

Sprawozdanie Metody Optymalizacji 2

Bartosz Michalak

Maj 2025

1 Zadanie 1

1. Model matematyczny

(a) Zmienne decyzyjne

Niech:

x_i liczba (całkowita nieujemna) standardowych desek przeciętych zgodnie z i -tym wzorcem cięcia, gdzie $i = 1, 2, \dots, N$ (zmienne całkowite nieujemne),

e_j liczba (całkowita nieujemna) nadmiarowo wyprodukowanych desek o szerokości w_j (zmienne całkowite nieujemne).

Każdy wzorec określa, ile sztuk danego rozmiaru (np. 7, 5, 3 cale) zawiera jedno cięcie. Nadmiarowe sztuki e_j pozwalają modelowi znaleźć rozwiązanie nawet wtedy, gdy dokładne dopasowanie wzorców do zapotrzebowania nie jest możliwe bez strat.

(b) Ograniczenia

Dla każdej szerokości w_j (np. 7, 5, 3 cali), wymagamy, aby liczba uzyskanych desek (łącznie ze sztukami nadmiarowymi) była równa zapotrzebowaniu d_j :

$$\sum_{i=1}^N a_{ij}x_i = d_j + e_j \quad \text{dla } j = 1, 2, \dots, k$$

gdzie:

a_{ij} liczba desek o szerokości w_j w i -tym wzorcu cięcia,

d_j zapotrzebowanie na deski o szerokości w_j .

(c) Funkcja celu

Minimalizujemy łączną ilość zmarnowanego materiału oraz karę za nadmiarowe cięcia:

$$\min \sum_{i=1}^N r_i x_i + \sum_{j=1}^k w_j e_j$$

gdzie:

r_i ilość odpadu (w calach) powstała przy cięciu deski zgodnie z wzorcem i ,

e_j liczba nadmiarowych sztuk o szerokości w_j ,

w_j szerokość deski danego typu – kara proporcjonalna do wielkości niepotrzebnie wyciętej deski.

2. Wyniki oraz ich interpretacja

Dane wejściowe

- Szerokość standardowej deski: 22 cale
- Zamówienia:
 - 110 desek o szerokości 7 cali
 - 120 desek o szerokości 5 cali
 - 80 desek o szerokości 3 cali

Wyniki

Z wykorzystaniem algorytmu programowania całkowitoliczbowego i rozwiązania za pomocą solvera Cbc:

- Znaleziono 4 wzorce cięcia.
- Łączna liczba zużytych desek: **74**
- Łączna ilość odpadu: **18** cali
- Użyte wzorce cięcia:
 - Wzorzec 14: [5, 7, 3, 7] – użyto 45 razy
 - Wzorzec 20: [5, 3, 7, 5] – użyto 3 razy
 - Wzorzec 30: [7, 5, 5, 5] – użyto 17 razy
 - Wzorzec 75: [3, 3, 5, 3, 5, 3] – użyto 9 razy

Interpretacja

Model pozwala na znalezienie optymalnego sposobu cięcia desek przy zadanym zapotrzebowaniu, który minimalizuje ilość odpadów. W tym przypadku udało się znaleźć rozwiązanie bardzo bliskie idealnemu: wszystkie zamówienia zostały zrealizowane przy użyciu 74 standardowych desek i czterech typów wzorców cięcia, generując jedynie 18 cali odpadu.

Zadanie 2

1. Model matematyczny

(a) Zmienne decyzyjne

W problemie harmonogramowania z pojedynczą maszyną i minimalizowaniem ważonych czasów zakończenia, zmienne decyzyjne obejmują:

S_j czas rozpoczęcia zadania j (liczba całkowita nieujemna, $S_j \geq 0$),

C_j czas zakończenia zadania j (liczba całkowita nieujemna, $C_j \geq 0$),

y_{ij} zmienna binarna, która jeśli jest równa 1 oznacza, że zadanie i jest wykonywane przed zadaniem j .

Niech p_j oznacza czas przetwarzania zadania j , a r_j oznacza czas gotowości (ready time) zadania j .

(b) Ograniczenia

1. Dla każdego zadania j jego czas rozpoczęcia musi być większy lub równy czasowi gotowości r_j :

$$S_j \geq r_j \quad \text{dla } j = 1, 2, \dots, n$$

2. Czas zakończenia każdego zadania jest równy jego czasowi rozpoczęcia plus czas przetwarzania:

$$C_j = S_j + p_j \quad \text{dla } j = 1, 2, \dots, n$$

3. Dla każdej pary zadań i i j musimy określić, które zadanie wykonane jest przed drugim:

$$S_j \geq C_i - M \cdot (1 - y_{ij}) \quad \text{dla } i, j = 1, 2, \dots, n, \quad i \neq j$$

gdzie M to „duża liczba” (np. suma czasów wszystkich zadań).

4. Zapewnienie, że jedno zadanie wykonuje się przed drugim (i nie ma cykli):

$$y_{ij} + y_{ji} = 1 \quad \text{dla } i, j = 1, 2, \dots, n, \quad i \neq j$$

(c) Funkcja celu

Celem jest minimalizacja sumy ważonych czasów zakończenia:

$$\min \sum_{j=1}^n w_j C_j$$

gdzie:

w_j waga zadania j ,

C_j czas zakończenia zadania j .

Interpretacja: Celem jest minimalizacja sumy czasów zakończenia zadań, przy czym każde zadanie ma przypisaną wagę, która podkreśla jego znaczenie.

2. Wyniki oraz ich interpretacja

Zostały przeprowadzone testy jednostkowe dla różnych przypadków danych wejściowych:

- **Przypadek 1 — 4 zadania:**
 - Czasy wykonania: $p = [3, 2, 4, 1]$
 - Wagi: $w = [2, 1, 3, 4]$
 - Moment gotowości: $r = [0, 1, 2, 0]$
 - Wynik: $\sum w_j C_j = 46$ (wartość optymalna)
- **Przypadek 2 — brak ograniczeń dostępności:**
 - $p = [2, 1, 3], w = [1, 2, 3], r = [0, 0, 0]$
 - Wynik: poprawny, optymalna wartość dodatnia.
- **Przypadek 3 — każde zadanie dostępne w innym czasie:**
 - $p = [5, 3, 2], w = [4, 1, 2], r = [0, 5, 10]$
 - Każde zadanie rozpoczęto po swoim czasie gotowości.
- **Przypadek 4 — 5 zadań z różnymi wagami i dostępnościami:**
 - $p = [3, 4, 7, 5, 3], w = [2, 2, 3, 5, 10], r = [1, 2, 2, 2, 6]$
 - Optymalna wartość funkcji celu: **267**
- **Przypadek 5 — jedno zadanie:**
 - $p = [1], w = [1], r = [1]$
 - Wynik: $\sum w_j C_j = 2$

Interpretacja

Model poprawnie uwzględnia momenty gotowości zadań oraz przypisane im wagi. Dla każdego testu:

- Zadania rozpoczynają się nie wcześniej niż w momencie ich gotowości ($S_j \geq r_j$),
- Zakończenia są zgodne z definicją $C_j = S_j + p_j$,
- Wyznaczona suma ważonych czasów zakończenia jest minimalna (zgodna z oczekiwaniami testów).

Zadanie 3

1. Model matematyczny

(a) Zmienne decyzyjne

- $s_j \in \mathbb{Z}_{\geq 0}$ — czas rozpoczęcia zadania j , jednostka: jednostka czasu.
- $x_{jk} \in \{0, 1\}$ — zmienna przypisania: $x_{jk} = 1$ oznacza, że zadanie j jest przydzielone do maszyny k ; w przeciwnym przypadku $x_{jk} = 0$.
- $C_{\max} \in \mathbb{Z}_{\geq 0}$ — maksymalny czas zakończenia zadań (makespan), który model minimalizuje.
- $\delta_{ij} \in \{0, 1\}$ — zmienna porządku: $\delta_{ij} = 1$ oznacza, że zadanie i jest wykonywane przed zadaniem j na tej samej maszynie; inaczej $\delta_{ij} = 0$.

(b) Ograniczenia

1. Przypisanie zadania do jednej maszyny:

$$\sum_{k=1}^m x_{jk} = 1 \quad \forall j = 1, \dots, n$$

Każde zadanie jest przypisane dokładnie do jednej maszyny.

2. Ograniczenia poprzedzające:

$$s_j \geq s_i + p_i \quad \forall i \in \text{poprzednicy}(j)$$

Jeżeli zadanie i poprzedza j , to j musi rozpocząć się po zakończeniu i .

3. Ograniczenia rozłączności na jednej maszynie:

$$\begin{aligned} s_i + p_i &\leq s_j + M(1 - \delta_{ij} + 1 - (x_{ik} + x_{jk} - 1)) \\ s_j + p_j &\leq s_i + M(\delta_{ij} + 1 - (x_{ik} + x_{jk} - 1)) \quad \forall i < j, \forall k \end{aligned}$$

Jeśli dwa zadania są przypisane do tej samej maszyny, to jedno musi zakończyć się przed rozpoczęciem drugiego.

4. Warunki spójności porządku:

$$\delta_{ij} + \delta_{ji} = 1 \quad \forall i \neq j$$

Jeśli $i \neq j$, to jedno z zadań musi być wykonane przed drugim (jeśli są na tej samej maszynie).

5. Definicja makespanu:

$$s_j + p_j \leq C_{\max} \quad \forall j = 1, \dots, n$$

Makespan musi być nie mniejszy niż zakończenie każdego zadania.

(c) Funkcja celu

$$\min C_{\max}$$

Celem modelu jest minimalizacja maksymalnego czasu zakończenia zadania, czyli skrócenie całkowitego czasu wykonania wszystkich zadań (makespan). Funkcja celu odzwierciedla potrzebę jak najszybszego zakończenia całej produkcji lub projektu na wielu maszynach.

2. Wyniki i ich interpretacja

Wszystkie testy zakończyły się sukcesem i potwierdziły poprawność modelu dla różnych układów zadań oraz maszyn.

Test 1: Przykład z polecenia

- **Liczba zadań:** 9
- **Czasy trwania:** [1, 2, 1, 2, 1, 1, 3, 6, 2]
- **Relacje poprzedzające:** Jak w poleceniu
- **Liczba maszyn:** 3
- **Optymalny makespan:** $C_{\max} = 9$
- **Interpretacja:** Harmonogram został ułożony w taki sposób, by spełnić zależności między zadaniami i zrównoważyć obciążenie między trzema maszynami, co pozwoliło osiągnąć optymalny czas zakończenia.
- **Harmonogram (tekstowy Gantt):**

Machine 1:	1							6
Machine 2:	2	2	5	8	8	8	8	8
Machine 3:	3		4	4	7	7	7	9

Test 2: Łańcuch liniowy na jednej maszynie

- **Liczba zadań:** 3
- **Czasy trwania:** [2, 2, 2]
- **Liczba Maszyn:** 1
- **Relacje poprzedzające:** $1 \rightarrow 2 \rightarrow 3$
- **Optymalny makespan:** $C_{\max} = 6$
- **Interpretacja:** Każde zadanie musi być wykonane po poprzednim, co skutkuje sumowaniem czasów — wynik jest zgodny z oczekiwaniami.
- **Harmonogram:**

Machine 1:	1	1	2	2	3	3
------------	---	---	---	---	---	---

Test 3: Zadania równoległe bez zależności

- **Zadania:** 3 niezależne zadania o czasie 3
- **Maszyny:** 3
- **Optymalny makespan:** $C_{\max} = 3$
- **Interpretacja:** Brak zależności pozwala na jednoczesne rozpoczęcie wszystkich zadań, każde przypisane do innej maszyny — co pozwala zakończyć wszystko w czasie trwania najdłuższego zadania.
- **Harmonogram:**

Machine 1:	3	3	3
Machine 2:	2	2	2
Machine 3:	1	1	1

Test 4: Niezależne zadania na jednej maszynie

- **Zadania:** 3 niezależne zadania o czasach $[1, 2, 3]$
- **Maszyny:** 1
- **Optymalny makespan:** $C_{\max} = 6$
- **Interpretacja:** Ponieważ nie ma zależności, zadania są wykonywane jedno po drugim — całkowity czas to suma ich długości.
- **Harmonogram:**

Machine 1: 3 3 3 2 2 1

Podsumowanie

Model poprawnie obsługuje zarówno przypadki z zależnościami, jak i zadania niezależne, oraz działa zarówno dla jednej, jak i wielu maszyn. Wszystkie testy zakończyły się sukcesem

Zadanie 4

1. Model matematyczny

(a) Zmienne decyzyjne

- $x_{j,t}$: Zmienna binarna, która wynosi 1, jeśli zadanie j rozpoczyna się w czasie t , w przeciwnym razie wynosi 0.
- s_j : Czas rozpoczęcia zadania j (jednostki czasu). Zmienna ta oznacza, w którym momencie czasu zadanie j zaczyna się.
- f_j : Czas zakończenia zadania j (jednostki czasu). Zmienna ta oznacza, w którym momencie czasu zadanie j kończy się.
- $d_{k,t}$: Całkowite zapotrzebowanie na zasób typu k w chwili t (jednostki zasobu).
- C_{max} : Całkowity czas zakończenia ostatniego zadania (makespan). Zmienna ta reprezentuje czas, w którym ostatnie zadanie zostanie zakończone.

Dla każdego zadania j , F_j oznacza ostatni możliwy moment startu zadania, który jest określony jako:

$$F_j = t_{max} - T_j + 1$$

gdzie t_{max} to szacunkowy górny limit na czas zakończenia wszystkich zadań (suma wszystkich T_j), a T_j to czas trwania zadania j .

(b) Ograniczenia

1. Dokładnie jeden start każdego zadania

$$\sum_{t=1}^{F_j} x_{j,t} = 1 \quad \forall j = 1, \dots, n$$

Każde zadanie zaczyna się dokładnie raz.

2. Powiązanie zmiennych binarnych z czasem rozpoczęcia

$$s_j = \sum_{t=1}^{F_j} t \cdot x_{j,t} \quad \forall j$$

Wyznaczenie momentu startu zadania j na podstawie zmiennych binarnych.

3. Zakończenie zadania

$$f_j = s_j + T_j - 1 \quad \forall j$$

Czas zakończenia zadania j to jego start plus czas trwania (minus jeden, bo start jest już pierwszym momentem z czasu trwania).

4. Zapotrzebowanie na zasoby w każdej jednostce czasu

$$d_{k,t} = \sum_{\substack{j=1 \\ t-T_j+1 \leq \tau \leq \min(t, F_j)}}^n x_{j,\tau} \cdot R_{j,k} \quad \forall k, t$$

Łączne zapotrzebowanie na zasób k w chwili t to suma zapotrzebowań wszystkich zadań trwających w tym czasie.

5. Zależności między zadaniami

$$s_j \geq f_b + 1 \quad \forall j, \forall b \in \text{poprzednicy}(j)$$

Zadanie j może rozpocząć się dopiero po zakończeniu wszystkich jego poprzedników.

6. Ograniczenia zasobowe

$$d_{k,t} \leq N_k \quad \forall k, t$$

Całkowite zużycie zasobu k w czasie t nie może przekraczać jego dostępnej ilości.

7. Powiązanie z makespanem

$$f_j \leq C_{max} \quad \forall j$$

Makespan musi być nie mniejszy niż czas zakończenia dowolnego zadania.

(c) Funkcja celu

Funkcja celu modelu jest następująca:

$$\min C_{max}$$

Celem modelu jest zminimalizowanie czasu zakończenia ostatniego zadania (makespan). Chcemy znaleźć taki harmonogram, który pozwala ukończyć wszystkie zadania jak najszybciej, z zachowaniem ograniczeń zasobowych i logicznych (precedencji).

2. Wyniki i ich interpretacja

Instancja problemu: 8 czynności, 1 typ zasobu odnawialnego (limit: 30 jednostek).

Czynność	Czas trwania t_j	Zapotrzebowanie r_j	Poprzednicy
1	50	9	—
2	47	17	1
3	55	11	1
4	46	4	1
5	32	13	2
6	57	7	3, 4
7	15	7	4
8	62	17	5, 6, 7

Wyniki optymalizacji (solver: Cbc):

- $C_{max} = 237$
- Wszystkie ograniczenia zasobowe i kolejnościowe zostały spełnione.

Interpretacja: Harmonogram spełnia wszystkie ograniczenia, a poziomy zużycia zasobów nie przekraczają ustalonych limitów. Wartość $C_{max} = 237$ wyznacza najkrótszy czas, w którym możliwe jest wykonanie wszystkich czynności z zachowaniem ograniczeń.