



Katedra Wytrzymałości Materiałów
i Metod Komputerowych Mechaniki
www.kwmimkm.polsl.pl

Wydział Mechaniczny Technologiczny
Politechnika Śląska

OBLICZENIA EWOLUCYJNE

LABORATORIUM 1: Program *Evolutionary Algorithms*

opracował: dr inż. Witold Beluch
witold.beluch@polsl.pl

Cel ćwiczenia

Wykonując ćwiczenia laboratoryjne zapoznasz się z możliwościami programu *Evolutionary Algorithms* w zakresie obliczeń ewolucyjnych. Nauczysz się między innymi: wprowadzać funkcję celu, wprowadzać ograniczenia na zmienne, dokonywać wyboru operatorów ewolucyjnych i ustawiać ich parametry. Ponadto wykonasz obliczenia dla różnych ustawień programu i zapoznasz się z możliwościami programu w zakresie prezentacji wyników obliczeń.

Program *Evolutionary Algorithms*

Algorytmy genetyczne (AG) i ewolucyjne (AE) należą do grupy tzw. algorytmów optymalizacji globalnej, tzn. są algorytmami wysoce odpornymi na ekstrema lokalne. Do ich działania nie jest potrzebna informacja o gradiencie funkcji celu (przystosowania), która jest często trudna bądź nawet niemożliwa do uzyskania. Z kolei praca na populacji (zbiornie) potencjalnych rozwiązań skutkuje wielokierunkowym przeszukiwaniem przestrzeni poszukiwań, co ułatwia znalezienie optimum globalnego. Algorytmy te bazują na sztucznej ewolucji tzw. osobników, będących potencjalnymi rozwiązaniami zadania optymalizacji. W ramach ewolucji promowane i częściej powielane są osobniki o wyższym przystosowaniu (procedura selekcji) zaś nowe osobniki są tworzone z zastosowaniem operatorów ewolucyjnych, zazwyczaj w postaci różnych wariantów operatorów krzyżowania i mutacji. Więcej na temat AG i AE dowiesz się na wykładzie z tego przedmiotu.

Program *Evolutionary Algorithms* jest programem komputerowym autorstwa Piotra Orantka realizującym obliczenia ewolucyjne jak również umożliwiającym pracę ze sztucznymi sieciami neuronowymi. Wersja 7.07 programu jest dostępna wyłącznie jako anglojęzyczna. Program można pobrać ze strony Katedry (adres strony znajduje się na okładce niniejszej instrukcji). Program nie wymaga instalacji. Jeśli program odmawia współpracy należy sprawdzić, czy w ustawieniach systemu Windows jako separator dziesiętny jest ustawiona kropka (domyślny dla polskich ustawień regionalnych przecinek może powodować błędy w programie).

W ramach zapoznania się z programem będziemy rozpatrywać poniższe zadanie optymalizacji.

Zadanie

Znajdź maksimum¹ globalne funkcji dwóch zmiennych:

$$f(x, y) = 10 / (1 + \text{sqr}((x-4)*(x-4) + (y-6)*(y-6)))$$

gdzie **sqr** (*square*) oznacza podniesienie do kwadratu. Nie pomył tej funkcji z podobną – **sqrt** – oznaczającą pierwiastek kwadratowy (*square root*)...

Ograniczenia na zmienne:

$$0 \leq x \leq 10, \quad 0 \leq y \leq 10.$$

Parametry algorytmu ewolucyjnego:

Używane (aktywne) operatory ewolucyjne:

- mutacja równomierna (*uniform mutation*), type: linear, $p_{um}=0.07$;

¹ Algorytmy genetyczne i ewolucyjne z założenia rozwiązują zadania maksymalizacji funkcji. Jeśli konieczna jest minimalizacja pewnej funkcji $f(x)$, to należy maksymalizować funkcję $g(x) = -f(x)$, by po znalezieniu ekstremum zmienić jej znak:

$$\min f(x) = \max g(x) = \max \{-f(x)\}$$

- krzyżowanie proste (*simple crossover*), type: linear, $p_{sc}=0.3$;

Inne:

- funkcja kary: kara śmierci (*death penalty*);
- selekcja: turniejowa (*tournament selection*), $p_t=0.5$.

Liczebność populacji: 25 osobników. Należy wykonać kilkukrotnie obliczenia dla liczby pokoleń (pętli algorytmu) równej 10, 20 oraz 100.

Zaczynamy



Uruchom program *Evolutionary Algorithms*. Jeśli pojawi się okienko z informacją o programie naciśnij *OK*.

1. Ograniczenia. Przejdź do menu *Data*:

- Sprawdź w *Number of variables*, czy liczba zmiennych projektowych wynosi 2; jeśli tak nie jest, to wyniki mogą być błędne (wykres funkcji wyświetlany jest dla pierwszych 2 zmiennych, obliczenia zaś przeprowadzane są dla wszystkich zmiennych).
- W pozycji *Constraints* ustaw odpowiednie ograniczenia na zmienne x (zmienna 1) i y (zmienna 2).
- Sprawdź (i ewentualnie usuń) czy w pozycjach *Linear constraints* oraz *Nonlinear constraints* nie ma aktywnych ograniczeń liniowych i nieliniowych. Zauważ, że górne pole to tylko baza danych, aktywne ograniczenia są (ewentualnie) widoczne w dolnym polu. W obydwu przypadkach powinna w dolnej części okna widnieć informacja, że liczba aktywnych ograniczeń to 0.

2. Funkcja przystosowania (celu). Przejdź do menu *Tools->Function Editor*. Pojawi się okno edytora funkcji.

- Naciśnięcie przycisku *Read* powoduje wczytanie poprzednio zapisanej w pliku (tekstowym) funkcji. Nie naciskaj tego przycisku po wpisaniu własnej funkcji a przed naciśnięciem *Apply*, bo utracisz to, co wpisałaś/eś! Wpisz funkcję (tylko to, co ma kolor niebieski). Zwróć szczególną uwagę na liczbę nawiasów i znaki mnożenia!
- Naciśnij przycisk *Apply* a następnie *Show*. Pojawi się okno wykresu funkcji. Jeśli po naciśnięciu przycisku *Compute* pojawi się błąd, wróć do okna edytora funkcji i popraw błędny zapis funkcji. Jeśli zaś uaktywni się klawisz *Show* – naciśnij go. Powinien się wyświetlić wykres funkcji¹. Możesz zmienić typ wykresu z 3D na warstwicowy (*Graph type*, wymaga ponownego przeliczenia), ustawić opcje koloru i wypełnienia (*Options*) oraz obrócić wykres 3-D lub zmienić dokładność wyświetlania wykresu warstwicowego. Zauważ, że w wersji warstwicowej przemieszczenie kursora nad wykresem powoduje wyświetlanie wartości zmiennych oraz wartości funkcji, co pomaga oszacować wartość funkcji np. w okolicy optimum. Sprawdź działanie poszczególnych opcji. By wyjść z okna wykresu naciśnij *Close*.
- Naciśnij *OK* w oknie edytora funkcji.

3. Opcje programu. Przejdź do menu *Options->Computing Options*.

Sprawdź, czy aktywne są opcje: *Save all* (zapisywanie wyników w każdym pokoleniu) oraz *Random* (losowe tworzenie populacji startowej).

¹ Zauważ, że rozważana funkcja jest ciągła, gładka i jednomodalna (ma jedno ekstremum), zatem stosowanie algorytmów ewolucyjnych czy też innych metod optymalizacji globalnej jest tu mocno dyskusyjne – w takim przypadku szybciej i dokładniej zostanie znalezione rozwiązanie optymalne przy wykorzystaniu dowolnej metody gradientowej...

4. Ustawienia algorytmu. Wybierz *Settings*->*Evolutionary algorithm*.

Skorzystaj z danych w opisie zadania i wykonaj poniższe kroki:

- W ramce *Selection* wybierz rodzaj selekcji. Naciśnij przycisk *Settings* i ustaw odpowiednią wartość prawdopodobieństwa.
- Wybierz (uaktywnij) odpowiednie operatory. Ustaw (*Settings*) typ prawdopodobieństwa na liniowy. W ramce *Probability* wpisz w odpowiednich polach dwukrotnie tę samą wartość (oznacza to, że prawdopodobieństwo zmienia się co prawda liniowo, lecz jego wartości w pierwszym i ostatnim pokoleniu są jednakowe, czyli prawdopodobieństwo jest w efekcie stałe).
- Wybierz odpowiedni rodzaj selekcji. Kara śmierci oznacza, że ewentualnie powstałe (w wyniku działania operatorów ewolucyjnych) osobniki niedopuszczalne są eliminowane z populacji i zastępowane nowymi, co zwykle wymaga powtórzenia działania danego operatora.
- W polu *Population* wstaw odpowiednią wartość liczebności populacji.
- Naciśnij *OK*.

5. Obliczenia i wyniki. Przejdź do menu *Compute*->*Evolutionary algorithm*.

- W ramce *Generations* pozostaw w polu *First generation* numer pierwszego pokolenia równy 1, natomiast jako ostatnie pokolenie (*Last generation*) wpisz liczbę pokoleń z treści zadania.
- W polu *No. of generations without improvement* wpisz 1000 – nie będziemy z tej opcji¹ w ramach niniejszego ćwiczenia korzystać.
- Uruchom obliczenia naciskając *Compute*. Najlepsza wartość znaleziona przez AE podczas działania prezentowana jest (z niezbyt wysoką precyzją) w polu *Best eval in history*. Zauważ, że naciskając ponownie *Compute* uzyskujesz (zazwyczaj) inne wyniki, choć nie były zmieniane żadne parametry. Zapewne pamiętasz, że AE to algorytmy niedeterministyczne (ale nie losowe).
- W celu wyświetlenia obszerniejszej informacji na temat wyników wybierz w głównym menu *Results*->*Complete*. Najistotniejsze informacje zawierają pola *Best fitness function value* (wynik działania AE), *Best chromosome* (wartości zmiennych projektowych) oraz *Found in generation* (w którym pokoleniu znaleziono najlepsze rozwiązanie²).
- Zmieniaj odpowiednio do treści zadania liczbę pokoleń i wykonuj obliczenia.

6. Inne użyteczne funkcje programu.

- Z menu *Tools* wybierz *Neural Networks Editor*. Pojawi się okno edytora sztucznych sieci neuronowych. Ta opcja programu wykracza poza ramy niniejszego przedmiotu, nie będziemy się więc nią dalej zajmować...
- W menu *Compute* wybierz *Selected point in the domain*. Okno, które się otworzy, pozwala m.in. na obliczenie wartości funkcji przystosowania dla dowolnie wprowadzonych zmiennych. Wprowadź (w polu *Value of the variable*) wartości zmiennych, których numery zmieniaj w polu *Variable*. Np. wprowadź: zmienna 1 wartość 4.1, zmienna 2 wartość 3.7. Naciśnij przycisk *Compute*. W polu *Fitness function value* zostanie wyświetlony wynik (wartość funkcji).

¹ Ta opcja pozwala na ustawienie warunku zatrzymania algorytmu w postaci *kryterium minimalnej szybkości poprawy*. W tym przypadku algorytm jest zatrzymywany, jeśli przez podaną liczbę pokoleń nie zostanie znalezione lepsze rozwiązanie.

² Jeśli jest to ostatnie lub jedno z ostatnich pokoleń to może to sugerować, że przyjęto zbyt małą liczbę pokoleń...

- W menu *Results* wybierz *Evolutionary operators*. Będziesz mógł przeanalizować, ile osobników w danym pokoleniu było poddanych działaniu poszczególnych operatorów. Zauważ, że przy krzyżowaniu liczba ta jest zawsze parzysta, a przy mutacji nie. Zastanów się, dlaczego? Naciśnij przycisk *Options* i zapoznaj się z opcjami wykresu.
- W menu *Results* wybierz *Fitness function*. To co widzisz to tzw. krzywe zbieżności, przedstawiające wartości rozwiązań (np. najlepszego, średniego) w kolejnych pokoleniach – podobne wykresy będziesz tworzył i porównywał w ramach sprawozdań z niektórymi ćwiczeń. Naciśnij przycisk *Options* i zapoznaj się z opcjami wykresu.
- W menu *Results* wybierz *Numerical representation*. Pojawi się okno, w którym możesz zaobserwować wartości zmiennych (genów) wszystkich osobników (chromosomów) w dowolnym pokoleniu (zmieniając numer pokolenia w polu *Population*). Zauważ, że niektóre osobniki są identyczne. Zastanów się, dlaczego?
- W menu *Results* wybierz *Graphical representation*. Wciśnij przycisk *Compute* a następnie *Show*. Zmieniaj numer pokolenia w polu *Population* i obserwuj rezultaty. Zauważ, że zazwyczaj wyświetlanych osobników jest mniej, niż wynosi liczebność populacji – to oczywiście wynika z poprzedniego punktu. Porównaj zawartość okna *Numerical representation* z oknem *Graphical representation* (oba okna mogą być jednocześnie otwarte).



Pamiętaj – **zawsze** rozpoczynając pracę z programem należy sprawdzić, czy:

- liczba zmiennych projektowych wynosi 2 (*Data*->*Number of variables*);
- nie ma przypadkowo wprowadzonych ograniczeń liniowych oraz nieliniowych (*Data* -> *Linear/Nonlinear constraints*).

Sprawozdanie

Z niniejszego ćwiczenia laboratoryjnego nie jest wymagane sprawozdanie.



Ważne – w przypadku niektórych ćwiczeń będziesz zapisywał na zajęciach pewne pliki, które będą niezbędne do wykonania sprawozdania. Pamiętaj zatem, by od następnych zajęć mieć z sobą jakiś **nośnik danych** (np. pendrive).

Literatura i źródła

- [1] J. Arabas: Wykłady z algorytmów ewolucyjnych. WNT, Warszawa, 2003.
- [2] Z. Michalewicz: Algorytmy genetyczne + struktury danych = programy ewolucyjne. WNT, Warszawa, 1996.
- [3] L. Rutkowski: Metody i techniki sztucznej inteligencji. PWN, Warszawa, 2006.