Metody Optymalizacji - Laboratorium 2

Wojciech Sęk

6 maja 2023

1 Zadanie 1

1.1 Model

1.1.1 Zmienne decyzyjne

Dla m danych sposobów pocięcia standardowej deski na deski żądanych szerokości przez x_j dla $j \in [m]$ oznaczamy liczbę desek pociętych w j-ty sposób.

1.1.2 Ograniczenia

- Niech $\alpha_{j,i}$ oznacza liczbę desek rodzaju i wyciętych, gdy używamy j-tej metody cięcia deski. Niech δ_i oznacza podaż na deski rodzaju i. Suma wyprodukowanych desek danego rodzaju musi być równa podaży na dany rodzaj:

$$(\forall i \in [n]) \left(\sum_{j=1}^{m} x_j \cdot \alpha_{j,i} = \delta_i \right)$$

1.2 Funkcja celu

Minimalizujemy sumę odpadów z produkcji wszystkich desek. Niech λ_i oznacza szerokość odpadu w i-tej metodzie. Funkcją celu jest wtedy

$$f(\mathbf{x}) = \sum_{i=1}^{m} x_i \cdot \lambda_i$$

1.3 Wyniki

Dla standardowej deski szerokości 22 i żądania na 80 desek szerokości 3, 120 desek szerokości 5 i 110 desek szerokości 7 optymalnie będzie rozciąć deski wg następującej tabeli, gdzie i oznacza numer metody (łącznie metod było 42, pomijamy wiersze dla $x_i = 0$). Niech $\alpha_i(j)$ oznacza liczbę desek szerokości i uzyskanych w danej metodzie:

i	x_i	$x_i \cdot \alpha_i(3)$	$x_i \cdot \alpha_i(5)$	$x_i \cdot \alpha_i(7)$	$x_i \cdot \lambda_i$
26	11	0	0	33	11
33	9	45	0	9	0
37	33	0	99	33	0
39	14	14	14	28	0
42	7	21	7	7	7
Σ_i	11	80	120	110	18

Mamy wtedy resztki szerokości 18.

2 Zadanie 2

2.1 Model

2.1.1 Zmienne decyzyjne

Dla n zadań i horyzontu czasowego T wprowadzamy binarną zmienną $x_{i,j}$, która dla $i \in [n], j \in [T]$ oznacza, że zadanie i rozpoczyna się w momencie $j-1$. T to najpóźniejszy możliwy czas rozpoczęcia zadania, który równa się

$$\max_{i \in [n]} r_i + \sum_{i \in [n]} p_i + 1$$

gdzie r_i to najwcześniejszy moment rozpoczęcia zadania i a p_i to czas jego trwania. W najgorszym przypadku zadania rozpoczną się z największym opóźnieniem r, muszą trwać co najmniej tyle co suma ich wykonywania. Czynnik $+1$ wynika z tego, że $x_{*,t}$ oznacza moment $t-1$.

2.1.2 Ograniczenia

- Każde zadanie rozpoczyna się dokładnie raz:

$$(\forall i \in [n]) \left(\sum_{t \in [T]} x_{j,t} = 1 \right)$$

- j-te zadanie rozpoczyna się nie wcześniej niż w momencie r_j

$$(\forall j \in [n]) \left(\sum_{t \in [T]} (t-1) \cdot x_{j,t} \geqslant r_j \right)$$

- W dowolnym momencie czasu wykonujemy co najwyżej jedno zadanie

$$(\forall t \in [T]) \left(\sum_{j \in n} \left(\sum_{s \in \max(1, t+1-p_j)}^{t} x_{j,s} \right) \leqslant 1 \right)$$

2.2 Funkcja celu

Niech w_j oznacza wagę j-ego zadania. Minimalizujemy ważoną sumę czasów zakończenia zadań

$$f(\mathbf{x}) = \sum_{j \in [n]} \sum_{t \in [T]} w_j * (t - 1 + p_j) * x_{j,t}$$

2.3 Wyniki

W ramach sprawdzenia implementacji użyłem dwóch zadań o następujących parametrach

$$p_1 = 10, \quad p_2 = 10, \quad w_1 = 10000, \quad p_2 = 1, \quad r_1 = 5, \quad r_2 = 0$$

Model określił, że zadanie pierwsze powinno rozpocząć się w momencie 5 a drugie w momencie 15. Jest to zgodne z oczekiwaniami, że mimo większego ograniczenia r na zadanie pierwsze, zaczynamy od niego, ponieważ ma dużo większą wagę.

3 Zadanie 3

3.1 Model

3.1.1 Zmienne decyzyjne

Dla n zadań, m maszyn i horyzontu czasowego T wprowadzamy binarną zmienną $x_{i,j,k}$, która dla $i \in [n], j \in [T], k \in [m]$ oznacza, że zadanie i rozpoczyna się w momencie $j-1$ na maszynie k. T to najpóźniejszy możliwy czas rozpoczęcia zadania, który równa się

$$\sum_{i\in[n]} p_i + 1$$

gdzie p_i to czas trwania zadania i. W najgorszym przypadku zadania muszą trwać co najmniej tyle co suma ich wykonywania. Czynnik $+1$ wynika z tego, że $x_{*,t,*}$ oznacza moment $t-1$.
Wprowadzamy również zmienną c_{MAX}, która ma ograniczyć od góry czas zakończenia dowolnego zadania.

3.1.2 Ograniczenia

- Każde zadanie kończy najpóźniej w momencie c_{MAX}:

$$(\forall j \in [n])(\forall t \in [T])(\forall k \in [m])\left((t-1+p_j)\cdot x_{j,t,k} \leqslant c_{MAX}\right)$$

- Każde zadanie rozpoczyna się dokładnie raz na dokładnie jednej maszynie:

$$(\forall i \in [n])\left(\sum_{t\in[T]}\sum_{k\in[m]} x_{j,t,k} = 1\right)$$

- W dowolnym momencie czasu na jednej maszynie wykonujemy co najwyżej jedno zadanie

$$(\forall t \in [T])(\forall k \in [m])\left(\sum_{j\in n}\left(\sum_{s\in\max(1,t+1-p_j)}^{t} x_{j,s,k}\right) \leqslant 1\right)$$

- Niech π_i oznacza zbiór poprzedników zadania i, gdzie przez poprzednika rozumiemy zadanie, którego czas zakończenia musi być mniejszy bądź równy czasowi rozpoczęcia zadania i. Warunek można zapisać jako:

$$(\forall b \in [n])(\forall a \in \pi_b)\left(\sum_{t=1}^{T-p_a+1}\sum_{k\in[m]}(t+p_a-1)\cdot x_{a,t,k} \leqslant \sum_{t=1}^{T-p_b+1}\sum_{k\in[m]}(t-1)\cdot x_{b,t,k}\right)$$

3.2 Funkcja celu

Minimalizujemy czas zakończenia ostatniego zadania

$$f(\mathbf{x}, c_{MAX}) = c_{MAX}$$

3.3 Wyniki

Dla przykładowych danych zaplanowano następujący schemat o czasie wykonywania 9:

Przedział czasu	$M1$	$M2$	$M3$
(0,1)	2	3	
(1,2)	2		1
(2,3)		5	4
(3,4)		8	4
(4,5)		8	7
(5,6)	6	8	7
(6,7)		8	7
(7,9)		8	9

4 Zadanie 4

4.1 Model

4.1.1 Zmienne decyzyjne

Dla n zadań, i horyzontu czasowego T wprowadzamy binarną zmienną $x_{i,j}$, która dla $i \in [n], j \in [T]$ oznacza, że zadanie i rozpoczyna się w momencie $j-1$. T to najpóźniejszy możliwy czas rozpoczęcia zadania, który równa się

$$\sum_{i \in [n]} \tau_i + 1$$

gdzie τ_i to czas trwania zadania i. W najgorszym przypadku zadania muszą trwać co najmniej tyle co suma ich wykonywania. Czynnik $+1$ wynika z tego, że $x_{*,t}$ oznacza moment $t-1$.
Wprowadzamy również zmienną c_{MAX}, która ma ograniczyć od góry czas zakończenia dowolnego zadania.

4.1.2 Ograniczenia

- Każde zadanie kończy najpóźniej w momencie c_{MAX}:

$$(\forall j \in [n])(\forall t \in [T])\left((t-1+\tau_j) \cdot x_{j,t} \leqslant c_{MAX}\right)$$

- Każde zadanie rozpoczyna się dokładnie raz:

$$(\forall i \in [n])\left(\sum_{t \in [T]} x_{j,t} = 1\right)$$

- W dowolnym momencie dla każdego zasobu nie możemy przekroczyć jego chwilowego zużycia. Niech p to liczba zasobów, N_i oznacza ilość zasobu i oraz niech $r_{j,i}$ oznacza zapotrzebowanie zadania j na zasób i. Wtedy warunek możemy określić jako:

$$(\forall t \in [T])(i \in [p])\left(\sum_{j=1}^{n} \sum_{s=\max(1,t+1-\tau_j)}^{t} x_{j,s} \cdot r_{j,i} \leqslant N_i\right)$$

- Niech π_i oznacza zbiór poprzedników zadania i, gdzie przez poprzednika rozumiemy zadanie, którego czas zakończenia musi być mniejszy bądź równy czasowi rozpoczęcia zadania i. Warunek można zapisać jako:

$$(\forall b \in [n])(\forall a \in \pi_b)\left(\sum_{t=1}^{T-\tau_a+1} (t+\tau_a-1) \cdot x_{a,t} \leqslant \sum_{t=1}^{T-\tau_b+1} (t-1) \cdot x_{b,t}\right)$$

4.2 Funkcja celu

Minimalizujemy czas zakończenia ostatniego zadania

$$f(\mathbf{x}, c_{MAX}) = c_{MAX}$$

4.3 Wyniki

Dla przykładowych danych znaleziono następujący schemat o czasie wykonywania 237:

Przedział czasu	**Zużycie zasobu r_1**	**Wykonywane zadania**
(0,50)	9	1
(50,54)	4	4
(54,96)	15	3,4
(96,109)	28	2,3
(109,116)	17	2
(109,143)	24	2,6
(143,144)	20	5,6
(144,159)	27	5,6,7
(159,173)	20	5,6
(173,175)	13	5
(173,237)	17	8