

Algorytm szeregowania produkcji z przezbiorzeniami z algorytmem genetycznym

Kacper Bednarski #259783

January 21, 2025

Numer Grupy: 1

Prowadzący: dr inż. Jerzy Ramert-Greblicki

1 Opis projektu

Projekt dotyczy optymalizacji harmonogramu produkcji w środowisku przemysłowym przy użyciu zaawansowanych metod takich jak algorytmy genetyczne. System umożliwi tworzenie harmonogramów, które uwzględniają takie czynniki jak czas konfiguracji maszyn, czas produkcji, dostępność maszyn oraz terminy realizacji zleceń.

1.1 Cel Projektu

Głównym celem projektu jest opracowanie narzędzia wspomagającego efektywne planowanie produkcji, które minimalizuje czas produkcji oraz czasy przestoju wynikające z rekonfiguracji maszyn.

Funkcjonalność systemu:

1. Analiza danych wejściowych:

- System wczytuje dane dotyczące zadań produkcyjnych, wymagań czasowych, charakterystyki maszyn i produktów oraz zapotrzebowania.
- Dane wejściowe są przetwarzane w formacie CSV, co pozwala na łatwą integrację z istniejącymi systemami zarządzania.

2. Generowanie harmonogramu:

- Użycie algorytmu genetycznego do wygenerowania optymalnych harmonogramów pracy maszyn.
- Uwzględnienie ograniczeń związanych z dostępnością maszyn, czasem rekonfiguracji i czasem produkcji.

3. Wizualizacja wyników:

- Prezentacja harmonogramu w formie interaktywnych wykresów Gantta, umożliwiających przejrzystą wizualizację działań na poszczególnych maszynach.
- Dodatkowa analiza czasów najlepszych harmonogramów w formie wykresu trendu.

4. Eksport danych wyjściowych:

- Generowanie szczegółowych raportów z harmonogramów w formacie CSV, które mogą być dalej przetwarzane lub udostępniane innym działom.

1.2 Wykorzystane technologie

1. Python: Język programowania wykorzystywany do implementacji logiki algorytmicznej oraz przetwarzania danych.
2. Biblioteki analityczne i wizualizacyjne:
 - *pandas* i *numpy* do przetwarzania danych
 - *matplotlib* i *plotly* do wizualizacji wyników.
3. Algorytm genetyczny: Wykorzystany do optymalizacji harmonogramów produkcyjnych w sposób iteracyjny.

2 Opis kodu

1. Funkcje do obsługi plików CSV:
 - `read_csv(file_path)`: Wczytuje pliki CSV do listy słowników, gdzie każdy wiersz jest reprezentowany jako słownik.
 - `parse_inputs()`: Parsuje dane wejściowe z plików CSV:
 - Lista produktów (`ListaANC.csv`).
 - Przypisania produktów do maszyn (`PrasyPrzypisanie.csv`).
 - Czasy konfiguracji maszyn (`setupx.csv`).
 - Prędkości produkcji (`ListaANCPelna.csv`).
 - Zapotrzebowania i terminy dostaw (`batch-dt.csv`).
2. Funkcje wyjściowe:
 - `generate_csv_outputs`: Zapisuje wyniki harmonogramowania do plików CSV:
 - Plik `summary_output.csv` zawiera podsumowanie czasu produkcji i konfiguracji dla każdej maszyny.
 - Plik `detailed_schedule.csv` zawiera szczegółowy harmonogram zadań dla każdej maszyny, w tym czas rozpoczęcia i zakończenia produkcji.
 - `generate_gantt_chart_matplotlib`: Tworzy wykres Gantta przedstawiający harmonogram produkcji na maszynach:
 - Wykorzystuje bibliotekę `Plotly` do wizualizacji.
 - Prezentuje zadania według maszyn, uwzględniając czasy rozpoczęcia i zakończenia produkcji oraz kolorowanie według kategorii produktów.
3. Algorytm genetyczny:
 - `genetic_algorithm`: Implementuje algorytm genetyczny z elitarną selekcją, krzyżowaniem i mutacją.
 - Populacja początkowa: Każdy harmonogram w populacji jest generowany losowo przy użyciu funkcji `generate_schedule`.
 - Krzyżowanie: Funkcja `crossover` łączy dwa harmonogramy, tworząc nowe potomstwo.
 - Mutacja: Funkcja `mutate` wprowadza losowe zmiany w harmonogramach, aby zwiększyć różnorodność populacji.
 - Selekcja: Najlepsze harmonogramy są wybierane na podstawie ich jakości (`fitness`), a następnie krzyżowane i mutowane, aby utworzyć nową populację.
4. Generowanie harmonogramu:
 - `generate_schedule`: Tworzy losowy harmonogram produkcji, przypisując produkty do maszyn z uwzględnieniem dostępnych stacji i ograniczeń czasowych.
5. Ocena harmonogramu:
 - `evaluate_schedule`: Analizuje jakość harmonogramu, uwzględniając czas produkcji, konfiguracji i brak przerw między zadaniami.

3 Wyniki algorytmu dla proponowanych danych wejściowych

Algorytm zwraca najlepszy odnaleziony diagram jako interaktywny graph Gantta, którego działanie zostało zaprezentowane w dołączonym filmie. Wyniki diagramu oraz wykres przedstawiający najlepszych osobników algorytmu genetycznego poprzez generacje znajdują się na Figure 1, Figure 2. Przerwy pomiędzy punktami na grafie oznaczają czasy przebrojeń między produktami.



Figure 1: Ostateczny diagram produkcji

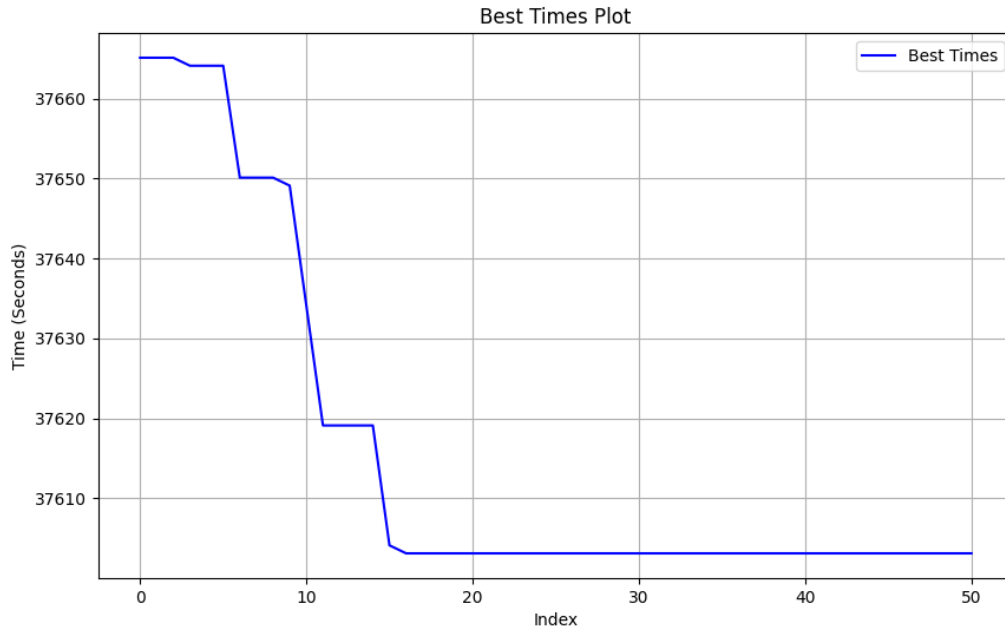


Figure 2: Wykres najlepszego czasu algorytmu genetycznego poprzez generacje