



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE
Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

Projekt dyplomowy

Modelowanie i optymalizacja produkcji rolnej
Modeling and Optimization of Agricultural Production

Autor:	Piotr Paweł Hudaszek
Kierunek studiów:	Automatyka i Robotyka
Opiekun pracy:	Dr inż. Piotr Kadłuczka

Kraków, 2022

Spis treści

Wstęp.....	3
1 Opis zagadnienia	4
1.1 Model opisowy	4
1.2 Model matematyczny	4
1.3 Model a rzeczywisty problem.....	6
2 Opracowanie algorytmu ewolucyjnego	7
2.1 Zasada działania algorytmów ewolucyjnych.....	7
2.2 Schemat algorytmu	7
2.3 Mutacja	7
2.4 Ocena rozwiązania i funkcja kary	7
2.5 Selekcja.....	7
2.6 Krzyżowanie	7
2.7 Parametry algorytmu oraz warunek zakończenia	7
3 Implementacja programu	8
3.1 Struktury danych.....	8
3.2 Implementacja mutacji, selekcji i krzyżowania.....	8
3.3 Implementacja interfejsu graficznego	8
4 Testy programu	9
4.1 <TODO>.....	9
5 Podsumowanie	10
6 Literatura.....	11

Wstęp

<TODO>

Program ma za zadanie zoptymalizować terminarz upraw na określonej liczbie pól tak aby jak najlepiej wykorzystać dostępne zasoby. Został on napisany w Pythonie jako że jest to jeden z bardziej popularnych języków programowania który wciąż się rozwija i zyskuje popularność. Jako algorytm optymalizacji wybrano algorytm genetyczny.

W pierwszym rozdziale został opisany model matematyczny wraz z rozważaniem na temat użytych uproszczeń i przydatności modelu w realnych zastosowaniach. Rozdział drugi jest poświęcony przystosowaniu algorytmu genetycznego do podanego problemu w tym opisanie użytych operatorów genetycznych. W dalszej części pracy zostały zaprezentowane rozwiązania implementacyjne oraz przeprowadzone testy.

1 Opis zagadnienia

1.1 Model opisowy

Opisywane zagadnienie polega na optymalizacji zysku z uprawy w przedsiębiorstwie rolnym. Jest możliwość uprawy różnych rodzajów produktów. Każdy rodzaj potrzebuje określone zasoby na danym etapie uprawy. Przedsiębiorca ma dostęp do pewnej ilości każdego rodzaju zasobów. Zasoby dzielą się na dwie klasy: zasoby dzienne które są przydzielone na dany dzień oraz zasoby całkowite przydzielone na cały rozpatrywany okres uprawy. Jeśli uprawa zostanie przeprowadzona i zasobów nie zabraknie to przedsiębiorca uzyska określony bazowy zysk z jednostki pola, który zależy od rodzaju produktu. Rozpatrywane przedsiębiorstwo ma określoną ilość pól. Każde pole można opisać jego obszarem oraz współczynnikami które określają jak dobrze dane pole pasuje do uprawy danego rodzaju produktu. Aby uzyskać rzeczywisty zysk należy przemnożyć zysk bazowy z współczynnikiem dopasowania oraz wielkością pola.

Optymalizacja polega na wyborze jaki produkt jest uprawiany na danym polu oraz kiedy rozstanie rozpoczęta uprawa. Na polu można uprawiać kilka produktów po sobie. Data rozpoczęcia uprawy jest ograniczona przez ramy czasowe danego produktu.

1.2 Model matematyczny

Dane:

$t \in T$ - dzień

$p \in P$ - pole dostępne do uprawy

S_p - powierzchnia pola p

y_{pn} - rodzaj n -tej rozpoczętej uprawy na polu p

x_{pn} - dzień rozpoczęcia n -tej uprawy na polu p

d_y - dochód z jednostki pola na którym był uprawiany produkt y

w_{py} - współczynnik dopasowania pola p do produktu y

$z_d \in Z_d$ - dostępny zasób dzienny (odnawialny po każdym dniu)

$z_c \in Z_c$ - dostępny zasób całkowity (przydzielony na cały rozpatrywany okres uprawy)

z_{dyt} - zasoby dzienne potrzebne do uprawy produktu y w t dniu uprawy

$z_{c yt}$ - zasoby całkowite potrzebne do uprawy produktu y w t dniu uprawy

t_{y0} - pierwszy dzień w którym można zacząć uprawę produktu y

t_{yN} - ostatni dzień w którym można zacząć uprawę produktu y

o_y - czas trwania uprawy produktu y

Postać rozwiązania:

Lista dwu elementowych wektorów dla każdego pola gdzie pierwszy element x oznacza dzień rozpoczęcia a drugi y rodzaj uprawy.

$$\left\{ \begin{array}{cccc} (x_{11}, y_{11}) & (x_{12}, y_{12}) & \cdots & (x_{1N_1}, y_{1N_1}) \\ (x_{21}, y_{21}) & (x_{22}, y_{22}) & \cdots & (x_{2N_2}, y_{2N_2}) \\ \vdots & \vdots & \ddots & \vdots \\ (x_{PN_1}, y_{PN_1}) & (x_{PN_2}, y_{PN_2}) & \cdots & (x_{PN_P}, y_{PN_P}) \end{array} \right\}$$

Funkcja celu:

Zysk ze sprzedaży uzyskanych produktów .

$$f(x_{pn}, y_{pn}) = \sum_{p \in P} \sum_{n \in N_p} S_p d_{y_{pn}} w_{py_{pn}} \rightarrow \max$$

Ograniczenia:

Dla każdego dnia t każdego rodzaju zasobu dziennego ilość użyta nie może przekroczyć dostępnej.

$$\forall z_d \in Z_d \quad \forall t \in T \quad \sum_{p \in P} z_{dtp} \leq z_d$$

Dla każdego rodzaju zasobu całkowitego ilość użyta przez cały okres uprawy nie może przekroczyć dostępnej.

$$\forall z_c \in Z_c \quad \sum_{t \in T} \sum_{p \in P} z_{ctp} \leq z_c$$

Uprawę produktu typu y można zacząć w wyznaczonym oknie czasowym.

$$t_{y0} \leq o_y \leq t_{yN}$$

W danym dniu na jednym polu można uprawiać tylko jeden rodzaj produktu.

$$\forall p \in P \forall n \in N_p \quad x_{pn} + d_y < x_{pn+1}$$

1.3 Model a rzeczywisty problem

Tak sformułowany model pozwala oddać typowe zależności podczas planowania uprawy tak jak ograniczenie terminu rozpoczęcia uprawy i konieczne zasoby. Ilość zasobów nie jest zdefiniowana na sztywno pozwala to użytkownikowi programu wprowadzić zasoby specyficzne dla danego typu uprawy i sprawić że program jest bardziej uniwersalny. Model można też zastosować do innych zagadnień niż rolnicze na przykład do bilansowania zasobów w zakładzie produkcyjnym.

Model nie bierze pod uwagę wpływu nieprzewidywalnych zmian cen oraz pogody. Natomiast użytkownik może wprowadzić odpowiednio większe potrzebne ilości zasobów lub dłuższy czas uprawy aby mieć pewien margines bezpieczeństwa.

Mimo tych ograniczeń uważam że model może być przydatny. Szczególnie w uprawie szklarniowej gdzie wpływ pogody jest ograniczony oraz w przypadku gdy optymalne wykorzystanie któregoś z zasobów jest kluczowe. Tak jest na przykład w przypadku rejonów z ograniczoną ilością wody[1].

2 Opracowanie algorytmu ewolucyjnego

2.1 Zasada działania algorytmów ewolucyjnych

Dlaczego działają, kiedy nie, inspirowane naturą, bloki budujące

2.2 Schemat algorytmu

Programy ewolucyjne są zbudowane według poniższego ogólnego schematu [Michalewicz Zbigniew, Algorytmy genetyczne ...].

```
procedure program ewolucyjny
begin
  t ← 0
  ustal początkowe P(t)
  oceń P(t)
  while (not warunek zakończenia) do
  begin
    t ← t + 1
    wybierz P(t) z P(t-1)
    zmień P(t)
    oceń P(t)
  end
end
```

W tej pracy przedstawię

....

2.3 Mutacja

2.4 Ocena rozwiązania i funkcja kary

2.5 Selekcja

2.6 Krzyżowanie

2.7 Parametry algorytmu oraz warunek zakończenia

3 Implementacja programu

3.1 Struktury danych

Opisy najważniejszych klas, opis implementacji postaci rozwiązania

3.2 Implementacja mutacji, selekcji i krzyżowania

3.3 Implementacja interfejsu graficznego

Rozważanie różnych narzędzi, uzasadnienie wyboru. Opis API pomiędzy backend i frontend.

Opis ze screenshotami.

4 Testy programu

4.1 <TODO>

5 Podsumowanie

6 Literatura