prawdopodobieństwo wyboru osobnika zależy od jego jakości)
- różne modele – nie wszystkie biologicznie akceptowalne

Selekcja proporcjonalna
(ang. *fitness proportionate selection*)

N – wielkość populacji

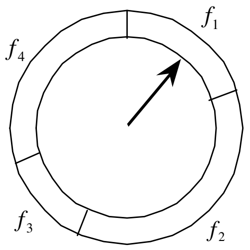
f_i – miara przystosowania i-tego osobnika

$\bar{f} = \frac{1}{N}\sum_j f_j$ – średnia miara przystosowania całej populacji

prawdopodobieństwo wyboru i-tego osobnika wynosi:

$$p_i = \frac{f_i}{\sum_j f_j} = \frac{f_i}{\bar{f}N}$$

Roulette wheel algorithm




```
for i := 1 to N do
  begin
    r := random
    sum := j := 0
    while sum ≤ r do
      begin
        j := j + 1
        sum := sum + p_j
      end
    select(j)
  end
```


Model wartości oczekiwanej

ograniczenie ilości osobników losowanych metodą ruletki do wartości oczekiwanej $f_i/\bar{f}$

zaleta: mniejszy rozrzut losowy i „utrata” alleli

Stochastic universal sampling

```
r := random
sum := 0
for i := 1 to N do
  begin
    sum := sum + E(N_i)
    while sum > r do
      begin
        r := r + 1
        select(i)
      end
  end
```

Selekcja proporcjonalna – cd.

dla algorytmów o stałej wielkości populacji wartość oczekiwana liczby kopii osobnika po selekcji wynosi:

$$E(N_i) = p_i N = f_i/\bar{f}$$

gdy $f_i \gg \bar{f}$ to $N_i \gg 1$ i geny i-tego osobnika szybko rozprzestrzeniają się na całą populację – **przedwczesna zbieżność**

gdy $\forall f_i \approx f_{max}$ to $N_1 \approx \ldots \approx N_N \approx 1$ i nie ma wyraźnie lepszych rozwiązań – **stagnacja**

Techniki skalowania

1. skalowanie liniowe
$$f' = af + b$$
2. skalowanie σ-obcinające (σ-*truncation*)
$$f' = f - (\bar{f} - c\sigma) \quad \text{gdy } f' < 0 \text{ to } f' \approx 0$$
3. skalowanie potęgowe
$$f' = f^k$$
4. model Boltzmana
$$f' = e^{f/T}$$

Selekcja rankingowa (ang. *rank selection*)

osobniki są sortowane wg wartości *fitness*

$$f_i \geq f_j \quad \text{dla} \quad i < j$$

prawdopodobieństwo wyboru przyporządkowywane jest kolejnym osobnikom zgodnie z zadanym rozkładem (najczęściej liniowym lub wykładniczym)

Selekcja rankingowa – cd.

zalety:

- brak przedwczesnej zbieżności
- brak stagnacji
- niepotrzebna jawna funkcja przystosowania

wady:

- potrzeba porządkowania populacji
- trudności w analizie teoretycznej
- brak uzasadnienia biologicznego

Selekcja turniejowa
(ang. *tournament selection*)

z grupy osobnikow losowo wybranych z populacji wybierany jest najlepszy

zalety:

- brak przedwczesnej zbieżności
- brak stagnacji
- brak potrzeby porządkowania całej populacji
- niepotrzebna jawna funkcja przystosowania
- odwzorowanie selekcji naturalnej

Model elitarny
(ang. *elitist selection*)

kopia najlepszego osobnika jest zawsze przekazywana do następnego pokolenia

zaleta: gwarantowana zbieżność

wada: ryzyko wpdanięcia w ekstremum lokalne

alternatywa: zapamiętywanie najlepszego osobnika poza populacją

Model ze ściskiem
(ang. *crowding model*)

za każdym razem, gdy generowany jest nowy osobnik, usuwany jest inny, najbardziej do niego podobny z losowo wybranej grupy

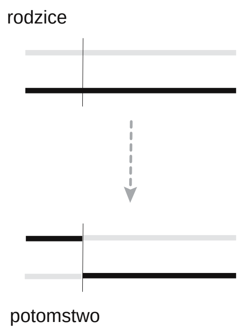
zaleta: zmniejszenie konkurencji w niszach ekologicznych

Operatory wariacji w algorytmach genetycznych

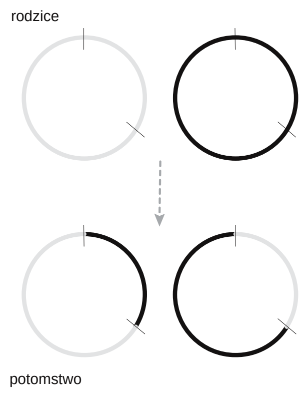
- krzyżowanie $p_c \in [0.5, 0.8]$ (alternatywa: klonowanie)
  - jednopunktowe (CP=1)
  - wielopunktowe (CP=k)
  - jednorodne
- mutacja $p_m \in [0.001, 0.01]$ lub $p_m = 1/\ell$

Krzyżowanie

jednopunktowe:



dwupunktowe:



Pojęcie schematu

schemat – wzorzec opisujący podzbiór ciągów kodowych podobnych ze względu na ustalone pozycje

reprezentacja schematu – ciąg symboli z alfabetu genów, poszerzonego o symbol uniwersalny $*$ (gwiazdka) = *nieistotne* (jeżeli osobniki $\vec{a}_i \in \{0,1\}^\ell$, to schematy $H_j \in \{0,1,*\}^\ell$)

$o(H)$ – rząd schematu H – liczba pozycji ustalonych w schemacie H

$\delta(H)$ – długość definiująca (rozpiętość) schematu H – odległość między dwoma skrajnymi pozycjami ustalonymi w schemacie H

Przykłady schematów

alfabet $V = \{0,1\}$
długość ciągu kodowego $\ell = 6$

$H_1 = 110*11$	$H_2 = *0**1*$	$H_3 = 1*****$
$o(H_1) = 5$	$o(H_2) = 2$	$o(H_3) = 1$
$\delta(H_1) = 5$	$\delta(H_2) = 3$	$\delta(H_3) = 0$

2^ℓ możliwych ciągów kodowych
3^ℓ możliwych schematów
$2^{\ell - o(H)}$ osobników reprezentujących schemat
2^ℓ schematów reprezentowanych przez osobnika

Selekcja proporcjonalna a schematy

N – wielkość populacji

f_i – miara przystosowania i-tego osobnika

$\bar{f} = \frac{1}{N}\sum_j f_j$ – średnia miara przystosowania całej populacji

$p_i = \frac{f_i}{\sum_j f_j} = \frac{f_i}{\bar{f}N}$ – prawdopodobieństwo wyboru i-tego osobnika

$E(N_i) = p_i N = \frac{f_i}{\bar{f}}$ – wartość oczekiwana liczby kopii osobnika po selekcji

$m(H,t)$ – liczba osobników reprezentujących schemat H w pokoleniu t

wartość oczekiwana liczby osobników reprezentujących schemat H w pokoleniu $t+1$:

$$E[m(H,t+1)] = \sum_{x_i \in H} E(N_i) = \sum_{x_i \in H} \frac{f_i(t)}{\bar{f}(t)} = m(H,t)\frac{f(H,t)}{\bar{f}(t)}$$

Krzyżowanie i mutacja a schematy

- prawdopodobieństwo przeżycia schematu po krzyżowaniu jednopunktowym:

$$S_c \geq 1 - p_c \left(\frac{\delta(H)}{\ell - 1} \right)$$

gdzie p_c – prawdopodobieństwo krzyżowania

- prawdopodobieństwo przeżycia schematu po mutacji:

$$S_m \geq (1 - p_m)^{o(H)} \approx 1 - p_m \cdot o(H)$$

gdzie $p_m \ll 0$ – prawdopodobieństwo mutacji

Twierdzenie o schematach (ang. *Schema Theorem*)

dolna granica oczekiwanej liczby reprezentantów schematu H w następnym pokoleniu algorytmu genetycznego, otrzymanym przy użyciu selekcji proporcjonalnej oraz krzyżowania jednopunktowego i mutacji, wyraża się następującym wzorem:

$$E[m(H, t+1)] \geq m(H, t) \frac{f(H, t)}{\bar{f}(t)} \left[1 - p_c \left(\frac{\delta(H)}{\ell - 1} \right) \right] (1 - p_m)^{o(H)}$$

lub w przybliżeniu:

$$E[m(H, t+1)] \geq m(H, t) \frac{f(H, t)}{\bar{f}(t)} \left[1 - p_c \left(\frac{\delta(H)}{\ell - 1} \right) - p_m \cdot o(H) \right]$$

Wnioski z twierdzenia o schematach

dla schematów niskiego rzędu i o małej rozpiętości destruktywne efekty krzyżowania i mutacji można pominąć, a wtedy:

$$E[m(H, t+1)] = m(H, t) \frac{f(H, t)}{\bar{f}(t)} = m(H, t)(1 + \varepsilon)$$

jeżeli przyjmiemy, że $f(H, t) = \bar{f}(t) + \varepsilon \bar{f}(t)$, czyli schemat H ma dopasowanie o ε% większe niż średnia w populacji

zakładając, że ε nie zmienia się w czasie, długoterminowy efekt przetwarzania schematu H będzie następujący:

$$E[m(H, t)] = m(H, 0)(1 + \varepsilon)^t$$

czyli oceniane powyżej średniej **schematy niskiego rzędu** i o **małej rozpiętości** uzyskują **wykładniczo rosnącą** liczbę reprezentantów w kolejnych pokoleniach

Hipoteza cegiełek (ang. *Building Block Hypothesis*)

algorytm genetyczny poszukuje lepszych rozwiązań przez zestawianie wysokiej jakości schematów niskiego rzędu i o małej rozpiętości (tzw. cegiełek)

- oceniając poszczególne ciągi kodowe GA pośrednio ocenia jakość reprezentowanych przez nie schematów
- krzyżowanie pozwala na zestawianie ze sobą wysokiej jakości schematów niskiego rzędu, budując tak samo dobre lub lepsze schematy wyższego rzędu
- jednocześnie przetwarzanych jest 2^ℓ do $N \cdot 2^\ell$ schematów, ale część z nich jest niszczona – Holland oszacował, że „użytecznie'' przetwarzanych jest N^3 schematów (ukryta równoległość, ang. *implicit parallelism*)

Część II

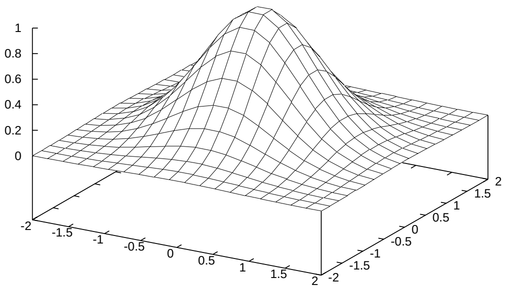
Strategie ewolucyjne

(1+1)-Evolutionary Strategy

1. Losowo wygeneruj i oceń osobnika.
2. Utwórz osobnika potomnego dokonując mutacji zgodnie z rozkładem normalnym o odchyleniu standardowym σ.
3. Oceń potomka.
4. Porównaj rodzica i potomka. Lepszego pozostaw do następnego pokolenia.
5. W razie potrzeby co k kroków dokonaj modyfikacji odchylenia standardowego σ: jeżeli liczba udanych mutacji jest większa niż 1/5 – zwiększ σ, jeżeli mniejsza – zmniejsz σ.
6. Jeżeli nie spełniony warunek zakończenia, to przejdź do 2.

(1+1)-ES – przykład działania

$$f(x, y) = e^{-x^2-y^2}$$



(1+1)-ES – reprezentacja i parametry

$(x, y) \in \boldsymbol{R}^2$ – osobnik jest zmiennopozycyjną reprezentacją rzeczywistych parametrów funkcji

parametry działania:

- początkowe odchylenie standardowe mutacji: $\sigma = 1.0$
- współczynniki modyfikacji odchylenia standardowego: $c_1 = 0.82$ oraz $c_2 = 1/0.82 = 1.22$
- liczba pokoleń między modyfikacjami odchylenia standardowego: $k = 5$

(1+1)-ES – symulacja odręczna

wylosowany osobnik $(x, y) = (1.7, -0.9)$ o dopasowaniu 0.025

kolejne pokolenia:

osobnik (x,y)	fitness $f(x,y)$	mutacja $(\delta x, \delta y)$	potomek (x', y')	fitness $f(x', y')$	udana mutacja
(1.7,-0.9)	0.025	(-0.4,-0.7)	(1.3,-1.6)	0.015	nie
(1.7,-0.9)	0.025	(-0.8,0.5)	(0.9,-0.4)	0.379	tak
(0.9,-0.4)	0.379	(0.2,-1.1)	(1.1,-1.5)	0.105	nie
(0.9,-0.4)	0.379	(-0.5,0.8)	(0.4,0.4)	0.726	tak
(0.4,0.4)	0.726	(0.9,0.1)	(1.3,0.5)	0.144	nie

w $k = 5$ pokoleniach liczba udanych mutacji = 2, czyli należy zwiększyć odchylenie standardowe: $\sigma' = c_2 \cdot \sigma = 1.22 \cdot 1.0 = 1.22$

Strategie ewolucyjne

reprezentacja: zmienooprzecinkowa

selekcja: deterministyczna

operatory genetyczne (wariacji):

- głównie mutacja: zaburzenie wartości zgodnie z rozkładem normalnym
- drugorzędna rekombinacja: dyskretna lub pośrednia (wymiana lub uśrednienie poszczególnych wartości)

model ewolucji: cech osobniczych

Reprezentacja zmiennoprzecinkowa

$$\vec{a} = \vec{x} \in \boldsymbol{R}^n$$

lub

$$\vec{a} = \vec{x} \in \times_{i=1}^{n} [u_i, v_i] \subset \boldsymbol{R}^n$$

- naturalna dla problemów optymalizacji parametrycznej (optymalizacji funkcji)
- nie wymaga kodowania

Parametry strategii ewolucyjnej

osobniki ES często zawierają także zestaw parametrów opisujących działanie mutacji, ewoluujących jednocześnie z rozwiązaniami zadania:

- odchylenia standardowe mutacji σ_i
- tzw. kąty rotacji α_{ij} pełniące rolę współczynników korelacji mutacji

po ich uwzględnieniu otrzymamy:

$$\vec{a} = (\vec{x}, \vec{\sigma}, \vec{\alpha}) \in \boldsymbol{R}^n \times \boldsymbol{R}_+^{n_\sigma} \times [-\pi, \pi]^{n_\alpha}$$

gdzie: $n_\sigma = 1, \ldots, n$

$n_\alpha = 0, \ldots, (2n - n_\sigma)(n_\sigma - 1)/2$

Mutacja w strategiach ewolucyjnych

- zaburzenia wartości poszczególnych parametrów (cech) zgodnie z rozkładem normalnym o danym odchyleniu standardowym (niewielkie zmiany powinny zdarzać się częściej)
- dotyczy zarówno reprezentacji rozwiązania, jak i samych parametrów mutacji
- nie można dopuścić do nadmiernego zmniejszenia odchylenia standardowego ($\sigma_i \geq \varepsilon_\sigma$) oraz opuszczenia przedziału wartości dopuszczalnych przez kąty rotacji ($\alpha_j \in [-\pi, \pi]$)

Uogólniony schemat mutacji

$$\sigma_i' = \sigma_i \cdot e^{\tau' \cdot N(0,1) + \tau \cdot N_i(0,1)}$$
$$\alpha_j' = \alpha_j + \beta \cdot N_j(0,1)$$
$$\vec{x}' = \vec{x} + \vec{N}(\vec{0}, \mathbf{C}(\vec{\sigma}', \vec{\alpha}'))$$

gdzie: $N(0,1)$ – realizacja jednowymiarowej zmiennej losowej o rozkładzie normalnym ze średnią zero i jednostkowym odchyleniu

τ', τ – współczynniki globalnej (dla osobnika) i indywidualnej (dla pojedynczego odchylenia) szybkości zmian mutacji:

$$\tau \propto 1/\sqrt{2n} \quad \text{oraz} \quad \tau' \propto 1/\sqrt{2\sqrt{n}}$$

β – szybkość zmian współczynników korelacji mutacji:

$$\beta \approx 0.0873\ (5^\circ)$$

$\vec{N}(\vec{0}, \mathbf{C})$ – realizacja wielowymiarowej zmiennej losowej o uogólnionym rozkładzie normalnym ze średnią $\vec{0}$ i macierzą kowariancji $\mathbf{C}^{-1} = \mathbf{C}^{-1}(\vec{\sigma}, \vec{\alpha})$

Przypadki szczególne mutacji

dla $n_\alpha = 0$:

$$\sigma_i' = \sigma_i \cdot e^{\tau' \cdot N(0,1) + \tau \cdot N_i(0,1)}$$
$$x_i' = x_i + \sigma_i' \cdot N_i(0,1)$$

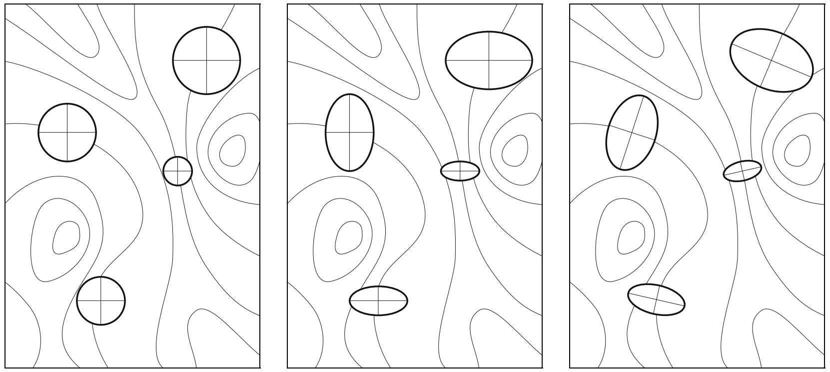
dla $n_\sigma = 1$:

$$\sigma_i' = \sigma_i \cdot e^{\tau_0 \cdot N(0,1)}$$
$$x_i' = x_i + \sigma_i' \cdot N_i(0,1)$$

gdzie: τ_0 – współczynnik szybkości zmian mutacji:

$$\tau \propto 1/\sqrt{n}$$

Ilustracja efektów mutacji



$n_\sigma = 1$ $\quad$ $n_\sigma = 2 \quad n_\alpha = 0$ $\quad$ $n_\sigma = 2 \quad n_\alpha = 1$

Rekombinacja w strategiach ewolucyjnych

- dokonywana na poziomie pojedynczych parametrów (cech)
- dotyczy reprezentacji rozwiązania oraz parametrów mutacji (jakkolwiek może być stosowana jednocześnie w różnych formach dla poszczególnych wektorów składowych)
- **płciowa** (potomek tworzony jest na podstawie dwóch losowo wybranych rodziców) lub **panmiktyczna** (dla wybranego rodzica partner dobierany jest niezależnie dla każdego parametru)
- **dyskretna** (dziedziczenie poszczególnych parametrów) lub **algebraiczna** (kombinacja liniowa parametrów rodziców)

Formy rekombinacji

$$x_i' = \begin{cases} x_i^r & \text{brak} \\ x_i^r \vee x_i^s & \text{dyskretna} \\ x_i^r \vee x_i^{s_i} & \text{dyskretna panmiktyczna} \\ x_i^r + (x_i^s - x_i^r)/2 & \text{pośrednia} \\ x_i^r + (x_i^{s_i} - x_i^r)/2 & \text{pośrednia panmiktyczna} \\ x_i^r + \chi(x_i^s - x_i^r) & \text{uogólniona pośrednia} \\ x_i^r + \chi_i(x_i^{s_i} - x_i^r) & \text{uogólniona pośrednia panmiktyczna} \end{cases}$$

gdzie: $r, s \in \{1, \ldots, N\}$ – identyfikują wylosowanych rodziców

$\chi \in [0, 1]$ – zmienna losowa o rozkładzie jednostajnym w przedziale $[0, 1]$

(obecność indeksu i oznacza, że losowanie powtarzane jest niezależnie dla każdej składowej x_i)

Selekcja w strategiach ewolucyjnych

- wybór osobników do następnej populacji (dokonywana po zastosowaniu operatorów wariacji)
- bezpośrednio związana z miarą przystosowania – deterministyczna (wybór najlepszych osobników)
- dwa modele:
  - strategia „z plusem” (μ+λ)-ES
  - strategia „z przecinkiem” (μ,λ)-ES

(μ+λ)-ES

- wybór μ osobników z populacji rodziców oraz potomków (μ rodziców + λ potomków)
- zwykle $\lambda \leq \mu$, przyjmuje się nawet $\lambda = \mu = 1$
- gwarantuje przeżywanie najlepszego osobnika, przez co jest monotonicznie zbieżna

(μ,λ)-ES

- wybór μ osobników spośród λ potomków
- niezbędne jest $\lambda > \mu$, w praktyce przyjmuje się $\lambda \gg \mu$ (np. $\mu/\lambda \approx 1/7$) oraz $\mu \gg 1$ (np. $\mu = 15$)
- nie gwarantuje przeżywania najlepszego osobnika, przez co łatwiej omija lokalne ekstrema i lepiej działa przy dynamicznie zmieniającym się środowisku

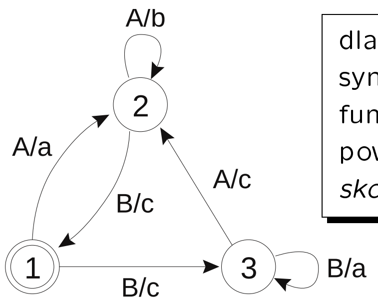
Część III

Programowanie ewolucyjne

Symulacja sztucznej inteligencji

pierwsze realizacje *programowania ewolucyjnego* – symulacja inteligentnego zachowania rozumianego jako:

1. umiejętność przewidywania zmian w środowisku
2. zdolność do reakcji właściwej dla uzyskanej predykcji oraz realizowanego celu



dla środowiska opisanego ciągiem symboli ze skończonego alfabetu funkcjonalność taką zapewnia odpowiednio skonstruowany *automat skończony*

Ewoluujące automaty skończone

- populacja automatów skończonych reprezentuje możliwe zachowania w określonym środowisku
- operatory wariacji: jedynie mutacja w różnych formach:
  - ◇ zmiana przejścia między stanami
  - ◇ zmiana symbolu wyjściowego
  - ◇ dodanie stanu
  - ◇ usunięcie stanu
  - ◇ zmiana stanu początkowego
- selekcja: deterministyczna typu (μ+μ) – jakość automatów szacowana przez porównanie symboli wyjściowych z kolejnymi elementami ciągu opisującego środowisko

Programowanie ewolucyjne

reprezentacja: zależna od zadania, najczęściej zmiennoprzecinkowa

selekcja: probabilistyczna

operatory genetyczne: jedynie mutacja (dla reprezentacji zmiennoprzecinkowej – zaburzenie wartości zgodnie z rozkładem normalnym)

model ewolucji: cech gatunkowych

Parametry programowania ewolucyjnego

podobnie jak w ES osobniki EP zawierać mogą zestaw parametrów opisujących działanie mutacji, ewoluujących jednocześnie z rozwiązaniami zadania:

- wariancje mutacji ν_i (meta-EP)
- współczynniki korelacji mutacji ρ_{ij} (Rmeta-EP)

po ich uwzględnieniu otrzymamy:

$$\vec{a} = (\vec{x}, \vec{\nu}, \vec{\rho}) \in \boldsymbol{R}^n \times \boldsymbol{R}^n_+ \times [-1, 1]^{n(n-1)/2}$$

Mutacja w programowaniu ewolucyjnym

- zaburzenia wartości poszczególnych parametrów (cech) zgodnie z rozkładem normalnym o danej wariancji
- dotyczy zarówno reprezentacji rozwiązania, jak i samych parametrów mutacji
- nie można dopuścić ujemnej wariancji, ale może być ona dowlonie bliska 0 ($\nu_i < 0 \Rightarrow \nu_i := \varepsilon_\nu > 0$) oraz opuszczenia przedziału wartości dopuszczalnych przez współczynniki korelacji ($\alpha_j \in [-1, 1]$)

Możliwe schematy mutacji

EP (wersja standardowa)

$$x'_i = x_i + \sqrt{\beta_i \cdot f(\vec{x}) + \gamma_i} \cdot N_i(0, 1)$$

meta-EP

$$x'_i = x_i + \sqrt{\nu_i} \cdot N_i(0, 1)$$
$$\nu'_i = \nu_i + \sqrt{\zeta \nu_i} \cdot N_i(0, 1)$$

gdzie: $N(0, 1)$ – realizacja jednowymiarowej zmiennej losowej o rozkładzie normalnym ze średnią zero i jednostkowym odchyleniu

β_i, γ_i, ζ – parametry ustalane dla konkretnego zadania

Możliwe schematy mutacji – cd.

Rmeta-EP

$$\vec{x}' = \vec{x} + \vec{N}(\vec{0}, \mathbf{C}(\vec{\nu}, \vec{\rho}))$$
$$\nu_i' = \nu_i + \sqrt{\zeta \nu_i} \cdot N_i(0,1)$$
$$\rho_j' = \rho_j + \sqrt{\varpi \rho_j} \cdot N_j(0,1)$$

gdzie: $\vec{N}(\vec{0}, \mathrm{C})$ – realizacja wielowymiarowej zmiennej losowej o uogólnionym rozkładzie normalnym ze średnią $\vec{0}$ i macierzą kowariancji $\mathbf{C}^{-1} = \mathbf{C}^{-1}(\vec{\nu}, \vec{\rho})$

ζ, ϖ – parametry ustalane dla konkretnego zadania

Część IV

Ewolucyjne techniki obliczeniowe

Schemat algorytmu ewolucyjnego

$t := 0$
initialize $P[0]$
evaluate $P[0]$
while not *terminate* **do**
  begin
    $P'[t] :=$ **variation** $P[t]$
    evaluate $P'[t]$
    $P[t+1] :=$ **select** $P'[t] \cup Q$
    $t := t+1$
  end

$t = 0, 1, \ldots$ – kolejne iteracje algorytmu

$P[t] = \{\vec{a}_1^t, \ldots, \vec{a}_\mu^t\}$ – populacja μ osobników $\vec{a}_i \in I$, reprezentujących potencjalne rozwiązania zadania

$P'[t]$ – populacja λ potomków, generowanych przy użyciu operatorów wariacji

Q – dodatkowy zbiór osobników na potrzeby selekcji (najczęściej przyjmuje się $Q = P[t]$ lub $Q = \emptyset$)

Kodowanie

przestrzeń poszukiwań:

zadania: wartości parametrów opisujących przeszukiwane obiekty (systemy, procesy) $\vec{x}_i \in D$ ($\rightarrow$ *fenotypy*)

algorytmu ewolucyjnego: ciągi kodowe $\vec{a}_i \in I$ ($\rightarrow$ *genotypy*)

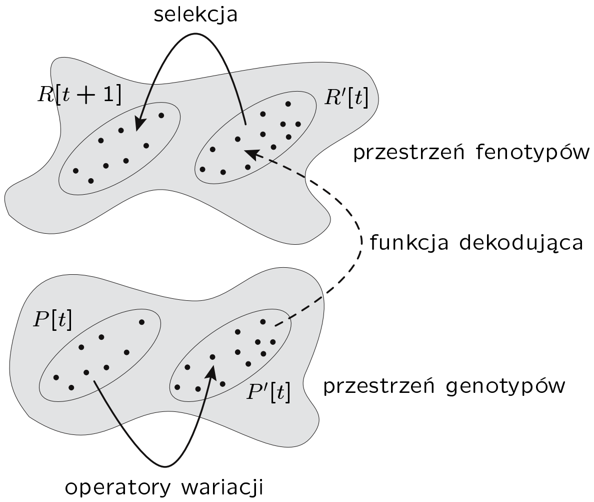
potrzebna funkcja kodująca:

$$h : D \rightarrow I$$

oraz dekodująca:

$$h' : I \rightarrow D$$

Przestrzeń geno- i fenotypów



Wybór reprezentacji

- reprezentacja dyskretna (binarna) i ciągła (zmiennoprzecinkowa), kodowanie mieszane
- związek z operatorami wariacji, operatory dedykowane
- niedopuszczalne ciągi kodowe a efektywność procesu ewolucji
- złożoność obliczeniowa funkcji dekodującej
- reprezentacje pozycyjna i o zmiennej długości

Operatory wariacji

bezpłciowe (ang. *asexual*) – nowy ciąg kodowy generowany jest na podstawie dokładnie **jednego** osobnika rodzicielskiego przy wprowadzeniu niewielkich losowych zmian w materiale genetycznym (*mutacja*)

płciowe (ang. *sexual*) nowy ciąg kodowy generowany jest na podstawie materiału genetycznego **dwóch** osobników rodzicielskich (*rekombinacja*), najczęściej przez wymianę fragmentów ich ciągów kodowych (*krzyżowanie*)

panmiktyczne (ang. *panmictic*) – nowy ciąg kodowy generowany jest na podstawie **wielu** osobników rodzicielskich (również nazywany rekombinacją, mimo braku biologicznego uzasadnienia tego terminu)

Selekcja

- operacja naśladująca zasadę doboru naturalnego (przeżywanie najsilniejszych) – ściśle związana z miarą przystosowania
- wybór osobników do puli rodzicielskiej lub do następnego pokolenia
- różne modele – nie wszystkie biologicznie akceptowalne
- probabilistyczna lub deterministyczna

Różnorodność populacji a napór selekcyjny

selekcja	„miękka” (ang. *soft*)	„twarda” (ang. *hard*)
poszukiwanie	zorientowane na objętość	zorientowane na ścieżkę
sposób działania	generowanie nowych rozwiązań	wykorzystanie zdobytych informacji
różnorodność populacji	duża	mała
zbieżność algorytmu	nacisk na niezawodność	nacisk na szybkość

Wielkość populacji pośredniej

- dla algorytmu ewolucyjnego o ustalonym stanie (ang. *steady-state*) $\lambda = 1$ lub generalnie $\lambda \ll \mu$
- gdy $1 \leq \lambda \leq \mu$ możliwy jest wybór wielkości współczynnika wymiany (ang. *generation gap*)
- w naturze zjawiskiem normalnym jest nadwyżka reprodukcyjna (ang. *reproduction surplus*), czyli $\lambda > \mu$

Klasyczne algorytmy ewolucyjne

- algorytmy genetyczne (ang. *Genetic Algorithm*)
  - programowanie genetyczne (ang. *Genetic Programming*)
  - systemy klasyfikujące (ang. *Classifier Systems*)
- strategie ewolucyjne (ang. *Evolutionary Strategies*)
- programowanie ewolucyjne (ang. *Evolutionary Programming*)

Porównanie cech

	GA	**ES**	**EP**
model ewolucji	genetycznej	cech osobniczych	cech gatunkowych
reprezentacja	binarna	zmiennoprzecinkowa	
mutacja	odwrócenie bitu	zaburzenie wartości zgodnie z rozkładem Gaussa	
rekombinacja	krzyżowanie	dyskretna lub pośrednia	brak
adaptacja	brak	parametry mutacji	brak
selekcja	probabilistyczna	deterministyczna	probabilistyczna

Dobór parametrów algorytmu ewolucyjnego

1. parametry ustalone dla całego algorytmu
   - drogą kolejnych eksperymentów
   - meta-algorytm ewolucyjny
2. adaptacja parametrów w trakcie działania algorytmu
   - ustalone reguły modyfikacji parametrów w oparciu o obserwowany stan systemu
   - równoczesna ewolucja rozwiązań i parametrów algorytmu

Ocena jakości działania algorytmu ewolucyjnego

- efektywność *on-line* (bieżąca):

$$\frac{1}{T}\sum_{t=1}^{T} f(t)$$

- efektywność *off-line* (poziom zbieżności):

$$\frac{1}{T}\sum_{t=1}^{T} f^*(t), \quad f^*(t) = \max\{f(1), f(2), \ldots, f(t)\}$$

- najwyższa wartość oceny w ostatnim pokoleniu:

$$\max_{i=1,\ldots,N} f_i(T)$$

Optymalizacja z ograniczeniami

- dla problemów **optymalizacji funkcji** zbiór rozwiązań dopuszczalnych $D \in \boldsymbol{R}^n$ wyznaczają warunki:
  - nierównościowe (ograniczenia): $g_k(\vec{x}) \geq 0$, $k = 1, 2, \ldots, K$
  - równościowe (więzy): $h_l(\vec{x}) = 0$, $l = 1, 2, \ldots, L$
- dla problemów **optymalizacji kombinatorycznej** zbiór rozwiązań dopuszczalnych $D \in \bigtimes_{i=1}^{n} \Omega_i$ wynika najczęściej z semantyki dziedziny

Metody uwzględniania ograniczeń

- **eliminacja** osobników reprezentujących rozwiązania niedopuszczalne

  wada: generowanie niepotrzebnych rozwiązań zmniejsza efektywność
- zastosowanie **algorytmów naprawy** rozwiązań niespełniających ograniczeń

  wada: algorytm naprawy może mieć dużą złożoność obliczeniową
- **zapobieganie** generowaniu rozwiązań niedopuszczalnych przez zastosowanie specjalnej reprezentacji i/lub operatorów wariacji

wspólna wada: sztywne ograniczenia mogą utrudniać proces poszukiwania

Metody uwzględniania ograniczeń – cd.

- zastosowanie **funkcji kary** jako dodatkowego składnika funkcji oceny osobnika

  wada: konieczność określenia funkcji kary (w większości problemów optymalizacji kombinatorycznej jest to praktycznie niemożliwe, bo ograniczenia opisują poprawność syntaktyczną rozwiązań)

  wada: ograniczenia mogą zostać naruszone (np. rozwiązanie o wysokiej jakości blisko granicy poprawności)

Funkcja kary w optymalizacji funkcji

w zadaniu minimalizacji funkcji $f(\vec{x})$ funkcja oceny może mieć postać:

$$f(\vec{x}) + r\sum_{k=1}^{K}\Phi[g_k(\vec{x})]$$

gdzie r jest współczynnikiem kary, a Φ funkcją kary, np.:

$$\Phi[g_k(\vec{x})] = \begin{cases} g_k^2(\vec{x}) & \text{jeśli } g_k(\vec{x}) < 0 \\ 0 & \text{w przeciwnym wypadku} \end{cases}$$

(analogicznie dla więzów)

Część V

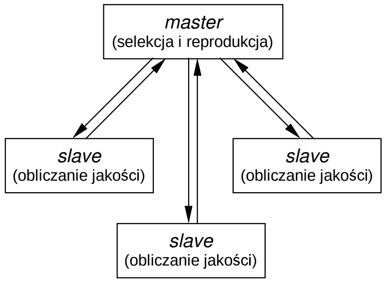
Techniki zaawansowane

Równoległe algorytmy ewolucyjne

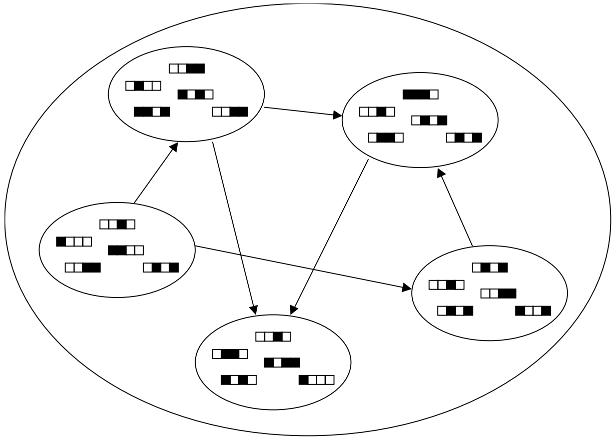
- model *master-slave*, inaczej hodowlany (ang. *farming*)
- model wyspowy (ang. *island*), inaczej migracyjny lub gru-bozioarnisty
- model dyfuzyjny (ang. *diffusion*), inaczej komórkowy (ang. *cellular*) lub drobnozioarnisty
- modele hybrydowe

Model *master-slave*

- równoległe obliczanie jakości osobników, względnie działanie operatorów genetycznych
- możliwa realizacja:
  - synchroniczna
  - półsynchroniczna
  - asynchroniczna
- ze względu na duże obciążenie komunikacyjne ma sens tylko wtedy, gdy zrównoleglane operacje są bardzo złożone



Model wyspowy



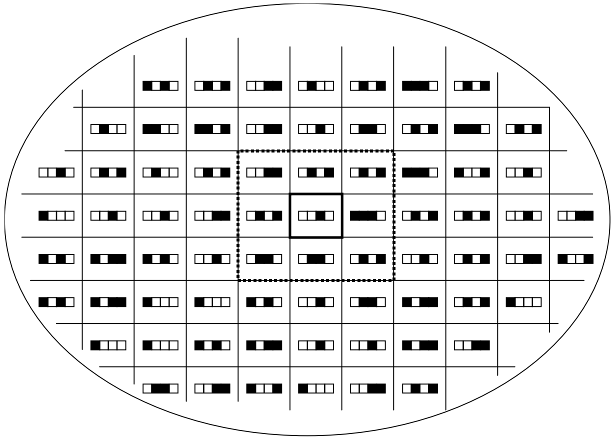
Model wyspowy – działanie

- podział populacji między określoną liczbę obszarów (wysp), w ramach których proces ewolucji przebiega niezależnie
- istotna rola migracji osobników pomiędzy wyspami – choć jej natężenie powinno być niewielkie
- stosowane różne strategie migracji w zależności od:
  - dróg połączeń między wyspami
  - sposobu wyboru emigrujących osobników
  - czasu wykonywania (migracja synchroniczna i asynchroniczna)
  - faktu pozostawiania kopii emigrującego osobnika

Model wyspowy – zalety

- intensywne przeszukiwanie przestrzeni rozwiązań (bo poszukiwania prowadzone są niejako jednocześnie w wielu punktach) przyspiesza działanie algorytmu i zmniejsza ryzyko przedwczesnej zbieżności
- dobre odzwierciedlenie rzeczywistego przebiegu ewolucji w środowisku naturalnym
- możliwa efektywna realizacja w systemie rozproszonym

Model dyfuzyjny



Model dyfuzyjny – cd.

- każdy osobnik populacji zajmuje unikalne położenie (adres) w pewnej przestrzeni
- ewolucja osobników zachodzi jedynie w zasięgu lokalnego otoczenia wyznaczonego przez wybraną funkcję sąsiedztwa
- podobnie jak w przypadku modelu wyspowego ma miejsce intensywne poszukiwanie (znalezione dobre rozwiązania wolno rozprzestrzeniają się w strukturze modelu), co przyspiesza działanie algorytmu i zmniejsza ryzyko przedwczesnej zbieżności
- możliwa efektywna realizacja w architekturze równoległej (SIMD)

Ewolucyjne systemy wieloagentowe

- poszukiwanie ewolucyjnego modelu obliczeniowego, stanowiącego możliwie pełną analogię do procesów obserwowanych w naturze
- połączenie paradygmatów agentowych i ewolucyjnych – decentralizacja procesu ewolucji
- wprowadzenie mechanizmów ewolucyjnych do środowiska agentowego, czyli poddanie procesom ewolucyjnym populacji agentów

Ograniczenia klasycznych AE

model ewolucji przyjęty w AE nie odzwierciedla wielu istotnych aspektów naturalnych procesów ewolucyjnych, np.:

- ✓ dynamicznie zmieniające się warunki środowiska
- ✓ jednoczesny wpływ wielu kryteriów
- ✓ brak wiedzy globalnej i synchronizacji pokoleniowej
- ✓ ewolucja gatunków
- ✓ ewoluujące odwzorowanie cech geno- i fenotypowych

najistotniejszymi ograniczeniami klasycznych obliczeń ewolucyjnych wydają się:

1. cały proces ewolucji jest realizowany przy pomocy jednego algorytmu, przez co jest on scentralizowany
2. obiekty podlegające ewolucji są zredukowane do struktur danych (genów), przetwarzanych przez wspólny algorytm

Systemy agentowe

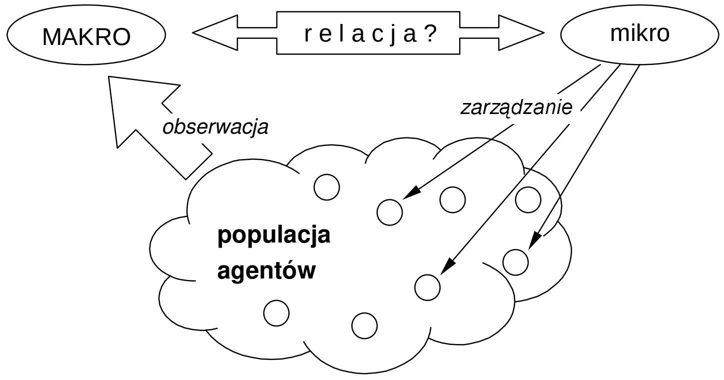
agent – system, który usytuowany jest w pewnym środowisku, i którego jednocześnie jest częścią; agent obserwuje (odbiera, odczuwa) to środowisko oraz działa w nim, w czasie, według własnego planu, wpływając na to, co będzie mógł zaobserwować w przyszłości

najważniejsze cechy: *usytuowanie, autonomia, reaktywność, działanie celowe*

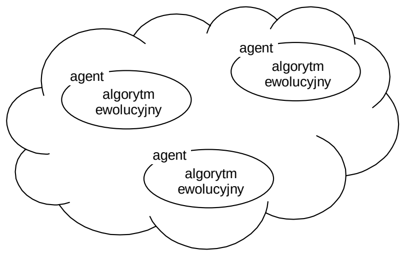
system agentowy – system komputerowy, w którym kluczową abstrakcją jest pojęcie agenta

system wieloagentowy – system komputerowy, składający się z wielu współdziałających agentów

Masowe systemy agentowe

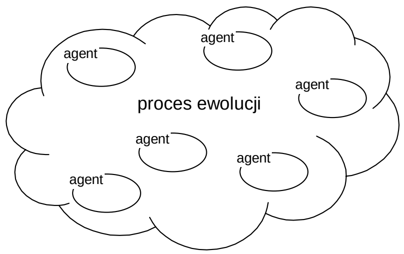


Ewolucja a systemy agentowe



algorytm ewolucyjny wspomagający działanie poszczególnych agentów

ewolucyjny system wieloagentowy (EMAS)



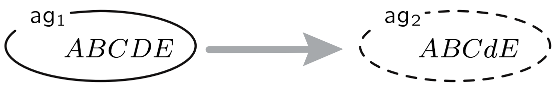
Zasada działania EMAS

- reprezentacja zadania i rozwiązań
- wyposażenie każdego agenta w umiejętność reprodukcji
- mechanizmy selekcji
- przestrzeń systemu agentowego i możliwości migracji

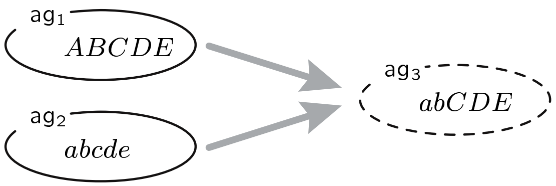
Reprodukcja a dziedziczenie cech

zasada dziedziczenia:

a) z mutacją cech



b) z rekombinacją cech



Mechanizmy selekcji

1. nie mogą zostać przeniesione w niezmienionym kształcie mechanizmy selekcji z klasycznych algorytmow ewolucyjnych, gdyż:
   i. nie można założyć istnienia wiedzy globalnej, ani globalnej synchronizacji działań, zatem nie można dokonać równoczesnej oceny jakości wszystkich agentów
   ii. każdy agent autonomicznie podejmuje decyzje co do wykonywanych akcji i nie ma możliwości bezpośredniego wymuszenia na nim określonego działania
2. należy rozważyć dynamikę liczebności populacji agentów

proponuje się zastosowanie zasady selekcji opierającej się na obecności **wyczerpywalnych zasobów** (w najprostszym przypadku jednego zasobu, nazywanego *energią życiową*)

Energia w EMAS

energia **początkowa** agenta:

$$e_{\mathrm{ag}_i}[t_0] = e_{init} > 0$$

zmiana poziomu energetycznego wiąże się z **wykonywanymi** przez agenta **akcjami**:

$$e_{\mathrm{ag}_i}[t+1] = e_{\mathrm{ag}_i}[t] + \Delta e^{+}_{\mathrm{ag}_i}[t] - \Delta e^{-}_{\mathrm{ag}_i}[t]$$

gdzie: $\Delta e^{+}_{\mathrm{ag}_i}[t]$ – całkowita energia otrzymana w kroku t

$\Delta e^{-}_{\mathrm{ag}_i}[t]$ – całkowita energia utracona w kroku t

prawdopodobieństwo **reprodukcji**: $p^{g}_{\mathrm{ag}_i}[t] = p^{g}(e_{\mathrm{ag}_i}, t)$

prawdopodobieństwo **śmierci**: $p^{e}_{\mathrm{ag}_i}[t] = p^{e}(e_{\mathrm{ag}_i}, t)$