

Spis treści

1 Wprowadzenie	2
2 Środowisko pracy	3
3 Opis metodyki	4
3.1 Filtracja sygnału	4
3.2 Detekcja oddechów	5
3.3 Segmentacja sygnału	6
4 Aproksymacja sygnału	7
4.1 Aproksymacja z wykorzystaniem algorytmu genetycznego (GA)	7
4.2 Funkcja celu (fitness)	8
5 Ocena jakości aproksymacji	9

Rozdział 1

Wprowadzenie

W naszym projekcie wykonaliśmy aproksymator funkcji dla danych oddechowych. Analiza sygnałów oddechowych odgrywa istotną rolę w wielu dziedzinach życia. Coraz powszechniejsze urządzenia pozwalają na śledzenie parametrów oddechowych, co ma kluczowe znaczenie w diagnostyce, opiece nad pacjentami oraz w analizie wysiłku fizycznego.

Sygnały oddechowe bywają trudne w interpretacji, ponieważ charakteryzują się
-zmiennością
-obecnością szumów
-nieliniowością przebiegu

Dlatego tego typu dane wymagają odpowiedniego wstępnego przetwarzania, czyli filtracji, a następnie efektywnej aproksymacji, która umożliwia dalszą analizę lub rekonstrukcję ich kształtu.

Celem niniejszego projektu było opracowanie i przetestowanie metody aproksymacji sygnałów oddechowych przy użyciu algorytmu genetycznego (GA), który optymalizuje współczynniki w szeregu trygonometrycznym. Podejście to umożliwia dopasowanie funkcji aproksymującej nawet w przypadku sygnałów nieregularnych lub zakłóconych, gdzie klasyczne metody (np. FFT) mogą być mniej skuteczne.

Rozdział 2

Środowisko pracy

Projekt został zrealizowany w języku **Python 3**. Wykorzystano następujące biblioteki:

- `numpy` – obsługa obliczeń numerycznych, wektorowych i macierzowych; podstawowe operacje na sygnałach,
- `pandas` – wczytywanie i analiza danych z plików CSV (m.in. sygnały akcelerometru i etykiety),
- `matplotlib.pyplot` – generowanie wykresów porównawczych i diagnostycznych (filtracja, aproksymacja, fitness GA),
- `scipy.signal` – filtracja pasmowoprzepustowa (funkcje `butter` i `filtfilt`) oraz detekcja lokalnych ekstremów (funkcja `find_peaks`),
- `random` i `numpy.random` – losowa generacja chromosomów, mutacji i selekcji w algorytmie genetycznym.

Projekt podzielony został na następujące pliki:

- `przygotowywanie_danych.py` – wczytywanie danych, filtracja sygnału, dzielenie na fragmenty, liczenie częstości oddechów,
- `trygonometryczna.py` – klasyczna aproksymacja trygonometryczna (FFT),
- `ga_approx.py` – implementacja algorytmu genetycznego do optymalizacji współczynników aproksymacji,
- `OstatniOddech.py` – główny plik projektu, integrujący wszystkie etapy przetwarzania i wizualizacji wyników.

Dzięki temu projekt umożliwia pełne porównanie klasycznej metody FFT z podejściem opartym na algorytmie genetycznym oraz wizualizację wyników w postaci wykresów.

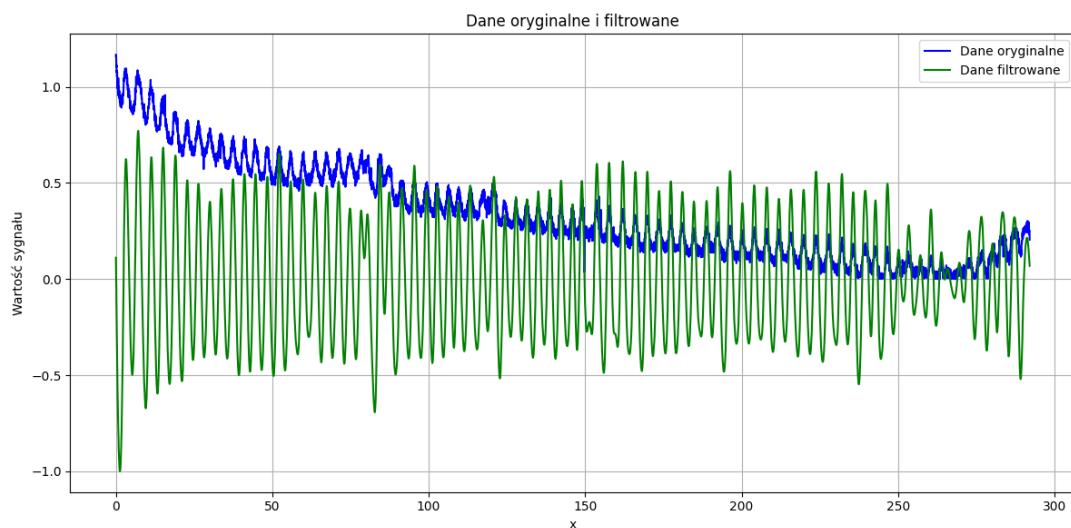
Rozdział 3

Opis metodyki

3.1 Filtracja sygnału

Pierwszym krokiem przetwarzania była filtracja sygnału oddechowego z wykorzystaniem filtra pasmowoprzepustowego typu Butterwortha. Filtracja została wykonana przy użyciu funkcji `butter` i `filtfilt` z biblioteki `scipy.signal`. Dzięki temu możliwe było usunięcie wolnozmiennych trendów i szumów wysokoczęstotliwościowych.

Ważnym elementem była również normalizacja danych wejściowych do zakresu [-1,1]. Z każdej stony 3 procent danych zostało odcięte by uniknąć nieprzewidzianych błędów przy detekcji czytów. Ma to znikomy wpływ na całościowe wyniki, ale poprawia stabilność działania. Często ucięcie małej części danych z końców sygnału jest uważane za dobrą praktykę.



Rysunek 3.1: Porównanie sygnału oryginalnego z przefiltrowanym.

Na powyższym rysunku widać wyraźną różnicę między danymi przed filtrowaniem, a danymi po filtrowniu. Widać wyraźnie, że filtracja pasmowoprzepustowa usuwa komponenty wolnozmienne (dryft i składową DC), a także tłumi zakłócenia spoza przedziału [0.1 Hz – 0.5 Hz]. Zkres ten dpojada typowym zakresom dla rytmu oddechowego człowieka w spoczynku. Normalizacja filtrowanego sygnału do przedziału [-1,1], co ułatwia jego dalszą

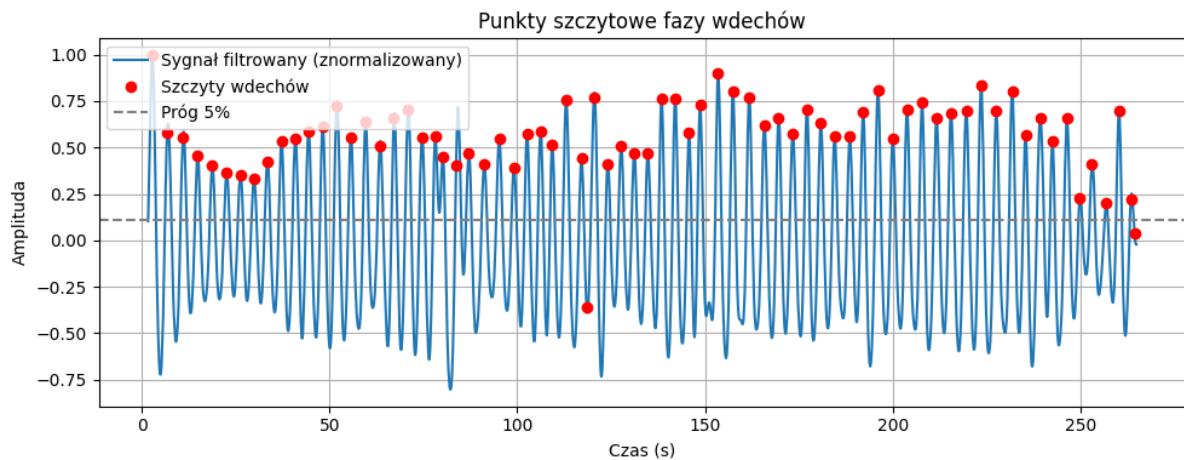
analizę i porównanie między fragmentami. Jest to pomocne dla algorytmu genetycznego, ponieważ ułatwia dobór zakresów mutacji i krzyżowania.

3.2 Detekcja oddechów

W ramach eksperymentu pokazanego na rysunku 3.2 wykorzystano dane z pliku `acc_normal_labelled`, w którym oznaczenia wartości 1.0 wskazują momenty wdechów. Program filtryuje dane w taki sam sposób jak w przypadku danych nieetykietowanych, a następnie wykrywa pozycje oznaczone jako oddech (etykieta 1).

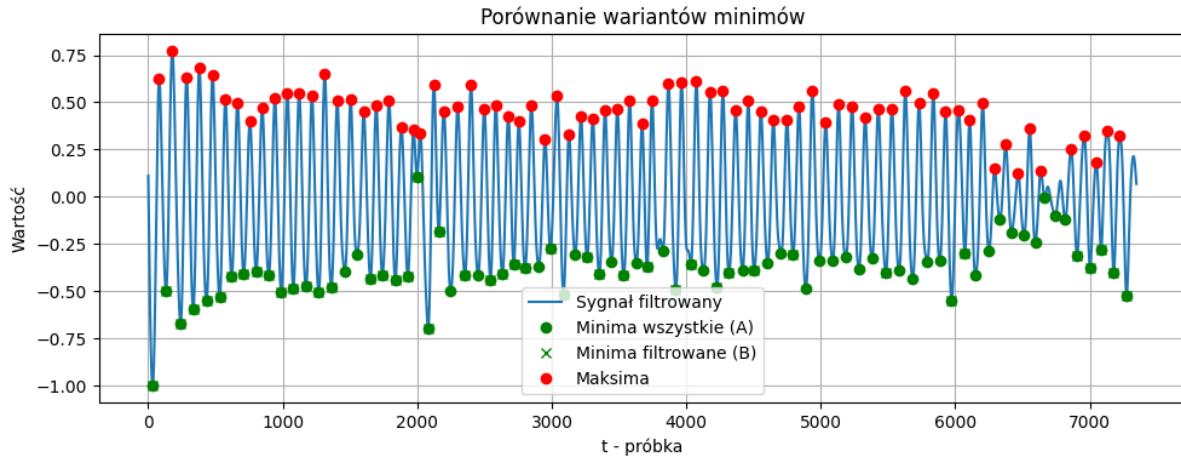
Dla każdego wykrytego oddechu wyznaczany jest szczyt, czyli ostatnia i najwyższa wartość sygnału w tej fazie. Aby uniknąć wpływu błędnych oznaczeń lub zakłóceń, odrzucone jest 5% najniższych wartości spośród wszystkich szczytów. Następnie spośród pozostałych wybierany jest najniższy szczyt, którego wartość służy jako próg referencyjny.

Wartość ta jest później stosowana jako globalny próg w analizie pozostałych danych – służy do odróżniania rzeczywistych oddechów od przypadkowych fluktuacji sygnału.



Rysunek 3.2: Przykład detekcji punktów szczytowych fazy wdechów.

3.3 Segmentacja sygnału



Rysunek 3.3: Wizualizacja detekcji minimów i maksimów w pełnym przebiegu sygnału. Zielone punkty to minima, czerwone – maksima.

Następnie korzystając z wyznaczonego wcześniej progu, program wyznacza istotne minima i maksima. Na początku znajduje minima, a następnie maksima, które filtrowane są w taki sposób, aby maksimum mogło znajdować się tylko pomiędzy minimami. Prowadzi to do sytuacji, w której w sygnale może pojawić się parę minimów koło siebie. Efekt jest pozytywny, ponieważ lepiej odzworowuje to przestoje oddechowe (dla próby wyrównania liczb minimów i maksimów program dawał gorsze wyniki).

Tak przetworzony zbiór ekstremów posłużył do wyznaczenia punktów podziału, co umożliwiło wydzielenie spójnych i porównywalnych fragmentów sygnału. Każdy fragment zawiera ustaloną liczbę oddechów, co stanowi podstawę do dalszej aproksymacji z użyciem algorytmu genetycznego. Jako, że znamy liczbę oddechów na fragment, to na jej podstawie ustalamy liczbę harmoniczną M , równą liczbie oddechów na fragment (u nas 5)

Rozdział 4

Aproksymacja sygnału

W celu rekonstrukcji przebiegu oddechu zastosowane jest podejście oparte na szeregu trygonometrycznym. Algorytm genetyczny wykorzystuje współczynniki tego szeregu do optymalizacji :

$$y[n] \approx a_0 + \sum_{k=1}^M \left(a_k \cos \left(\frac{2\pi kn}{N} \right) + b_k \sin \left(\frac{2\pi kn}{N} \right) \right),$$

gdzie N to liczba próbek w analizowanym fragmencie, a M – rząd modelu.

4.1 Aproksymacja z wykorzystaniem algorytmu genetycznego (GA)

Każdy fragment sygnału, zawierający ustaloną liczbę oddechów, traktowany jest jako osobny przypadek aproksymacji. Algorytm genetyczny przeszukuje przestrzeń współczynników a_k i b_k w celu minimalizacji błędu RMSE pomiędzy oryginalnym sygnałem a przybliżeniem na tym fragmencie. Algorytm genetyczny przeszukuje przestrzeń współczynników a_k i b_k w celu uzyskania najlepszego dopasowania. W każdym pokoleniu:

- **Selekcja turniejowa** – z populacji losujemy pary osobników, a następnie do kolejnego etapu przechodzi ten, którego dopasowanie do danych (fitness) jest lepsze. Ten sposób selekcji zachowuje równowagę między eksploracją dobrych rozwiązań a różnorodnością populacji.
- **Krzyżowanie liniowe** – tworzymy potomków jako kombinację liniową rodziców. Każdy gen nowego osobnika powstaje przez zmieszanie odpowiadających sobie współczynników a_k i b_k od dwóch rodziców, z wagą losowo wybieraną z przedziału $[0, 1]$. Dzięki temu możliwe jest dziedziczenie cech obu rodziców w płynny sposób.
- **Mutacja gaussowska** – z ustalonym prawdopodobieństwem każdy gen potomka może zostać zaburzony wartością pochodzącą z rozkładu normalnego. Wariancja mutacji jest skalowana względem amplitudy sygnału, co pozwala uniknąć zarówno zbyt gwałtownych zmian, jak i stagnacji w późniejszych pokoleniach.

Każdy osobnik w populacji reprezentuje zestaw współczynników szeregu trygonometrycznych. Po zakończeniu procesu ewolucyjnego najlepszy osobnik (czyli taki, który minimalizował błąd RMSE względem całego fragmentu sygnału) był używany do wygenerowania przebiegu aproksymacyjnego dla danego fragmentu.

4.2 Funkcja celu (fitness)

W celu oceny jakości dopasowania funkcji trygonometrycznej do przebiegu oddechowego zastosowano funkcję celu **fitness_extrema_sinusoidalny**, uwzględniającą dwa główne kryteria:

1. **Zgodność w punktach charakterystycznych (ekstremach):** Pierwszy składnik to błąd średniokwadratowy (RMSE) liczone w punktach wykrytych minimów i maksimów:

$$\text{err}_{\text{extrema}} = \sqrt{\frac{1}{n} \sum_{i \in \text{ekstrema}} (y[i] - \hat{y}[i])^2},$$

gdzie n to liczba ekstremów, $y[i]$ – wartość oryginalnego sygnału, a $\hat{y}[i]$ – wartość aproksymacji w tej samej próbce.

2. **Gładkość poza ekstremami:** Drugi składnik to miara „szorstkości” (roughness), obliczana jako średnia wartość bezwzględna drugiej różnicy dyskretnej aproksymacji, przy czym okolice ekstremów i próbki jest pomijanych:

$$\text{roughness} = \frac{1}{N'} \sum_{j \notin \text{mask}} |\hat{y}[j+1] - 2\hat{y}[j] + \hat{y}[j-1]|,$$

gdzie N' to liczba próbek poza maską.

Oba składniki łączone są w końcowej funkcji fitness w sposób odwrotny (im mniejszy błąd, tym wyższy wynik):

$$\text{fitness} = \frac{1}{1 + 4 \text{err}_{\text{extrema}} + 3 \text{roughness}}.$$

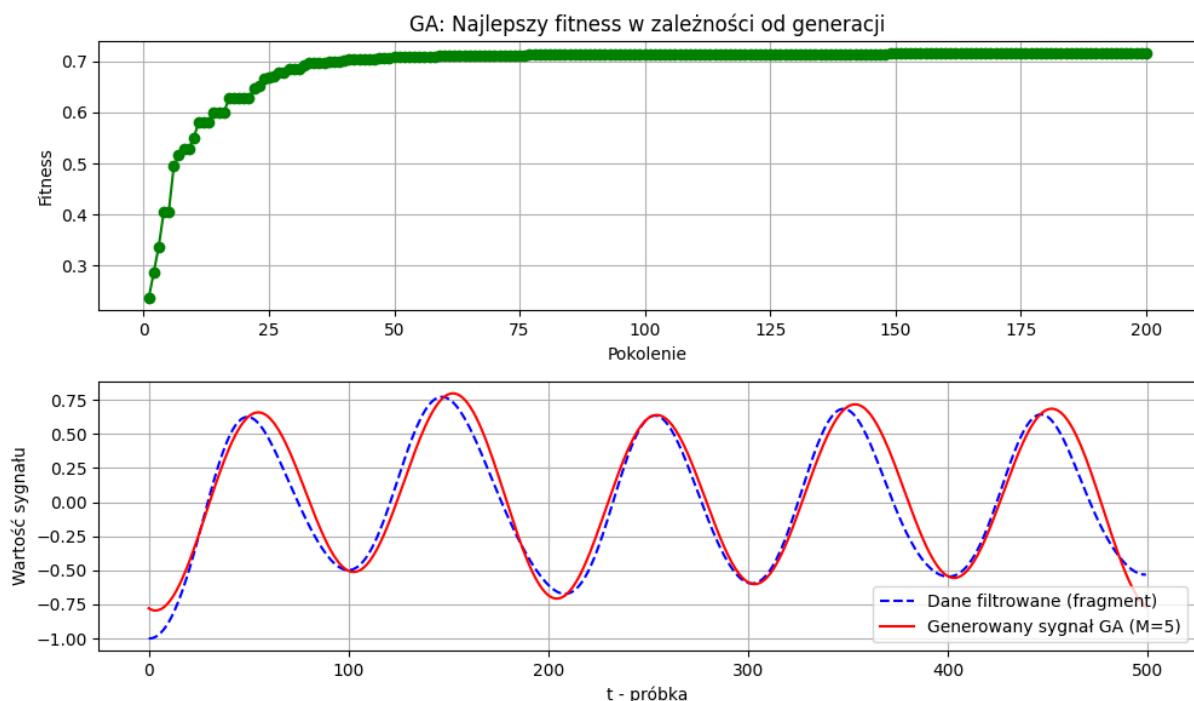
Współczynniki wagowe 4 i 3 dobrano eksperymentalnie, nadając priorytet dopasowaniu w ekstremach, przy jednoczesnym zachowaniu płynności przebiegu.

Rozdział 5

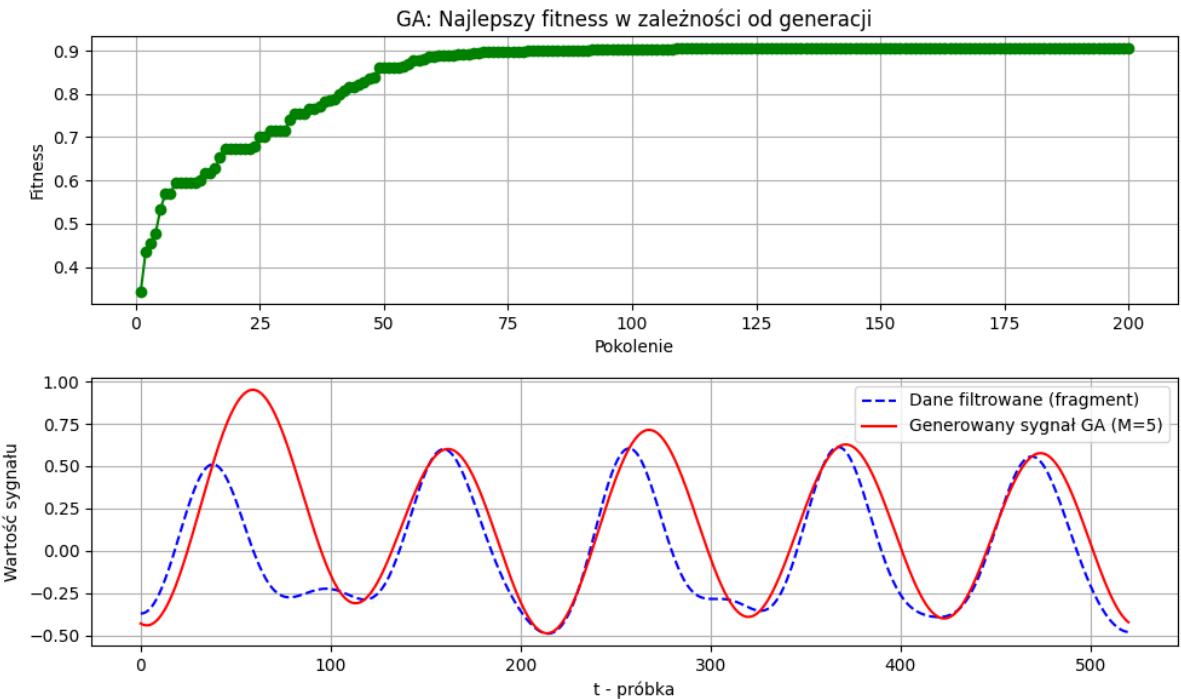
Ocena jakości aproksymacji

Wykresy - Przykład aproksymacji fragmentu sygnału metodą GA - składają się z górnego wykresu, który pokazuje jak zmienia się najlepszy fitness w czasie (czyli z każdą generacją) oraz dolnego z porównaniem oryginalnego fragmentu sygnału oddechowego z jego aproksymacją GA dla M=5.

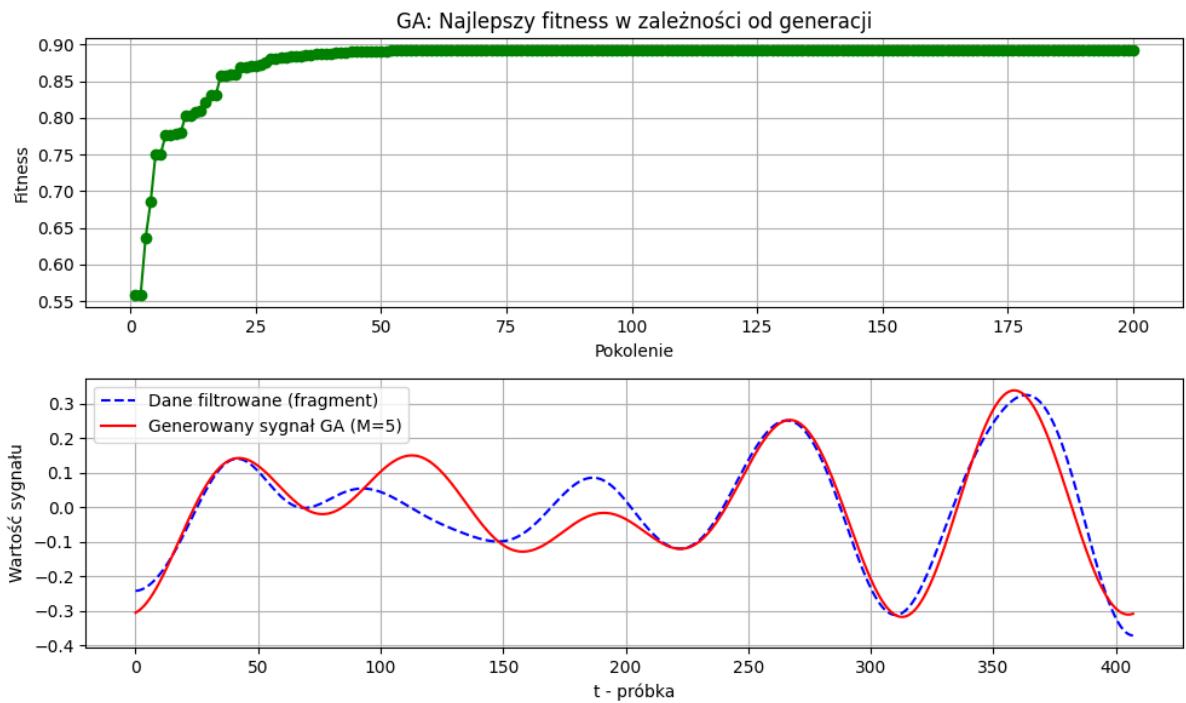
Pokazują skuteczność działania algorytmu genetycznego w różnych warunkach. Algorytm konsekwentnie zbiega do stabilnego rozwiązania, a przyjęty model (rzad M = 5) pozwala na realistyczne odwzorowanie przebiegu sygnału oddechowego. W niektórych fragmentach (np. fragment 7) występują lokalne niedopasowania, co może wynikać z ograniczeń modelu przy złożonej strukturze sygnału. Niemniej jednak, wyniki uzyskane dla wszystkich przypadków potwierdzają zdolność algorytmu do adaptacji i dopasowania się do danych rzeczywistych w warunkach szumowych.



Rysunek 5.1: Przykład aproksymacji fragmentu sygnału metodą GA.



Rysunek 5.2: Przykład aproksymacji fragmentu sygnału metodą GA.



Rysunek 5.3: Przykład aproksymacji fragmentu sygnału metodą GA.

Widać, że algorytm genetyczny może skutecznie aproksymować przebiegi sygnału oddechowego na bazie modelu trygonometrycznego. Mimo ograniczonego rzędu modelu, uzyskane dopasowania były stabilne, zbieżne i odporne na zakłócenia. Sprawdza się szczególnie w warunkach, gdzie dane są zaszumione, a klasyczne metody mogą zawodzić.