

Podstawy sztucznej inteligencji – projekt A może na wyspę?

Jakub Zieliński i Laura Wrońska

Spis treści

1. Cele projektu	2
2. Interpretacja treści zadania – przyjęte założenia	2
2.1 Model standardowy	2
2.2 Model wyspowy	2
3. Podział zadań w zespole	2
4. Wykorzystane narzędzia i biblioteki	2
5. Kompilacja i uruchomienie testów	2
5.1 Kompilacja	2
5.2 Testy	3
6. Badania	3
6.1 Zbadanie wpływu zmiany rozmiaru populacji na wynik optymalizacji	4
6.2 Zbadanie wpływu siły mutacji	5
6.3 Zbadanie wpływu prawdopodobieństwa krzyżowania	6
6.4 Zbadanie wpływu liczby wysp	8
6.5 Badanie wyników 15 funkcji z najlepszymi parametrami	9
7. Czego nauczyliśmy się dzięki projektowi?	10
8. Źródła	10

1. Cele projektu

Porównanie standardowego algorytmu ewolucyjnego z modelem wyspowym. Implementacja obu wersji i zestawienie uzyskanych przez nie wyników.

2. Interpretacja treści zadania – przyjęte założenia

2.1 Model standardowy

Wzorując się na przedstawionym na wykładzie algorytmie, zaimplementowaliśmy najpierw standardową wersję. Przyjęliśmy turniejowy model wyboru osobników do reprodukcji, prawdopodobieństwa wyboru do turnieju wyliczamy na podstawie jakości genomów, zgodnie z materiałami z wykładu. Spośród zwycięzców turnieju losujemy kolejne pary, które krzyżujemy do uzyskania P nowych osobników (gdzie P - rozmiar populacji). Każdy nowy genom poddawany jest mutacji, która jest główną siłą “napędową” ewolucji. Pokolenie $N+1$ tworzymy poprzez wybór 2 najlepszych osobników z pokolenia N (ponieważ potencjalnie jest już wśród nich rozwiązanie optymalne) oraz wybór $P-2$ najlepszych z powstałych na skutek krzyżowania.

2.2 Model wyspowy

W modelu wyspowym wykorzystujemy funkcje z wersji standardowej, używając ich do rozwoju każdej z podgrup.

Populację o “standardowym” rozmiarze rozdzielamy pomiędzy wyspy, na których rozwijają się podpopulacje. (8.1) Co pewną liczbę pokoleń oprócz operacji genetycznych (krzyżowanie, mutowanie) wykonujemy migrację. Stosujemy topologię okręgu jednokierunkowego. Migrują najlepsze osobniki, stanowiące 10% procent całej populacji (nie mniej niż jeden osobnik). Genomy osobników migrujących są usuwane z ich wyjściowej wyspy.

3. Podział zadań w zespole

- opracowanie koncepcji, ustalenie założeń implementacji – praca wspólna
- implementacja podstawowych struktur danych, funkcji składowych algorytmu standardowego oraz modelu wyspowego – *Laura Wrońska*
- implementacja głównej funkcji algorytmu standardowego – *Jakub Zieliński*
- implementacja funkcji integrujących algorytmiczną część programu z oficjalnym kodem benchmarku CEC2017; przygotowanie skryptów testujących – *Jakub Zieliński*
- opracowanie dokumentacji, wraz z opisem badań i wniosków – *Laura Wrońska*

4. Wykorzystane narzędzia i biblioteki

Projekt został zrealizowany z użyciem środowiska CLion. (8.2)

Kod programu napisaliśmy w języku C++. Korzystaliśmy ze standardowych bibliotek.

Do porównania wyników algorytmów posłużyliśmy się benchmarkiem CEC 2017. (8.3)

5. Kompilacja i uruchomienie testów

5.1 Kompilacja

```
cd build
./build.sh
```

W katalogu build znajduje się plik build.sh. Skrypt wywołuje cmake (w wersji +3.16) dla projektu, następnie buduje plik wykonywalne test_integration i test_final. Gotowe pliki zostają przeniesione do katalogu głównego oraz do katalogu tests.

5.2 Testy

Na przykład: `./run_cross.sh`

W katalogu tests znajdują się skrypty testowe oraz katalog z parametrami wewnętrznymi funkcji CEC2017. Każdy skrypt tworzy katalog wynikowy, a następnie wykonuje określoną ilość testów z zadanymi parametrami. Pliki z wynikami nie są za każdym razem tworzone od nowa, więc przy każdym następnym wywołaniu tego samego testu wyniki będą dopisywane na końcu.

6. Badania

Eksperymenty przeprowadzamy z użyciem funkcji benchmarku CEC 2017. Korzystamy z oficjalnych implementacji funkcji optymalizacyjnych, stworzyliśmy dla nich dedykowany adapter. Wszystkie z tych funkcji opisują problemy minimalizacji. W naszych badaniach dążymy więc do zmniejszenia uzyskanych wyników.

W każdym eksperymencie przyjmujemy określone parametry wejściowe, których wartości wyszczególniamy w odpowiednich tabelach. Modyfikujemy wartość jedynie badanej zmiennej. Każdą kombinację parametrów sprawdzamy 25 razy. Dla każdego sprawdzenia zapisujemy 14 wartości q (najlepszy genom po 1, 2, 3, 5, 10, 20, 30... 90, 100 procentach wykonania się algorytmu) wskazanych w dokumentacji CEC 2017.

Porównujemy je z optimum danej funkcji, czyli q^* , obliczając różnicę $q-q^*$, czyli odległość uzyskanej wartości od optimum. Właśnie te odległości są przedstawione w tabelach.

Z tych wartości wybieramy najlepszą (*best*; czyli w naszym przypadku najmniejszą) i najgorszą (*worst*; najgorsza wartość, która pojawiła się w całym przebiegu algorytmu dla danej populacji) oraz obliczamy średnią (*mean*) i medianę (*median*). Na podstawie tak wyliczonych wartości dokonujemy porównania konfiguracji algorytmu.

Badania przeprowadzamy dla osobników o genomie długości $D=10$.

Aby zmniejszyć objętość sprawozdania, postanowiliśmy początkowo ograniczyć się do badań na pierwszych 4 funkcjach benchmarku CEC. Na podstawie ich wyników wyciągamy częściowe wnioski i szukamy najlepszych parametrów. Dla konfiguracji, która pozwoliła najbardziej zbliżyć się do optimum, sprawdziliśmy wartości 15 funkcji benchmarku.

Oznaczenia parametrów:

- MG – maksymalna liczba generacji
- CP – prawdopodobieństwo krzyżowania
- MS – siła mutacji
- PS – rozmiar populacji
- E – rozmiar elity
- NoI – liczba wysp
- NoM – liczba migracji

6.1 Zbadanie wpływu zmiany rozmiaru populacji na wynik optymalizacji

Stałe parametry wejściowe:

MG	CP	MS	E	NoI	NoM
1000	0.3	1.5	2	10	20

	PS = 20							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	2.33e+06	2.19e+10	2.93e+09	2.01e+07	1.1e+10	8.64e+10	5.57e+10	6.2e+10
3	3.34e+03	7.8e+04	2.59e+04	2.2e+04	1.89e+04	1.6e+11	2.48e+10	8.64e+09
4	2.4	3.31e+03	237	9.48	945	3.86e+04	1.71e+04	1.96e+04
5	37.4	158	82.4	79.7	142	416	300	327

	PS = 50							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	4.59e+06	2.05e+10	1.43e+09	1.42e+07	1.1e+06	2.32e+10	2.6e+09	5.76e+06
3	272	4.08e+04	1.06e+04	8.14e+03	5.3	4.23e+04	6.36e+03	137
4	2.18	2.75e+03	129	8.65	0.42	3.9e+03	290	4.58
5	24.2	115	50.7	47.2	41.1	138	73.7	71.3

	PS = 100							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.44e+06	1.12e+10	9.08e+08	1.29e+07	1.17e+06	1.86e+10	1.66e+09	5.35e+06
3	79.6	3.89e+04	6.2e+03	3.66e+03	5.94	3.04e+04	4.71e+03	26.5
4	1.29	1.05e+03	74.5	7.44	0.36	2.16e+03	129	4.49
5	21.1	98.8	44.7	41.8	21	96.9	48.9	48.3

	PS = 250							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	2.92e+06	6.56e+09	4.61e+08	9.92e+06	1.13e+06	9.34e+09	8.82e+08	9.75e+06
3	31.4	2.72e+04	3.84e+03	1.38e+03	7.34	2.1e+04	2.79e+03	45.3
4	0.784	485	40.2	7.01	0.48	492	39.3	3.45
5	12.5	94.4	32.2	29.5	19.7	82.8	38.3	36.3

	PS = 500							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	3e+06	4.29e+09	3.22e+08	8.08e+06	1.22e+06	6.75e+09	4.44e+08	1.95e+07
3	23.2	1.08e+04	2.02e+03	504	5.19	1.25e+04	1.74e+03	118
4	1.08	332	30.1	6.47	0.30	352	34.1	3.86
5	9.66	67.6	23.6	20	10.6	70.7	34.4	34

	PS = 1000							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.85e+06	3.6e+09	2.24e+08	7.01e+06	6.79e+05	4.44e+09	2.89e+08	2.61e+07
3	18.4	1.09e+04	1.67e+03	275	3.33	1.2e+04	1.3e+03	194
4	2.05	205	19.6	5.12	0.40	240	20.8	4.44
5	7.56	60.6	20.6	18.1	6.30	56.70	29.40	29.30

Obserwacje: Czasami mniejsze populacje osiągnęły lepsze wyniki niż większe, np. dla PS=100 otrzymaliśmy całkiem niewielkie odległości od optimum. Ogólna tendencja wskazuje jednak na poprawę wyników wraz ze wzrostem liczebności populacji. Warto również zauważyć bardzo słabe wyniki dla modelu wyspowego, gdy rozmiar populacji wynosił 20. Wyniki były zdecydowanie gorsze niż w przypadku algorytmu standardowego. Ustawienie parametrów skutkujące małym rozmiarem pojedynczej wyspy spowodowało znaczne pogorszenie efektywności algorytmu, mimo tego że korzystamy z tej samej ilości zasobów.

Wnioski: W większości przypadków oba algorytmy uzyskały najlepsze wyniki dla najliczniejszej populacji. Wynika to z tego, że dzięki wylosowaniu większej liczby początkowych osobników dostajemy pełniejszy obraz przestrzeni i zapewniamy sobie większą liczbę punktów startowych do jej dalszej eksploracji. W przypadku używania modelu wyspowego trzeba zadbać o odpowiednią różnorodność i rozmiar populacji na każdej z wysp.

6.2 Zbadanie wpływu siły mutacji

Stałe parametry wejściowe:

MG	CP	PS	E	NoI	NoM
1000	0.3	1000	2	10	20

	MS = 0.01							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	3.96e+08	3.75e+09	1.64e+09	1.58e+09	3.23e+08	5.41e+09	2.09e+09	1.86e+09
3	386	1.03e+04	3.39e+03	3.26e+03	1.22e+03	1.36e+04	4.95e+03	4.61e+03
4	14	280	94.9	86.5	2.67	427	143	142
5	7.69	63.8	26.6	23.3	10.1	73.6	35.7	32.9

	MS = 0.05							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	4.56e+03	4.5e+09	9.94e+08	7.4e+08	2.47e+03	4.76e+09	1.23e+09	8.24e+08
3	0.02	9.15e+03	2.38e+03	2.01e+03	0.01	1.15e+04	3.28e+03	2.49e+03
4	0.34	212	62.6	56.7	1.42	336	102	80.5
5	4.98	68.9	23.4	20.9	7.96	69.7	28.9	23.9

	MS = 0.5							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	2.5e+05	3.74e+09	3.98e+08	8.19e+05	1.66e+05	4.5e+09	5.2e+08	2.94e+06
3	1.4	8.92e+03	1.25e+03	38.7	0.49	9.41e+03	1.6e+03	15.4
4	0.38	216	31.2	4.71	0.04	279	35.8	1.45
5	4.41	62.3	20.4	17.4	10	61.1	22.8	20.7

	MS = 1.5							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	7.76e+05	3.8e+09	2.31e+08	6.7e+06	1.92e+06	4.7e+09	2.7e+08	2.58e+07
3	18	9.06e+03	1.63e+03	273	3.08	9.44e+03	1.34e+03	192
4	1.24	210	19.1	5.31	0.30	246	21.4	4.27
5	8.52	61	21.1	18.8	10.4	52.7	29.5	29.2

	MS = 5							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	2.71e+07	2.21e+09	1.6e+08	5.83e+07	1.19e+07	1.96e+09	2.77e+08	2.73e+08
3	164	1.56e+04	3.13e+03	1.57e+03	88.4	1.09e+04	3.48e+03	3.32e+03
4	8.75	154	17.1	11	6.52	111	22.4	20.5
5	17.3	67.1	31.3	29.6	12.1	75.7	58.2	66.4

Obserwacje: Początkowo wraz ze wzrostem siły mutacji średnie wyniki algorytmów polepszają się, zwłaszcza dla modelu standardowego. Są dość dobre jeszcze dla MS=1.5, jednak później, dla MS=5 z powrotem się pogarszają. Warto też zauważyć, że przy mniejszej sile mutacji wartości *best* przyjmują często lepsze wartości, zaś *worst* – gorsze. Małe wartości MS (0.05) bardzo dobrze dawały sobie radę z optymalizacją funkcji 1. Dzięki małemu krokowi algorytmy były w stanie precyzyjniej niż w innych przypadkach zbliżyć się do optimum globalnego. Jednak należy też uważać, by nie ustawić zbyt małej wartości parametru.

Wnioski: Większa siła mutacji zwiększa szanse na wyjście z optimum lokalnego (w krótszym czasie). Nie powinna być jednak zbyt duża, bo wówczas algorytm nie będzie w stanie dostatecznie precyzyjnie zbliżyć się do optimum globalnego. Z drugiej strony należy ustawić MS dostatecznie duże, by było w stanie odpowiednio eksplorować przestrzeń i nie utknąć w optimum lokalnym.

6.3 Zbadanie wpływu prawdopodobieństwa krzyżowania

Stałe parametry wejściowe:

MG	MS	PS	E	NoI	NoM
1000	1.2	1000	2	10	20

	CP = 0 - BEZ KRZYŻOWANIA							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	2.48e+06	1.19e+10	1.12e+09	7.37e+06	1.1e+06	1.17e+10	1.04e+09	2.08e+07
3	22.9	3.07e+04	5.05e+03	215	4.58	2.8e+04	4.36e+03	178
4	1.06	733	69.2	3.56	0.13	723	66	3.83
5	12.2	88.2	29.8	26.3	12.8	69.9	39.1	39

	CP = 0.1							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.08e+06	4.37e+09	2.64e+08	5.71e+06	9.51e+05	5.34e+09	3.62e+08	1.92e+07
3	11.7	1.32e+04	1.33e+03	67	4.17	1.23e+04	1.52e+03	149
4	0.53	204	19.3	4.39	0.24	285	25.5	3.37
5	9.29	67	21	18.7	9.01	57.9	28.4	26.9

	CP = 0.5							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	7.51e+05	3.24e+09	3.15e+08	3.33e+06	7.77e+05	4.29e+09	3.18e+08	1.47e+07
3	11.6	1.14e+04	2.33e+03	995	2.87	9.58e+03	1.48e+03	89.8
4	0.41	275	27.4	5.41	0.29	271	25.7	3
5	8.75	67.9	23.8	22.3	8.98	58.9	27	26.8

	CP = 0.85							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.16e+06	5.37e+09	6.22e+08	3.44e+06	8.31e+05	5.57e+09	4.53e+08	1.23e+07
3	18.1	1.34e+04	4.25e+03	2.98e+03	2.81	9.29e+03	1.75e+03	77.4
4	2.6	401	48.6	5.88	0.2	303	31.1	3.51
5	18.5	74.1	36.1	35	11.3	58.4	29.8	28

	CP = 1 - KRZYŻOWANIE ZAWSZE WYSTĘPUJE							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	8.77e+05	7.01e+09	8.11e+08	2.84e+06	7.17e+05	4.68e+09	4.86e+08	1.17e+07
3	87.1	1.34e+04	5.07e+03	4.13e+03	2.59	1.07e+04	2.16e+03	101
4	2.02	474	60.1	6.13	0.19	392	39.7	3.39
5	22	84.6	38.1	35.3	14	59.5	31.2	31.8

Obserwacje: Dla przypadków skrajnych, gdzie krzyżowanie nie występuje wcale lub występuje zawsze, algorytmy uzyskują gorsze średnie wyniki. W pierwszej z tych sytuacji, oba modele osiągnęły zbliżone wartości. Najbliżej optimum algorytmy znalazły się dla CP=0.5, czyli właśnie wtedy, gdy ten współczynnik został „uśredniony”. Warto dodać, że dla mniejszych prawdopodobieństw krzyżowania, wartości *best* również były na dość dobrym poziomie.

Wnioski: Tak jak siła mutacji, prawdopodobieństwo krzyżowania nie powinno być ani za małe, ani za duże. Jeśli będzie zbyt małe, algorytm także ma szansę dotarcia do wartości optymalnej, jednak zajmie to więcej czasu, przez co wartość średnia się pogorszy. Dla zbyt dużego

przeciwnie – szybciej można uzyskać sensowne wartości, ale eksploatacja już odnalezionych optimów w celu osiągnięcia jeszcze lepszych wyników jest ograniczona. Trzeba też pamiętać, że przy CP=0 nie zaobserwujemy znacznej różnicy między modelem standardowym a wyspowym, skoro po migracjach nie będzie następować wymiana materiału genetycznego między podpopulacjami.

6.4 Zbadanie wpływu liczby wysp

Stałe parametry wejściowe:

MG	CP	MS	PS	E	NoM
1000	0.5	1.2	1000	2	20

	NoI = 2							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.07e+06	4.26e+09	3.49e+08	3.38e+06	1.1e+06	3.7e+09	3.14e+08	1.89e+07
3	6.83	1.03e+04	2.45e+03	976	1.22	9.02e+03	1.51e+03	155
4	0.78	242	27.2	5.14	0.40	216	25.5	5.25
5	7.66	62.7	24	22	10	52.2	30.1	30.3

	NoI = 5							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.3e+06	4.09e+09	3.49e+08	3.74e+06	9.36e+05	3.81e+09	3.18e+08	1.72e+07
3	5.9	1.11e+04	2.29e+03	922	2.34	7.56e+03	1.41e+03	117
4	0.53	244	26.2	5.2	0.23	295	25.8	3.94
5	6.14	62.7	24.9	22.8	8.24	49.7	28.3	28

	NoI = 10							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	1.2e+06	5e+09	3.65e+08	3.53e+06	7.23e+05	4.13e+09	3.13e+08	1.49e+07
3	7.11	1.02e+04	2.13e+03	845	2.35	9.42e+03	1.53e+03	89.9
4	0.75	298	28.2	5.37	0.22	287	26.8	3.57
5	9.76	64.8	25.2	22.5	11	56.4	26.6	26.3

	NoI = 20							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	4.41e+05	3.72e+09	3.51e+08	3.51e+06	5.5e+05	4.85e+09	3.6e+08	1.08e+07
3	10.6	9.98e+03	2.33e+03	1.07e+03	1.99	1.05e+04	1.5e+03	53.7
4	0.73	348	29.9	5.39	0.11	365	27.4	2.21
5	7.86	62.7	24	21.8	13	52.1	26.7	25.9

	NoI = 50							
	Model standardowy				Model wyspowy			
Nr F.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	6.71e+05	4.46e+09	3.63e+08	3.49e+06	8.06e+05	5.93e+09	5.65e+08	4.68e+06
3	8.01	1.38e+04	2.58e+03	1.17e+03	2.57	1.18e+04	1.74e+03	19.4
4	0.95	220	28	5.32	0.19	337	32.7	1.4
5	11	58.3	23.7	21.5	8.17	55.9	25.5	25.2

Obserwacje: Zmiana liczby wysp nie ma wpływu na działanie modelu standardowego, dlatego we wszystkich przypadkach uzyskał on podobne średnie wyniki. Na podstawie wartości *best* i *worst* można jednak zaobserwować działanie czynników losowych.

Przy mniejszej liczbie wysp (i bardziej licznych podpopulacjach) algorytm wyspowy osiąga gorsze wyniki; tak samo dla zbyt dużego rozdrobnienia na wyspy (przy NoI=50), kiedy podpopulacje stają się małymi grupami. Dla większości funkcji najlepsze wartości *mean* i *median* pojawiają się przy 20 wyspach.

Wnioski: W modelu z 20 wyspami po 50 osobników podpopulacje są dość duże, by każda z nich mogła rozwijać się samodzielnie w sensowny sposób, biorąc pod uwagę m. in. reprodukcję turniejową. Ponadto zostaje zachowana topologia migracji po okręgu (inaczej niż przy tylko 2 wyspach), a migracje między danymi grupami genomów występują na tyle rzadko, by zachować różnorodność między wyspami.

6.5 Badanie wyników 15 funkcji z najlepszymi parametrami

Na podstawie poprzednich badań ustaliliśmy parametry, które dają wyniki najbliższe optimum:

MG	CP	MS	PS	E	NoM	NoI
1000	0.5	1.2	1000	2	20	20

		Model standardowy				Model wyspowy			
Nr F.	W. opt.	Best	Worst	Mean	Median	Best	Worst	Mean	Median
1	100	7.64e+05	4.48e+09	3.53e+08	3.71e+06	6.5e+05	5.74e+09	4.46e+08	1.06e+07
3	300	6.31	1.1e+04	2.27e+03	928	3.91	9.93e+03	1.23e+03	54.3
4	400	0.728	239	26.1	5.43	0.17	309	27.1	2.43
5	500	7.57	66	24.7	22.2	12.4	54.3	26.6	25.4
6	600	1.09	35.6	8.65	5.46	0.66	35.5	8.8	5.12
7	700	25.9	69.4	37.8	35	20.3	90.4	62.2	64.6
8	800	5.55	46.3	12.4	9.1	5.29	40.7	16.8	15.8
9	900	0.48	181	13	1.22	0.37	267	28.7	3.33
10	1000	282	2.04e+03	1e+03	987	260	1.33e+03	702	704
11	1100	6.94	465	56.4	41.8	5.77	392	44.9	24.8
12	1200	8.85e+04	2.44e+07	1.08e+06	7.62e+05	1.31e+04	5.95e+06	1.23e+06	1.26e+06
13	1300	4.9e+03	3.87e+04	9.66e+03	9.31e+03	729	2.41e+05	8.58e+04	9.74e+04
14	1400	121	2.83e+03	942	905	43.7	9.06e+03	1.86e+03	1.52e+03
15	1500	594	1.61e+04	7.34e+03	7.57e+03	42.3	3.38e+04	6.42e+03	4.75e+03
16	1600	8.74	387	193	188	3.89	337	134	145

Obserwacje: W niemal wszystkich przypadkach algorytm wyspowy poradził sobie lepiej niż standardowy, niekiedy uzyskując wartości *best* lepsze nawet o kilka rzędów wielkości. Często jednak różnice między *best* i *worst* dla wersji wyspowej są większe, co skutkuje gorszą średnią.

Mimo to, tylko w jednym przypadku algorytm wyspowy miał większą odległość od optimum niż algorytm standardowy. Mediany dla obu modeli utrzymują się na podobnym poziomie i są dla jednego z nich lepsze lub gorsze w zależności od konkretnej funkcji.

Warto też zauważyć, że chociaż algorytmy nie osiągnęły optimum, bardzo się do niego zbliżyły – odległości od tego punktu liczone przy pomocy metryki odpowiedniej dla przestrzeni są bardzo małe, co daje nam dobre przybliżenie położenie optimum globalnego.

Wnioski: Uwzględniając losowość algorytmów, można powiedzieć, iż wersja wyspowa lepiej radzi sobie ze znajdowaniem najlepszej wartości (*best*). Wyniki są bardziej zróżnicowane, co wskazuje na lepszą eksplorację przestrzeni. Przy dłuższym działaniu programu ma więc większą szansę na dotarcie do optimum.

Wartości średnie oraz mediany są zwykle lepsze dla modelu standardowego, co oznacza, że ta wersja algorytmu jest bardziej stabilna i częściej uzyskuje sensowne, zbliżone do siebie wyniki.

7. Czego nauczyliśmy się dzięki projektowi?

Na potrzeby projektu zapoznaliśmy się ze szczegółową implementacją algorytmów ewolucyjnych. Mieliśmy okazję dogłębnie przeanalizować zarówno standardową wersję algorytmu, jak i jego mniej znaną, ciekawą odmianę, czyli model wyspowy. Podczas wykonywania badań musieliśmy zastanowić się, w jaki sposób ustalić parametry, by osiągnąć jak najlepszy wynik. Korzystaliśmy wówczas z wiedzy wyniesionej z wykładów, dzięki czemu ją ugruntowaliśmy oraz pogłęбилиśmy. Ponieważ następnie obserwowaliśmy, jakie wyniki uzyskują obie wersje, mogliśmy przekonać się, które parametry są najbardziej istotne, dla jakich wartości algorytmy radzą sobie najlepiej oraz jakie mają właściwości.

8. Źródła

8.1 Opis algorytmu wyspowego:

http://algorytmy-genetyczne.eprace.edu.pl/661,Algorytmy_Wielopopulacyjne_-_Model_Wyspowy_IslandModel.html

8.2 Clion: <https://www.jetbrains.com/clion/>

8.3 Benchmark CEC 2017: <https://github.com/P-N-Suganthan/CEC2017/>