

Sprawozdanie 2

Testowanie opracowanej metody heurystycznej

Mateusz Babiaczyk, Bartosz Nawrotek

2018-06-01

1 Zmiany w algorytmie

Po zaimplementowaniu algorytmu i zauważeniu jego słabych osiągnięć, doszliśmy do wniosku by wprowadzić zmiany w naszym algorytmie.

1.1 Mutacje

1.2 Krzyżowanie

Krzyżowanie zaczyna się w dokładnie taki sam sposób jak było w naszym pierwotnym algorytmie, a mianowicie od pewnego wylosowanego przedziału przepisuje oligonukleotydy do nowo tworzonego osobnika (kopiuje wycinek i wkleja go do nowego osobnika) z wybranego osobnika z populacji rodzicielskiej. Następnie uzupełniany jest koniec osobnika wartościami z innego osobnika z populacji rodzicielskiej, uważając oczywiście by dany oligonukleotyd nie został powtórzony. W ten sam sposób zostaje uzupełniony początek osobnika z nowej populacji.

Wszystkie oligonukleotydy które nie zostały dodane (na skutek dodawania ich z innego osobnika który za punktem cięcia mógł mieć oligonukleotyd taki sam jak drugi osobnik użyty do krzyżowania pomiędzy punktami cięcia) zostają dodane, każdy osobno, w miejsce w którym funkcja celu będzie najniższa. Dzięki takiemu nakierowaniu, nadal mieliśmy pewną dużą losowość przez krzyżowanie między losowymi punktami cięcia (która jest ważna w algorytmie genetycznym), a jednocześnie algorytm szybciej zbiegał do wartości optymalnych, tym samym dając lepsze rezultaty w krótszym czasie.

1.3 Kodowanie

2 Testy

2.1 Wyniki algorytmu dla błędów negatywnych, losowych

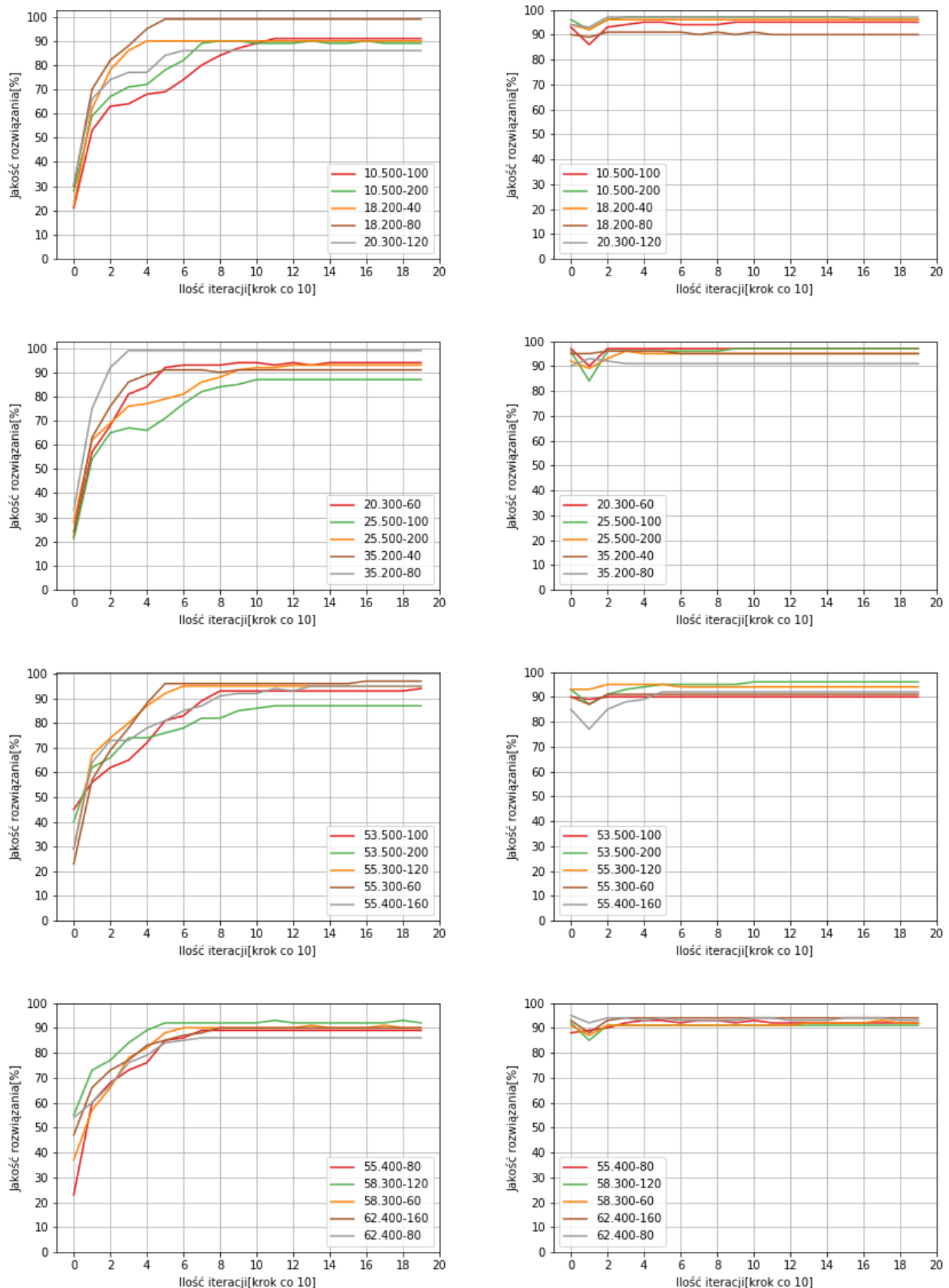


Figure 1: Porównanie algorytmu dla błędów negatywnych, losowych bez oraz z wykorzystanym osobnikiem wygenerowanym przez algorytm zachłanny

Jak widać, dla błędów negatywnych losowych algorytm otrzymuje rozwiązanie o średniej dokładności 90

2.2 Wyniki algorytmu dla błędów negatywnych na końcach sekwencji

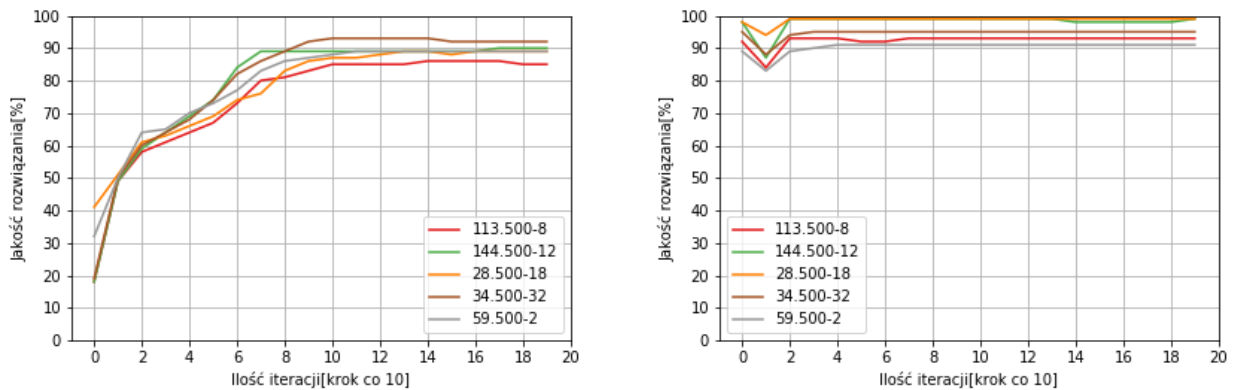


Figure 2: Porównanie algorytmu dla błędów negatywnych na końcach sekwencji bez oraz z wykorzystanym osobnikiem wygenerowanym przez algorytm zachłanny

W przypadku błędów negatywnych na końcach sekwencji wyniki są zbliżone do błędów negatywnych losowych i zachowują taką samą własność.

2.3 Wyniki algorytmu dla błędów pozytywnych, losowych

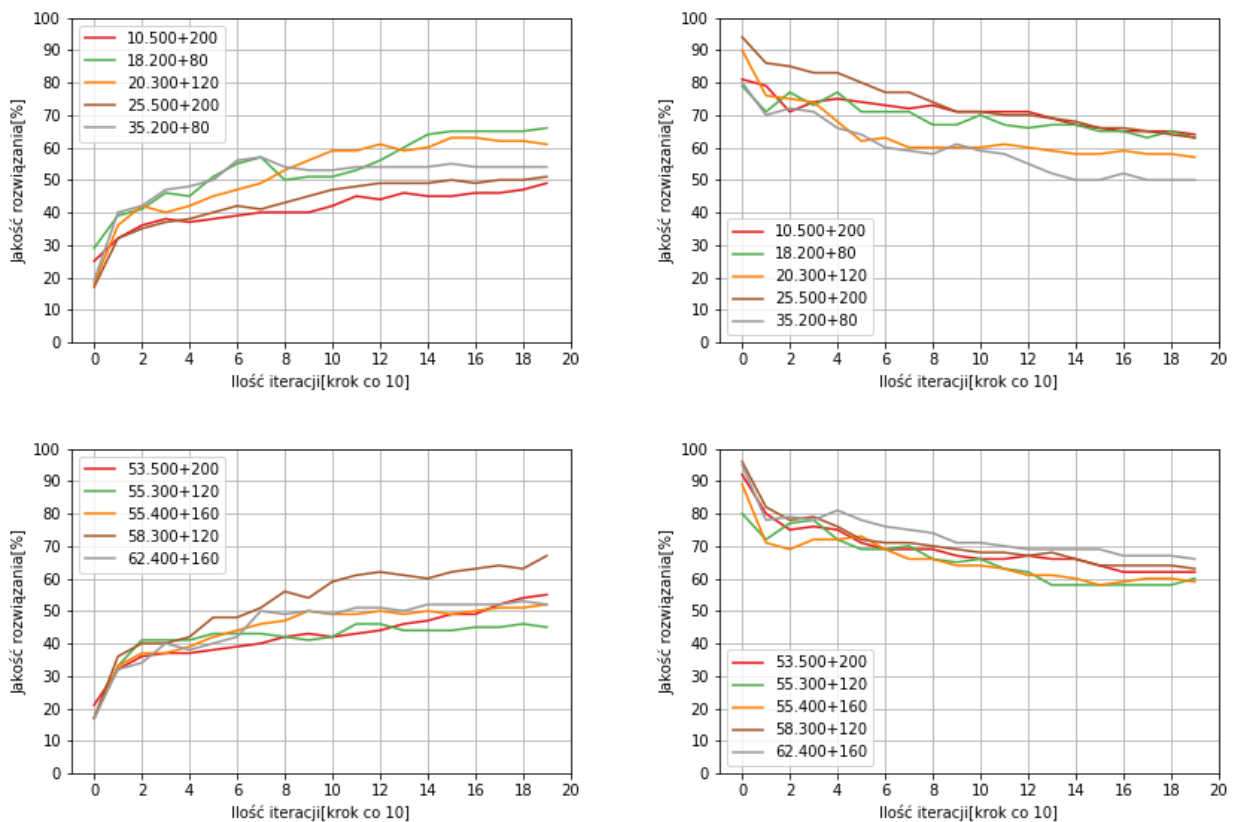


Figure 3: Porównanie algorytmu dla błędów pozytywnych, losowych bez oraz z wykorzystanym osobnikiem wygenerowanym przez algorytm zachłanny

Dla błędów pozytywnych losowych nasz algorytm nie działa już tak dobrze. Jak widać algorytm bez początku zachłannego uzyskuje średnie wyniki w okolicach 60

2.4 Wyniki algorytmu dla błędów pozytywnych, na końcach sekwencji

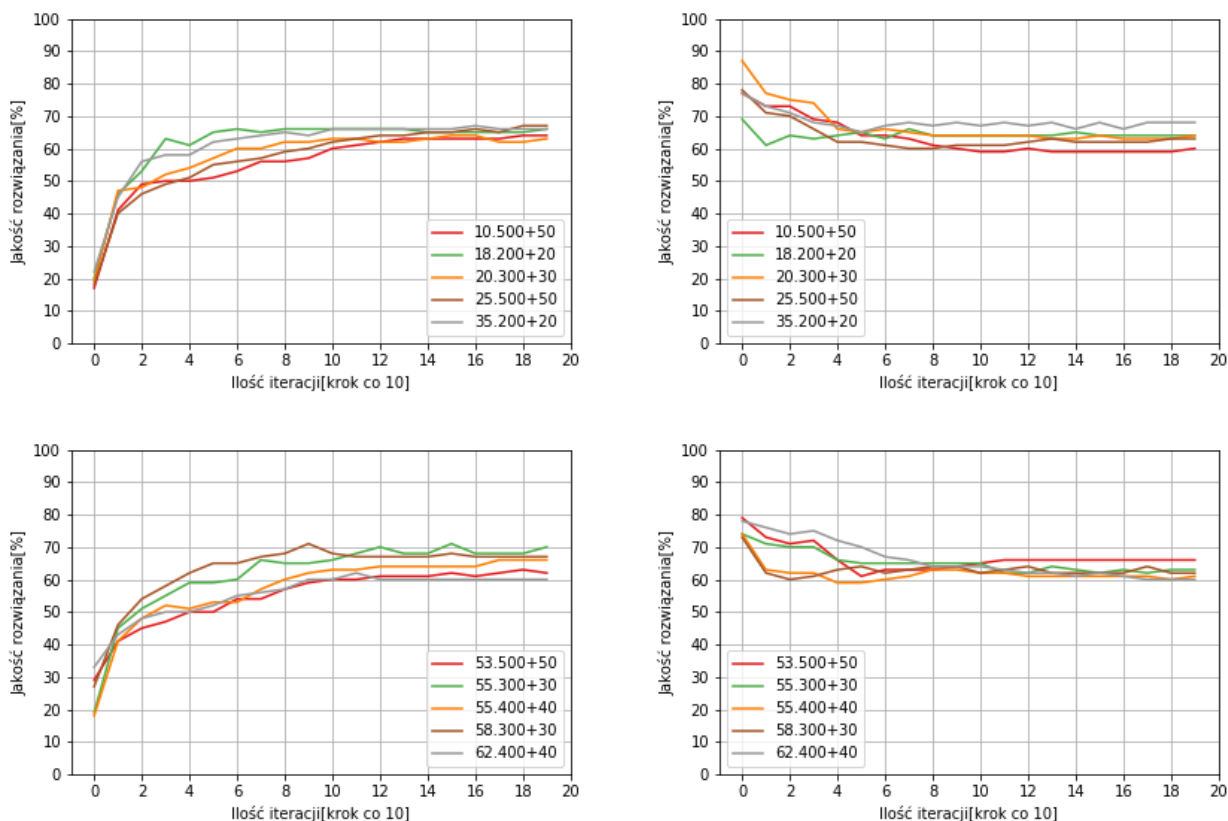


Figure 4: Porównanie algorytmu dla błędów pozytywnych na końcach sekwencji bez oraz z wykorzystanym osobnikiem wygenerowanym przez algorytm zachłanny

Ostatnią kategorią są błędy pozytywne z przekłamaniami na końcach oligonukleotydów. Jak widać radzi on sobie odrobinę lepiej niż dla błędów pozytywnych losowych, dodatkowo algorytm z rozpoczęciem zachłannym nie maleje aż tak bardzo (algorytm zachłanny ma większe problemy z optymalnym ustawieniem i jego wykonanie nie powoduje od razu ustawienia pierwszej iteracji na wysokości 90-95

3 Wnioski