

POLITECHNIKA WARSZAWSKA

WYDZIAŁ ELEKTRYCZNY

Michał Sut

4 grudnia 2017

1 Treść zadania

Napisać program rozwiązujący problem komiwojażera (minimalizacja drogi pomiędzy n miastami bez powtórzeń) przy pomocy algorytmu genetycznego. Zastosować reprodukcję przy użyciu nieproporcjonalnej ruletki, operator krzyżowania CX oraz mutację równomierną.

Program powinien umożliwiać użycie różnych wielkości populacji, liczby iteracji, prawdopodobieństwa mutacji.

Program powinien zapewnić wizualizację wyników w postaci wykresów średniego, maksymalnego i minimalnego przystosowania (długości funkcji) dla kolejnych populacji oraz 2 map (o wymiarach 10x10 punktów), na których będą wyświetlane miasta oraz drogi najdłuższa i najkrótsza.

Pokazać działanie programu na danych testowych składających się z 10 miast, opisanych za pomocą współrzędnych na mapie o wymiarach 10x10 punktów.

Dane testowe: miasta:

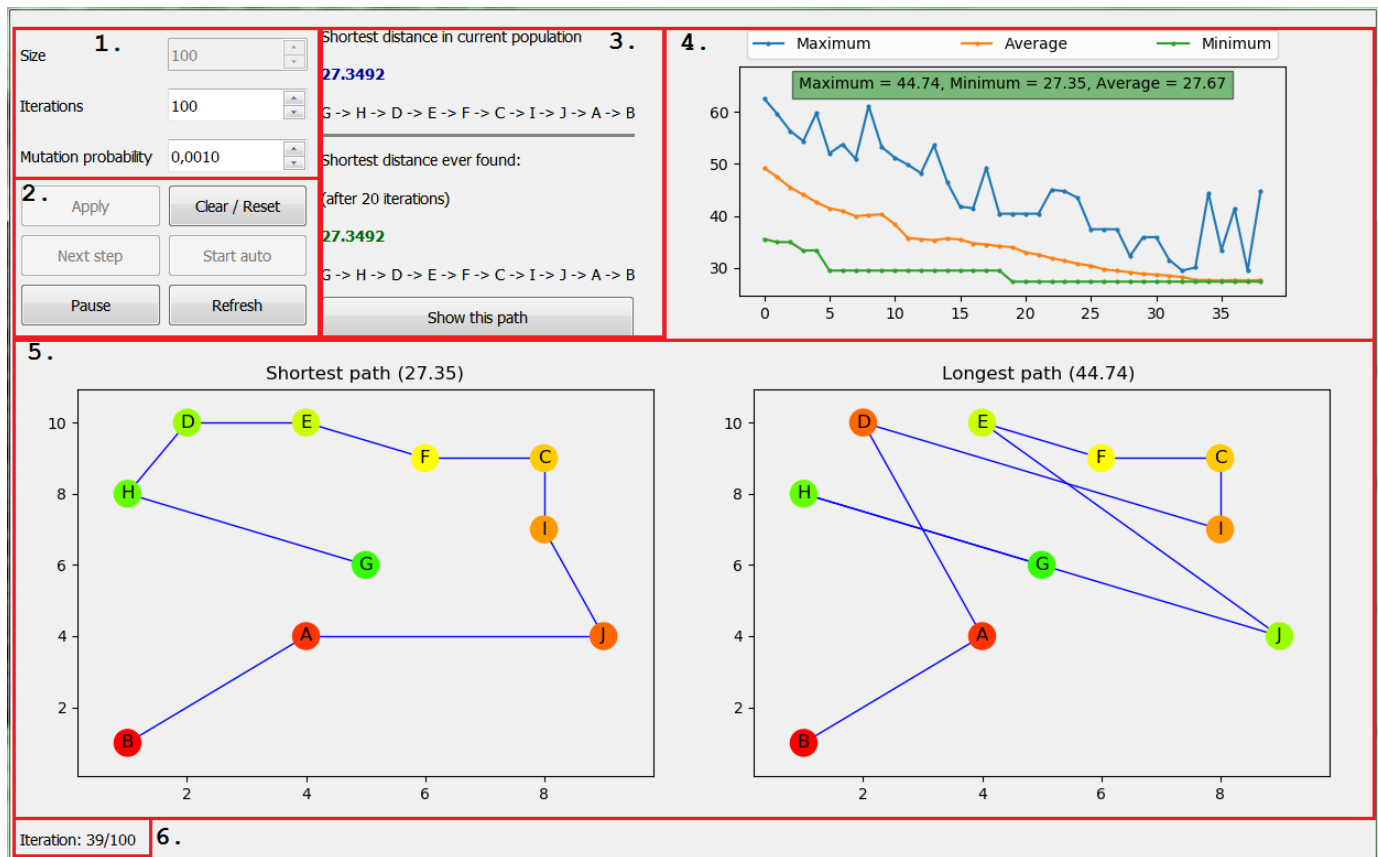
A(4,4), B(1,1), C(8,9), D(2,10), E(4,10), F(6,9), G(5,6), H(1,8), I(8,7), J(9,4)

2 Instrukcja działania programu

2.1 Uruchomienie

Program został napisany w języku Python, z wykorzystaniem bibliotek zawartych w dystrybucji Anaconda3. W związku z tym, należy uruchamiać program w tym środowisku. Można to zrobić poprzez zaimportowanie projektu do programu Visual Studio 2017, wybranie interpretera: Anaconda 4.2 i uruchomienie aplikacji. Plikiem startowym jest `ga.py`

2.2 Okno główne programu



2.3 Opis okna programu

1. Parametry algorytmu
2. Przyciski sterowania
3. Informacje o najkrótszych znalezionych trasach
4. Wykres wartości średnich, maksymalnych i minimalnych dla kolejnych generacji.
5. Mapy z miastami i trasami: po lewej – najkrótsza trasa, po prawej – najdłuższa
6. Informacja o numerze aktualnej iteracji

2.4 Zmiana ustawień programu

Ustawienia programu znajdują się w pliku `settings.py`. Zmianą podlegają następujące elementy:

1. **CITIES** - Nazwy miast
2. **POSITIONS** - Współrzędne miast
3. **COLORS** - Kolory jakimi zaznaczane są kolejne miasta

2.5 Użycie

2.5.1 Podstawowe operacje

1. Ustawić parametry populacji
2. Zainicjować populację klikając przycisk `Apply`

3. W tym momencie dostępne są trzy możliwości:

- (a) - Wykonanie tylko jednej generacji
- (b) - Wykonywanie kolejnych n generacji, gdzie n jest wybraną liczbą iteracji
- (c) - Wyczyszczenie populacji i zresetowanie programu

4. Rozpoczęty proces automatycznych generacji można wstrzymać przyciskiem

5. Aby zobaczyć najkrótszą kiedykolwiek znalezioną trasę należy brać przycisk

2.6 Warunek stopu

Program zatrzymuje się po określonej liczbie iteracji, którą użytkownik określa w panelu ustawień parametrów algorytmu. Liczbę iteracji można zwiększać w trakcie działania programu.

3 Opis eksperymentów

Eksperymenty przeprowadzone były dla miast o następujących współrzędnych: $A(4, 4)$, $B(1, 1)$, $C(8, 9)$, $D(2, 10)$, $E(4, 10)$

3.1 Różna liczebność populacji

W celu przeprowadzenia tych eksperymentów, kilkakrotnie uruchomiony zostanie algorytm z następującymi parametrami: prawdopodobieństwo mutacji – 0,001, liczba iteracji – 200.

3.1.1 Populacja: 10 osobników

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.1.2 Populacja: 50 osobników

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.1.3 Populacja: 120 osobników

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.1.4 Populacja: 600 osobników

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.2 Różne prawdopodobieństwa krzyżowania

Eksperymenty przeprowadzone będą dla populacji 10 osobników, z prawdopodobieństwem mutacji równym 0.001

3.2.1 Krzyżowanie: prawdopodobieństwo = 1,00

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.2.2 Krzyżowanie: prawdopodobieństwo = 0,90

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.2.3 Krzyżowanie: prawdopodobieństwo = 0,50

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.2.4 Krzyżowanie: prawdopodobieństwo = 0,00

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.3 Różne prawdopodobieństwa mutacji

Eksperymenty przeprowadzone będą dla populacji 10 osobników, z prawdopodobieństwem krzyżowania równym 1,00

3.3.1 Mutacja: prawdopodobieństwo = 0,000

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.3.2 Mutacja: prawdopodobieństwo = 0,001

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.3.3 Mutacja: prawdopodobieństwo = 0,004

Wyniki prezentują się następująco:

Lp	Min	Max	Mean	Best
1	31,63	34,72	33,48	31,63
2	35,36	35,36	35,36	35,36
3	35,99	35,99	35,99	34,15
4	31,08	31,08	31,08	31,08
5	31,96	31,96	31,96	31,96
	avg. min:	avg. max:	avg. mean:	avg. best:
	33,20	33,82	33,57	32,84

3.4 Podsumowanie eksperymentów

Eksperyment	średnie maksimum	odchylenie stand. średniej maks.
Populacja: 4	44,10	3,80
Populacja: 10	46,20	1,10
Populacja: 50	46,66	0,42
Populacja: 100	46,77	0,29
Krzyżowanie: 1,00	46,20	1,10
Krzyżowanie: 0,09	45,03	0,88
Krzyżowanie: 0,05	45,011	2,65
Krzyżowanie: 0,00	45,322	2,63
Mutacja: 0,000	41,3	3,49
Mutacja: 0,001	46,20	1,10
Mutacja: 0,004	46,58	0,38

4 Wnioski

Przeprowadzone eksperymenty pokazują, że zmiana parametrów populacji wpływa na znajdowanie maksimum funkcji dopasowania danej populacji.

4.1 Wniosek 1. Dokładność algorytmu

Algorytm nie zawsze znajduje wartość maksymalną. W większości przypadków znajdowane były wartości bliskie oczekiwanej, czyli osobniki, o kodzie zbliżonym to najlepszego. Wynika to z faktu, iż funkcja dopasowania, dla której przeprowadzone były eksperymenty zwraca zbliżone wartości dla osobników o kodzie bliskim najlepszego. Na fakt znalezienia lub nie znalezienia realnej wartości maksymalnej wpływają parametry populacji.

4.2 Wniosek 2. Wpływ liczby osobników w populacji na wynik działania algorytmu

Przeprowadzone eksperymenty pokazują, że statystycznie lepsze wyniki uzyskiwane są dla większych populacji. Największa z badanych przeze mnie populacji dała najwyższy średni wynik z 10 prób, co oznacza najmniejszy błąd względem wartości wyliczonej analitycznie.

Dodatkowo, można stwierdzić, że dla większych populacji wyniki są bardziej jednorodne i skupiają się wokół realnej wartości maksymalnej. Wynika to z faktu, iż najliczebniejsza z badanych populacji zwracała wyniki, których odchylenie standardowe było najmniejsze.

Jednakże, im większa populacja, tym więcej iteracji algorytmu jest potrzebne, aby wystąpił warunek stopu i znaleziona została wartość maksymalna.

4.3 Wniosek 3. Wpływ prawdopodobieństwa krzyżowania na wynik działania algorytmu

Krzyżowanie, jak każdy z parametrów, wpływa na dokładność wyniku, lecz jego wpływ w przeprowadzonym badaniu, nie był tak bardzo widoczny jak wpływ liczebności populacji.

Zmniejszenie prawdopodobieństwa krzyżowania powoduje, że znalezione wartości są bardziej zróżnicowane, czyli odchylenie standardowe wyników wzrasta.

Wykonany eksperyment pokazał, że gdy krzyżowanie jest pewne, średni wynik jest lepszy. Mniejsze wartości powodują wyniki średnio niższe, jednak nie można stwierdzić, że wynik jest proporcjonalny do prawdopodobieństwa krzyżowania.

Wpływ krzyżowania najbardziej widać w sytuacji, gdy w populacji startowej, nie ma osobników o najlepszym dopasowaniu lub zbliżonych do nich. Wyniki wtedy są gorsze, ponieważ tworzy się mniej nowych potencjalnych rozwiązań. Co więcej, mniejsza szansa na krzyżowanie powoduje mniejszą wymaganą liczbę iteracji do zakończenia wykonywania algorytmu

4.4 Wniosek 3. Wpływ prawdopodobieństwa mutacji na wynik działania algorytmu

Eksperyment pokazał, że zwiększenie szansy na mutacje, zwiększa prawdopodobieństwo znalezienia maksimum oraz zwiększa średni wynik z kilku prób. Dodatkowo, zróżnicowanie wyników jest mniejsze dla większej wartości prawdopodobieństwa mutacji. Spowodowane jest to tym, że mutacje znacznie zwiększają liczbę potencjalnych rozwiązań.

Zwiększenie tego parametru ma także negatywny skutek. Liczba iteracji potrzebna do zakończenia algorytmu dla populacji, w której mutacje występują częściej jest większa.