

Automatyczny dobór parametrów algorytmu genetycznego

Remigiusz Modrzejewski

22 grudnia 2008

Plan prezentacji

Wstęp

Atrakcyjność

Pułapki

Klasyfikacja

Algorytmy niezwiązane

Zarys ogólny

Miary jakości

Przykład algorytmu

Algorytmy ściśle związane

Zarys ogólny

Konstrukcja algorytmu

Przykład algorytmu

Atrakcyjność algorytmów genetycznych

Algorytmy genetyczne są atrakcyjne przede wszystkim w tych problemach optymalizacji, gdzie nie można łatwo zastosować metod wyspecjalizowanych, zaś przestrzeń rozwiązań jest zbyt duża i skomplikowana dla innych algorytmów szukania. Tak więc dwie kluczowe ich cechy to *wysoka niezależność od postawionego problemu* oraz *relatywnie szybka zbieżność do minimum globalnego*.

Własności te utrzymują dzięki:

- ▶ poszukiwaniu w wielu kierunkach – zapewnionemu przez różnorodność populacji oraz losowe mutacje
- ▶ dobremu stosunkowi eksploracji/eksploatacji – im bliżej jesteśmy rozwiązania, tym większa szansa, że nowe osobniki będą jemu bliskie

Pułapki w algorytmach genetycznych

Jak we wszystkich algorytmach optymalizacji, największym zagrożeniem jest możliwość **utknięcia w ekstremum lokalnym**. Najprostszą metodą przeciwdziałania jest zwiększenie populacji oraz częstotliwości mutacji, jednak problemem staje się wtedy **wolna zbieżność**.

Potrzebne jest więc *zapewnienie różnorodności* takie, by po znalezieniu się w obszarze zbliżonym do optimum mieć mimo wszystko *szybką zbieżność*. Dobranie operatorów i ich parametrów takich, by uzyskać oba te efekty, nie jest możliwe w przypadku ogólnym. Przydatna jest bardzo uważna analiza poparta wieloma eksperymentami dla każdego problemu z osobna.

Stąd potrzeba *automatycznego doboru parametrów* oraz *dynamicznego stosunku eksploracji/eksploatacji*. Wszystko to zawarte jest w **adaptacyjnych algorytmach genetycznych**.

Klasyfikacja ze względu na adaptowany aspekt

Adaptować w algorytmie genetycznym możemy:

- ▶ parametry operatorów
- ▶ parametry populacji
- ▶ wybór operatorów
- ▶ reprezentację
- ▶ funkcję przystosowania

Klasyfikacja Spearsa

William Spears zaproponował następującą klasyfikację adaptacyjnych algorytmów genetycznych:

- ▶ **Niezwiązane** – istnieje jakiś mechanizm centralnego sterowania, który dokonuje adaptacji algorytmu genetycznego. Z jednej strony jest to łatwe i wydajne podejście do problemu. Z drugiej jednak działa wbrew idei algorytmu genetycznego.
- ▶ **Luźno związane** – gdzie mechanizm sterowania zależny jest od populacji bądź operatorów genetycznych.
- ▶ **Ściśle związane** – gdzie adaptacja algorytmu genetycznego dokonywana jest przez ten algorytm genetyczny.

Plan prezentacji

Wstęp

Atrakcyjność

Pułapki

Klasyfikacja

Algorytmy niezwiązane

Zarys ogólny

Miary jakości

Przykład algorytmu

Algorytmy ściśle związane

Zarys ogólny

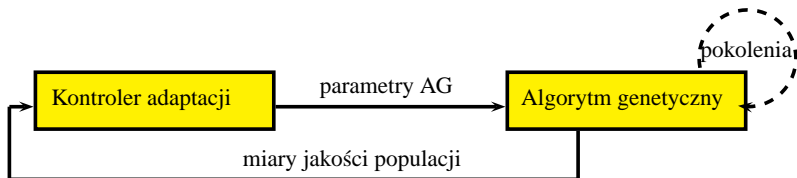
Konstrukcja algorytmu

Przykład algorytmu

Schemat



Schemat



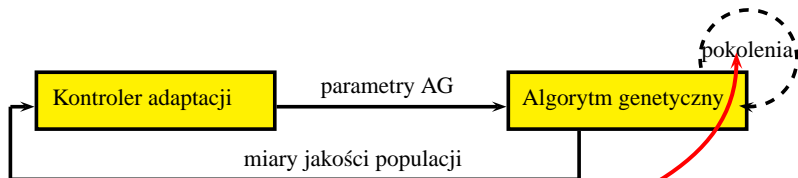
- ▶ Kontrolerem adaptacji powinna być jakaś możliwie prosta i odporna funkcja. Ze względów praktycznych najłatwiej użyć tutaj zestawu intuicyjnych reguł w *logice rozmytej*.

Schemat



- ▶ Kontrolerem adaptacji powinna być jakaś możliwie prosta i odporna funkcja. Ze względów praktycznych najłatwiej użyć tutaj zestawu intuicyjnych reguł w *logice rozmytej*.
- ▶ Kontroler ustawia parametry algorytmu genetycznego na podstawie pewnych *miar jakości populacji*.

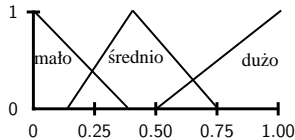
Schemat



- ▶ Kontrolerem adaptacji powinna być jakaś możliwie prosta i odporna funkcja. Ze względów praktycznych najłatwiej użyć tutaj zestawu intuicyjnych reguł w *logice rozmytej*.
- ▶ Kontroler ustawia parametry algorytmu genetycznego na podstawie pewnych *miar jakości populacji*.
- ▶ Aby uniknąć nadsterowności, parametry dostosowywane są co pewną liczbę pokoleń. Nie zbadano jeszcze możliwości adaptacji tej liczby.

Reguły logiki rozmytej

Najprostszym koncepcyjnie kontrolerem adaptacji jest zestaw intuicyjnych reguł. Jednak miary jakości dają wartości ciągłe i trudno określić dokładne progi dla kolejnych sytuacji, używa się więc logiki rozmytej:



1. Przekształć wartości na rozmyte.
2. Zastosuj na nich formuły logiczne typu:
 $\text{miara1} = \text{dużo} \wedge \text{miara2} = \text{mało}$
 \implies zwiększ parametr1 o *dużo*
3. Przekształć otrzymane rozmyte wnioski z powrotem na wartości ciągłe.

Genotypowe miary jakości

Genotypowe miary jakości pozwalają nam przede wszystkim przekonać się o różnorodności osobników w populacji. Operują bezpośrednio na genach i są to:

► Dla alleli dyskretnych oparte na:

1. Częstotliwości alleli na pozycji i , $D = \overline{1 - 4(0.5 - freq_i)}$, gdzie $freq_i$ to częstotliwość występowania 0 na pozycji i .
2. Liczbie straconych alleli w histogramie. Trzeba sprogować dostępne wartości w kilka słupków, następnie zaś wyliczyć z histogramu miarę:

$$AL = \sum_i \sum_j |\{x | hist(i, j) = 0\}|$$

gdzie i wylicza kolejne pozycje, a j kolejne słupki

3. Średniej odległości Hamminga.

- ▶ Dla alleli ciągłych oparte na:
 1. Odległości Euklideskiej:

$$ED = \frac{\overline{d(C_{max}, C_i)}}{\max\{d(C_{max}, C_i)\}}$$

gdzie C_{max} - chromosom chwilowo najlepszy

2. Różnie pojętej wariancji. Na przykład:
 - ▶ Wariancja uśrednionych chromosomów:

$$VAC = \overline{(\bar{S}_i - \bar{S})^2}$$

- ▶ Średnia wariancja alleli:

$$AVA = \overline{(S_{ij} - \bar{S}_j)^2}$$

W obu przypadkach i wylicza chromosomy a j pozycje alleli.

3. Entropii.

Fenotypowe miary jakości

Miary fenotypowe odnoszą się do jakości populacji. Operują one na wynikach funkcji przystosowania, która jest pojedynczą wartością ciągłą.

- ▶ Miary pośrednie nie korzystają z samych wartości funkcji przystosowania. Zamiast tego obserwują na przykład jaka część populacji pozostawia po sobie potomka.
- ▶ Miary bezpośrednie operują na różnych statystykach wartości funkcji przystosowania. Przykładowe dwie miary sprawdzające rozrzut przystosowania:

$$PDM_1 = \frac{f_{\max}}{\bar{f}}; PDM_2 = \frac{\bar{f}}{f_{\min}}$$

ARGAF

Za przykład posłużymy nam algorytm ARGAF autorstwa Francisco Herrera i Manuela Lozano. Jego główne założenia to:

- ▶ Użyte dwa operatory krzyżowania, jeden służący eksploracji, drugi eksploatacji.
- ▶ Selekcja rankingowa.
- ▶ Przyjęte stałe p_c i p_m . Adaptację osiągnięto przez automatyczny dobór p_e – prawdopodobieństwo wyboru operatora krzyżowania do eksploatacji, oraz η_{\min} określającego nacisk selekcyjny.
- ▶ Wejściami dla kontrolera są miary ED i PDM_1 .

Reguły i rezultaty ARGAF

p_e	PDM_1		
ED	Mało	Średnio	Dużo
Mało	Mało	Mało	Średnio
Średnio	Dużo	Dużo	Średnio
Dużo	Dużo	Dużo	Średnio

η_{min}	PDM_1		
ED	Mało	Średnio	Dużo
Mało	Mało	Średnio	Dużo
Średnio	Mało	Dużo	Dużo
Dużo	Mało	Mało	Dużo

Wyniki działania ARGAF zostały porównane z kilkoma wersjami algorytmu nie adaptacyjnego o różnych ustalonych parametrach, dla coraz to trudniejszych funkcji. Dla różnych funkcji różne zestawy parametrów okazywały się najlepsze. Przy każdej z nich ARGAF osiągał wyniki porównywalne z najlepszym z ręcznie ustawionych parametrów.

Plan prezentacji

Wstęp

Atrakcyjność

Pułapki

Klasyfikacja

Algorytmy niezwiązane

Zarys ogólny

Miary jakości

Przykład algorytmu

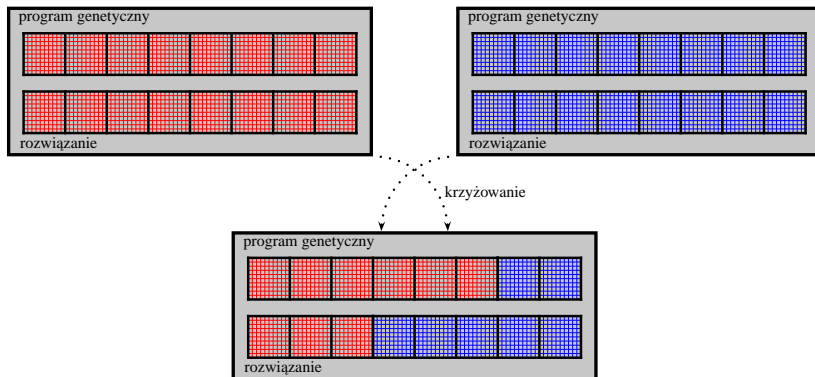
Algorytmy ściśle związane

Zarys ogólny

Konstrukcja algorytmu

Przykład algorytmu

Schemat



Założenie: proces który doprowadził do powstania najlepszych osobników jest również najlepszy. Każdy osobnik składa się z dwóch chromosomów. Poza rozwiązaniem problemu (podlegającym funkcji oceny) jest też zakodowany program genetyczny kontrolujący rozmnażanie.

Program genetyczny

- ▶ Zakodowany ciąg instrukcji, składających się z operatorów genetycznych i ich parametrów.
- ▶ Można przyjąć, że w ciągu występują po kolei wszystkie możliwe operatory i adaptować tylko ich parametry – oszczędność obliczeń.
- ▶ Można przyjąć, że część pól ustalonych na początku będzie zawsze dobra – wprowadzamy maski modyfikowalności (jeśli pole ma ustawioną maskę, to nie można go modyfikować).
- ▶ Można wprowadzać do instrukcji pola oznaczające którego chromosomu operator dotyczy.
- ▶ Można wprowadzać dużo bardziej szczegółowe określenia, jak na przykład na jakim zakresie locus można mutować.
- ▶ Można wprowadzać rzadko w innych podejściach operatory: dziedziczenie po dużych liczbach, hipermutacje...

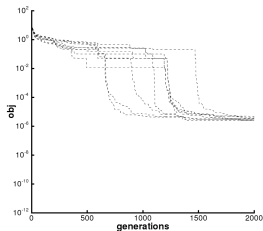
Przykład algorytmu ściśle związanego

Za przykład posłużymy nam algorytm autorstwa D. Quagliarelly i A. Viciniego.

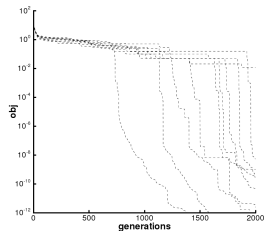
- ▶ Problemem jest minimalizacja funkcji o rozwiązaniu zakodowanym binarnie (kod Graya).
- ▶ Program genetyczny składa się tylko z 1 prostego krzyżowania (jednopunktowego), mutacji pojedynczego bitu i hipermutacji.
- ▶ Krzyżowanie występuje ze stałym prawdopodobieństwem 70%.
- ▶ Adaptacji podlega wyłącznie prawdopodobieństwo mutacji. Jest ono również ograniczone z góry i z dołu.
- ▶ Hipermutacja jest to operacja zwiększenia prawdopodobieństwa mutacji. Testowane były wersje z hipermutacją stałą, jak również zależną od odległości do najbliższego osobnika.

Porównanie wyników

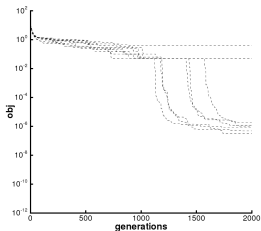
Stałe p_m :



Adaptowane p_m z hipermutacją:



Adaptowane p_m bez hipermutacji:



Wniosek: Adaptacja pozwala osiągnąć lepsze wyniki, jednak bez hipermutacji ma tendencję do wpadania w minima lokalne.

Opracował: Remigiusz 'lRem' Modrzejewski

Prezentacja przygotowana w ramach przedmiotu
Systemy uczące się na Katedrze Inżynierii Wiedzy.

Prezentacja dostępna pod adresem:
<http://lrem.net/pages/view/algorithms>

Prezentacja wykonana w całości za pomocą systemu L^AT_EX.