

**Algorytmy genetyczne.** Idea algorytmów genetycznych została wprowadzona do informatyki pod koniec lat 60-tych XX wieku przez J. Hollanda [Holland 1968 i 1975]. Kierowany przez niego zespół już na początku badań postawił sobie dwa, dość ambitne, cele: 1) opisać i wyjaśnić istotę procesów adaptacyjnych w przyrodzie, 2) stworzyć użyteczne oprogramowanie działające na wzór systemów biologicznych. Jakkolwiek cele te są dalekie od pełnej realizacji, techniki ewolucyjne doczekały się wielu zastosowań, z których większość mieści się w formule sztucznej inteligencji. Poniżej przedstawimy podstawy algorytmów genetycznych, kładąc nacisk na fakt, że realizujące je programy komputerowe można uznać za pewne narzędzie do rozwiązywania problemów<sup>42</sup>.

Zasady działania programów realizujących algorytmy genetyczne - nazywanych też *programami ewolucyjnymi* - nawiązują do znanych z przyrody mechanizmów reprodukcji, dziedziczenia cech i doboru naturalnego. W przeciwieństwie do programu tradycyjnego, który zmierza do rozwiązania problemu krok po kroku, zgodnie z zadaną strategią, **program ewolucyjny generuje całe populacje rozwiązań i sprawdza ich dopasowanie do wymogów problemu**. Generowanie kolejnych populacji odbywa się na zasadach przypominających proces ewolucji: rozwiązania mutują, wymieniają między sobą informacje, rozmnażają się, giną. Spośród nich selekcyjnie się wyróżniają wyniki coraz bardziej zbliżone do oczekiwanego - rolę środowiska, do którego w teorii ewolucji przystosowują się przedstawiciele danego gatunku przejmują specjalna funkcja oceny próbnych rozwiązań.

Prześledźmy sposób działania programu ewolucyjnego dokładniej. Załóżmy, że dany jest problem P oraz pewna przestrzeń jego potencjalnych rozwiązań. Rozwiązania te nazywa się *chromosomami* i koduje najczęściej w postaci ciągów zerjedynkowych. Program rozpoczyna działanie od pewnej grupy chromosomów (przeważnie wybranych losowo) określanej mianem *populacji początkowej*. W kolejnych krokach obliczeń rozpatrywane są pojedyncze populacje, w obrębie których chromosomy ewoluują. Oznacza to, że są poddawane działaniu operatorów genetycznych, takich jak *mutacja*, *krzyżowanie* i *inwersja*. W przypadku zerjedynkowej reprezentacji chromosomów mutacja polega na zamianie losowo wybranych zer na jedyneki i odwrotnie, inwersja oznacza odwrócenie kolejności w wybranych sekwencjach zer i jedynek, zaś krzyżowanie odpowiada rozmnażaniu - dwa losowo wybrane chromosomy dzielą się na części, z których jest składane potomstwo. W wyniku takich lub podobnych operacji w populacji pojawiają się nowe osobniki, które bądź są do niej dopisywane, bądź zastępują chromosomy dotychczas w niej istniejące. Każdy z chromosomów nowopowstałej populacji zostaje oceniony pod kątem dopasowania do problemu P. Ocena ta staje się podstawą dla przeprowadzenia selekcji, która ogólnie rzecz biorąc polega na wyborze z populacji bieżącej chromosomów o największej funkcji oceny. Dokładniej zaś, o wyborze decyduje

<sup>42</sup> Algorytmy genetyczne opracowano na podstawie książek [Goldberg 1989 (1989)] i [Michalewicz 1992].

prawdopodobieństwo: chromosomy lepsze są wybierane z prawdopodobieństwem większym, chromosomy gorsze z mniejszym, a zatem im wyższa funkcja oceny chromosomu tym szerszą reprezentację zyskuje on wśród wyselekcjonowanych. Efektem selekcji jest nowa populacja, która podobnie jak jej poprzedniczka zostanie przetworzona w kolejnym kroku obliczeń. Program ewolucyjny kończy pracę, kiedy zostanie spełnione pewne kryterium finalne, na przykład, w populacji pojawi się osobnik o wystarczająco dużej funkcji oceny lub średnia ocena populacji osiągnie pewną zadawalającą wartość.

Przedstawiony wyżej schemat opisuje szeroki wachlarz algorytmów genetycznych. Podstawowe różnice pomiędzy nimi dotyczą sposobu kodowania potencjalnych rozwiązań, operacji selekcji oraz operatorów odpowiadających za przetwarzanie chromosomów. Przyjrzyjmy się wybiórczo różnym odmianom tych składników algorytmu.

Stosunkowo najczęściej wykorzystywaną i najlepiej opisaną teoretycznie metodą kodowania chromosomów jest *notacja binarna*. Przyczyna jej popularności tkwi w istnieniu teorii (tzw. teorii schematów Hollanda), która uzasadnia zbieżność algorytmów operujących na ciągach zerojedynekowych, pozwala oszacować jej szybkość oraz wskazuje na możliwe przeszkody w poprawnym działaniu algorytmów. Mimo braku teorii opisującej inne metody kodowania w wielu zastosowaniach wygodniej jest operować na całych fragmentach chromosomów, które mogą być reprezentowane przez liczby rzeczywiste.

Niezależnie też, prowadzi się badania nad wprowadzeniem do algorytmów genetycznych występującej w przyrodzie *diploidalności*. Chromosomy są przetwarzane i dziedziczone jako dwa komplementarne łańcuchy genów, wśród których występują geny dominujące i recesywne. Przy ocenie chromosomu bierze się jednak pod uwagę tylko wypadkową informacji obydwu łańcuchów, czyli osobnika, w którym jedne geny się ujawniły, a inne pozostały ukryte. Metoda taka zapobiega zapomnieniu przez chromosomy pewnych informacji, które okazały się użyteczne we wcześniejszych etapach przetwarzania.

Operacja selekcji zazwyczaj opiera się na tzw. *regule ruletki*. W pojedynczym kroku algorytmu poszczególnym chromosomom są przypisywane prawdopodobieństwa selekcji równe wynikom dzielenia ocen poszczególnych chromosomów przez sumę ocen wszystkich chromosomów. Prawdopodobieństwa te można sobie wyobrazić jako mniejsze lub większe wycinki koła ruletki o polu równym jedności. Koło jest puszczone w ruch tyle razy, ile chromosomów liczy populacja. Za każdym razem, kiedy wskaźnik ruletki zatrzyma się na danym wycinku, do nowej populacji jest dołączany osobnik odpowiadający temu wycinkowi. Losowy charakter całej procedury powoduje, że niektóre osobniki mogą zostać wybrane kilkukrotnie, a inne w ogóle.

Opisana metoda doczekała się już wielu modyfikacji, które polegają z grubsza na pewnych „udoskaleniach” losowego charakteru selekcji. Najprostsza z nich ogranicza się do każdorazowego dołączania do populacji chromosomu o najwyższej funkcji oceny. W bardziej zaawansowanych modyfikacjach do populacji dołącza się oczekiwane liczby kopii poszczególnych osobników (wyz-

naczone na podstawie ich prawdopodobieństw selekcji), a dopiero pozostałe miejsca obsadza się według zasady ruletki.

Populacja, wśród której dokonuje się selekcji zawiera osobniki, które powstały poprzez wykonanie na innych osobnikach pewnych operacji rekonfigurujących. Do najbardziej typowych zalicza się opisane wcześniej krzyżowanie, mutację i inwersję. Częstotliwość ich wykonywania regulują odpowiednie prawdopodobieństwa. Na przykład, jeśli prawdopodobieństwo krzyżowania wynosi 0,3, to wybór chromosomów do tej operacji polega na losowaniu dla każdego chromosomu liczby z przedziału  $[0,1]$  i sprawdzaniu, czy jest ona mniejsza od 0,3. Jeśli tak się zdarzy, chromosom będzie brał udział w krzyżowaniu. Prawdopodobieństwa operacji rekonfigurujących albo stanowią niezmienny element algorytmu (przypadek najczęstszy), albo są zmieniane globalnie wraz z postępiami algorytmu, albo są dołączane do poszczególnych chromosomów i podlegają ewolucji.

Osobnym kierunkiem rozszerzeń tradycyjnego algorytmu genetycznego jest modelowanie gatunków. Począwszy od pewnego etapu ewolucji chromosomy dzieli się na grupy osobników podobnych, zaś ewolucja odbywa się tylko w ramach grup. Dzięki temu algorytm może poszukiwać jednocześnie wielu lokalnych ekstremów funkcji oceny.

Na koniec podkreślmy jeszcze raz, że cel konkretnego algorytmu wyznacza funkcja oceny. Z punktu widzenia analogii biologicznych funkcja ta modeluje środowisko, w którym podlega selekcji materiał genetyczny. Z punktu widzenia zastosowań definiuje ona najbardziej istotne cechy rozwiązywanych problemów.

Przedstawione wyżej opisy zobrazujemy zwięzłym schematem standardowego algorytmu genetycznego, uzupełnionym o schematy procedury selekcji i dwóch najbardziej typowych operacji rekombinujących (tj. mutacji i krzyżowania).

### STANDARDOWY ALGORYTM GENETYCZNY

**Parametry algorytmu:** metoda kodowania rozwiązań, funkcja ich oceny, metoda selekcji, prawdopodobieństwa mutacji i krzyżowania, warunek końcowy.

**Oznaczenia:**  $C_1, \dots, C_n$  - chromosomy;  $P_t = \{C_1, \dots, C_n\}$  - populacja w kroku  $t$ ;  $O(C_i)$  - ocena  $i$ -tego chromosomu;  $m, p_K$  - prawdopodobieństwa odpowiednio: mutacji i krzyżowania;  $KR, SEL$  - zbiory chromosomów wybranych odpowiednio: do krzyżowania i selekcji.

#### ALGORYTM

1. Stwórz populację początkową  $P_t$ ;  $t = 0$
2. Dopóki nie jest spełniony warunek końcowy wykonuj 3 ...6
  3. Przetwórz ( $P_t$ ); np. Wykonaj mutacje i krzyżowania.
  4. Dokonaj oceny ( $P_t$ )

5. Dokonaj\_Selekcji ( $P_t$ )
6.  $t = t + 1$
7. Rozwiązaniem problemu jest najlepszy chromosom z  $P_t$

## MUTACJA

1. Dla  $i = 1, \dots, n$  wykonuj kroki 2 i 3
  2. Generuj losowo liczbę  $r$  z przedziału  $[0,1]$
  3. Jeśli ( $r < p_M$ ), dokonaj mutacji w  $C_i$

## KRZYŻOWANIE

1. Dla  $i = 1, \dots, n$  wykonuj kroki 2 i 3
  2. Generuj losowo liczbę  $r$  z przedziału  $[0,1]$
  3. Jeśli ( $r < p_K$ ), wybierz  $C_i$  do krzyżowania ( $KR = KR \cup C_i$ )
  4. Skrzyżuj kolejne pary chromosomów z  $KR$

## SELEKCJA

1. Oblicz  $O(C_i)$ ,  $i = 1, \dots, n$ . Oblicz  $F = \sum_{i=1..n} O(C_i)$
2. Oblicz prawdopodobieństwa wyborów  $p_i = O(C_i)/F$
3. Oblicz prawdopodobieństwa łączne  $q_i = \sum_{j=1..i} p_j$
4. Dla  $i = 1, 2, \dots, n$  wykonuj kroki 5,6,7
  5. Generuj losową liczbę  $r$  z przedziału  $[0,1]$
  6. Szukaj  $j$  spełniającego warunek:  $q_j \leq r < q_{j+1}$
  7.  $SEL = SEL \cup C_j$

## 2.3 Logicyzm czy naturalizm?

### 2.3.1 Inspiracje logiczne i empiryczne

Omówione w poprzednim podrozdziale techniki przetwarzania danych zostały dobrane tak, aby ukazać zasadniczą odmienną dwóch kierunków badań nad sztuczną inteligencją. Kierunek pierwszy - reprezentowany m.in. przez techniki regułowe - obejmuje wszelkie próby automatyzacji procesów intelektualnych w kategoriach logiki. Proponujemy nazwać go *logicyzmem*. Kierunek drugi - reprezentowany jest m.in. przez algorytmy genetyczne - mieści w sobie wszelkie koncepcje inspirowane obserwacją przyrody, bez przyjmowania apriorycznego założenia o jej zależności od praw logiki. Proponujemy opatrzyć go etykietą *naturalizmu*<sup>43</sup>.

<sup>43</sup> Termin *naturalizm* ma bardzo wiele znaczeń. W tym wypadku wybrano go, aby podkreślić fakt, że pewne badania informatyków odwołują się do naturalnych mechanizmów wnioskowania czy przetwarzania informacji, które już istnieją w przyrodzie. Mechanizmy te opisuje np. biologia i psychologia. Zwracamy też uwagę, że terminów *logicyzm* i *naturalizm* używaliśmy już w rozdziale 1 (np. w kontekście typologii systemów informatycznych); dopiero tu jednak terminy te precyzujemy.