1 Cięcie desek

Celem modelu jest minimalizacja liczby standardowych desek o szerokości 22 cali, które należy przeciąć w taki sposób, aby zaspokoić zadane zapotrzebowanie na deski o szerokościach 7, 5 i 3 cali. Możliwe są różne wzory cięcia, a firma chce zminimalizować liczbę użytych desek oraz odpad materiałowy.

1.1 Dane

- $W = 22$ – szerokość standardowej deski (w calach),
- $w = [7, 5, 3]$ – dostępne szerokości desek wynikowych (w calach),
- $d = [110, 120, 80]$ – zapotrzebowanie na deskę szerokości odpowiednio 7, 5 i 3 cali,
- P – zbiór wszystkich dopuszczalnych wzorów cięcia; każdy wzór to trójka $p = (a, b, c)$, oznaczająca liczbę desek o szerokościach odpowiednio 7, 5 i 3 cali uzyskanych z jednej deski standardowej, przy czym:

$$7a + 5b + 3c \leq W.$$

1.2 Model

Zmienne decyzyjne

- $x_p \in \mathbb{Z}_{\geq 0}$ – liczba desek przeciętych według wzoru $p \in P$.

Funkcja celu

Minimalizujemy całkowitą liczbę użytych desek:

$$\min \sum_{p \in P} x_p.$$

Ograniczenia

1. **Zaspokojenie zapotrzebowania:**

$$\sum_{p \in P} a_{ip} \cdot x_p \geq d_i, \quad \text{dla } i = 1, 2, 3,$$

gdzie a_{ip} to liczba desek szerokości w_i wytwarzanych w schemacie cięcia p.

Interpretacja: Łączna liczba uzyskanych kawałków każdej szerokości musi być nie mniejsza niż zapotrzebowanie.

1.3 Wyniki

Model został rozwiązany za pomocą solvera GLPK. Otrzymano optymalne rozwiązanie z minimalnym odpadem:

$$\text{TotalWaste} = 18.0.$$

Wybrane wzory cięcia i liczba ich użyć

- Wzór (1, 0, 5): 9 desek – 1×7 cali, 5×3 cali,
- Wzór (1, 3, 0): 28 desek – 1×7 cali, 3×5 cali,
- Wzór (2, 1, 1): 37 desek – 2×7 cali, 1×5 cali, 1×3 cali.

2 Harmonogramowanie z czasami dostępności i wagami

Celem modelu jest optymalne ustalenie kolejności wykonywania zadań na jednej maszynie, tak aby zminimalizować sumę wagowanych czasów zakończenia $\sum w_j C_j$. Każde zadanie ma określoną długość p_j, wagę w_j oraz moment dostępności (release date) r_j. Zadania mogą się rozpocząć nie wcześniej niż w r_j, a każde musi zostać przypisane do dokładnie jednego miejsca w harmonogramie.

2.1 Dane

Rozważany przykład obejmuje $n = 4$ zadania, których dane zestawiono poniżej:

Zadanie j	1	2	3	4
p_j (czas trwania)	3	2	4	1
w_j (waga)	4	1	3	2
r_j (czas dostępności)	0	1	0	2

Parametr M użyty w modelu jako duża liczba pomocnicza wyznaczony został jako:

$$M = \sum p_j + \max_j r_j = 10.$$

2.2 Model

Zmienne decyzyjne

- C_j – czas zakończenia zadania j,
- $\delta_{i,j} \in \{0, 1\}$ – zmienna binarna, która przyjmuje wartość 1, jeśli zadanie i kończy się przed rozpoczęciem zadania j.

Funkcja celu

Minimalizujemy sumę wagowanych czasów zakończenia:

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

Ograniczenia

1. **Czas dostępności:**

$$C_j \geq r_j + p_j \quad \forall j.$$

2. **Kolejność zadań (jeśli i przed j):**

$$C_i + p_j \leq C_j + M \cdot (1 - \delta_{i,j}) \quad \forall i \neq j.$$

3. **Antysymetria i wykluczenie cykli:**

$$\delta_{i,j} + \delta_{j,i} = 1 \quad \forall i \neq j, \quad \delta_{i,i} = 0.$$

2.3 Wyniki

Model został rozwiązany za pomocą solvera GLPK. Otrzymano optymalny harmonogram z wartością funkcji celu równą:

$$\sum w_j C_j = 54{,}0.$$

Czasy zakończenia zadań

- Zadanie 1: $C_1 = 3{,}0$,
- Zadanie 2: $C_2 = 10{,}0$,
- Zadanie 3: $C_3 = 8{,}0$,
- Zadanie 4: $C_4 = 4{,}0$.

Macierz relacji kolejności $\delta_{i,j}$

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Wiersz i, kolumna j: wartość 1 oznacza, że zadanie i kończy się przed rozpoczęciem zadania j.

3 Harmonogramowanie z poprzedzeniami i wieloma maszynami

Celem modelu jest ustalenie harmonogramu wykonywania zadań na $m = 3$ równoległych maszynach z uwzględnieniem relacji poprzedzeń między zadaniami oraz minimalizacją czasu zakończenia ostatniego zadania (makespanu, $C_{\max}$). Każde zadanie ma określony czas trwania, a relacje poprzedzeń determinują, które zadania muszą być zakończone przed rozpoczęciem innych.

3.1 Dane

Dane wejściowe obejmują $n = 9$ zadań i $m = 3$ identyczne maszyny. Dla każdego zadania j podano:

- p_j – czas trwania zadania j,
- Preds $= \{(i, j)\}$ – relacje poprzedzeń, gdzie $i \prec j$ oznacza, że zadanie i musi zakończyć się przed rozpoczęciem zadania j.

Parametr M (tzw. „duże M") został ustawiony jako suma wszystkich czasów trwania zadań:

$$M = \sum_{j=1}^{n} p_j = 19.$$

Dane liczbowe przedstawiono w tabelach poniżej:

Zadanie j	1	2	3	4	5	6	7	8	9
p_j	1	2	1	2	1	1	3	6	2

Relacje poprzedzeń:

$$\text{Preds} = \{(1, 4), (2, 4), (2, 5), (3, 4), (3, 5), (4, 6), (4, 7), (5, 7), (5, 8), (6, 9), (7, 9)\}$$

3.2 Model

Zmienne decyzyjne

- s_j – czas rozpoczęcia zadania j,
- $y_{j,k} \in \{0, 1\}$ – 1 jeśli zadanie j jest przypisane do maszyny k,
- $z_{i,j,k} \in \{0, 1\}$ – 1 jeśli zadanie i jest wykonywane przed zadaniem j na maszynie k,
- $C_{\max}$ – maksymalny czas zakończenia.

Funkcja celu

$$\min C_{\max}$$

Interpretacja: Minimalizujemy czas zakończenia ostatniego zadania.

Ograniczenia

1. **Każde zadanie jest przypisane do dokładnie jednej maszyny:**

$$\sum_{k=1}^{m} y_{j,k} = 1 \quad \forall j.$$

2. **Relacje poprzedzeń:**

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

3. **Zadania na tej samej maszynie nie mogą się nakładać w czasie:**

$$s_j \geq s_i + p_i - M \cdot (1 - z_{i,j,k}) \quad \forall i \neq j, \forall k,$$

oraz

$$z_{i,j,k} + z_{j,i,k} \geq y_{i,k} + y_{j,k} - 1.$$

4. **Definicja makespanu:**

$$s_j + p_j \leq C_{\max} \quad \forall j.$$

3.3 Wyniki

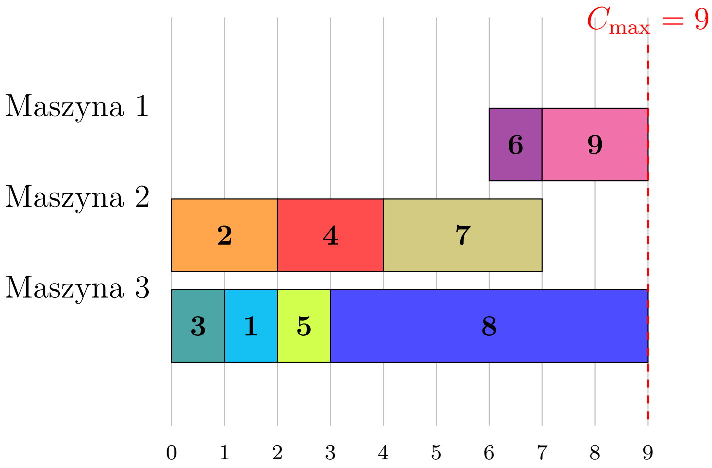
Model został rozwiązany z wykorzystaniem solvera GLPK. Uzyskano harmonogram o minimalnym makespanie równym:

$$C_{\max} = 9{,}0.$$

Przydział zadań do maszyn i czasy rozpoczęcia

- Zadanie 1: maszyna 3, start = 1,0, koniec = 2,0,
- Zadanie 2: maszyna 2, start = 0,0, koniec = 2,0,
- Zadanie 3: maszyna 3, start = 0,0, koniec = 1,0,
- Zadanie 4: maszyna 2, start = 2,0, koniec = 4,0,
- Zadanie 5: maszyna 3, start = 2,0, koniec = 3,0,
- Zadanie 6: maszyna 1, start = 6,0, koniec = 7,0,
- Zadanie 7: maszyna 2, start = 4,0, koniec = 7,0,
- Zadanie 8: maszyna 3, start = 3,0, koniec = 9,0,
- Zadanie 9: maszyna 1, start = 7,0, koniec = 9,0.

Diagram Gantta



Rysunek 1: Diagram Gantta dla optymalnego harmonogramu z makespanem $C_{\max} = 9$

4 Harmonogramowanie z zasobami odnawialnymi i poprzedzeniami

Celem modelu jest wyznaczenie harmonogramu realizacji zadań na jednej maszynie przy ograniczonym zasobie odnawialnym, z uwzględnieniem relacji poprzedzeń między zadaniami. Każde zadanie ma określony czas trwania, zużywa określoną ilość zasobu oraz może rozpocząć się dopiero po zakończeniu poprzedników. Celem jest minimalizacja całkowitego czasu wykonania, czyli makespanu ($C_{\max}$).

4.1 Dane

Zadanie obejmuje $n = 8$ zadań i pojedynczy zasób odnawialny o dostępności $N = 30$ jednostek. Każde zadanie j ma przypisany:

- t_j – czas trwania zadania,
- r_j – ilość zasobu zużywana przez całe zadanie,
- ES_j, LS_j – najwcześniejszy i najpóźniejszy możliwy czas rozpoczęcia,
- Zależności poprzedzeń: zbiór par (i, j) takich, że $i \prec j$.

Parametry zadane w modelu:

Zadanie j	1	2	3	4	5	6	7	8
t_j (czas)	50	47	55	46	32	57	15	62
r_j (zasób)	9	17	11	4	13	7	7	17

Zadania są powiązane zależnościami poprzedzeń. Relacje te definiują zbiór:

$$\text{Preds} = \{(1,2), (1,3), (1,4), (2,5), (3,6), (4,6), (4,7), (5,8), (6,8), (7,8)\}.$$

Model wykorzystuje redukcję horyzontu czasowego za pomocą obliczenia najwcześniejszych ES_j i najpóźniejszych dopuszczalnych startów LS_j dla każdego zadania. Ograniczenia zasobowe uwzględniane są w każdym momencie τ w horyzoncie.

Zmienna decyzyjna $x_{j,\tau} \in \{0, 1\}$ wskazuje, czy zadanie j rozpoczyna się w chwili τ, a zmienna $C_{\max}$ reprezentuje makespan.

4.2 Model

Zmienne decyzyjne

- $x_{j,\tau} \in \{0, 1\}$ – przyjmuje wartość 1, jeśli zadanie j rozpoczyna się w chwili τ,
- $C_{\max}$ – czas zakończenia najpóźniej kończącego się zadania.

Funkcja celu

$$\min C_{\max}$$

Interpretacja: Minimalizujemy czas zakończenia ostatniego zadania.

Ograniczenia

1. **Każde zadanie rozpoczyna się dokładnie raz:**

$$\sum_{\tau=ES_j}^{LS_j} x_{j,\tau} = 1 \quad \forall j.$$

2. **Relacje poprzedzeń:**

$$\sum_{\tau} \tau \cdot x_{j,\tau} \geq \sum_{\tau} \tau \cdot x_{i,\tau} + p_i \quad \forall (i,j) \in \text{Preds}.$$

3. **Definicja makespanu:**

$$\sum_{\tau} (\tau + p_j) \cdot x_{j,\tau} \leq C_{\max} \quad \forall j.$$

4. **Ograniczenia zasobu odnawialnego:** w każdej chwili τ,

$$\sum_{j} r_j \cdot \sum_{\tau'=\max(ES_j, \tau - p_j + 1)}^{\min(LS_j, \tau)} x_{j,\tau'} \leq N.$$

4.3 Wyniki

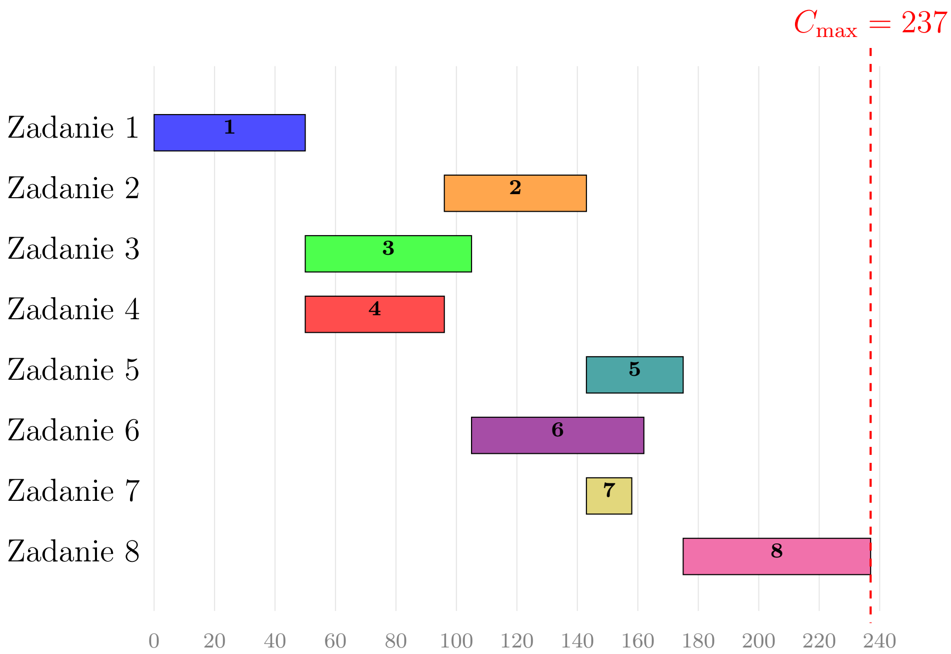
Model został rozwiązany przy użyciu solvera Cbc z limitem czasu 4 minut. Otrzymano rozwiązanie optymalne z wartością makespanu równą:

$$C_{\max} = 237.$$

Czasy rozpoczęcia i zakończenia zadań

- Zadanie 1: start = 0, koniec = 50,
- Zadanie 2: start = 96, koniec = 143,
- Zadanie 3: start = 50, koniec = 105,
- Zadanie 4: start = 50, koniec = 96,
- Zadanie 5: start = 143, koniec = 175,
- Zadanie 6: start = 105, koniec = 162,
- Zadanie 7: start = 143, koniec = 158,
- Zadanie 8: start = 175, koniec = 237.

Diagram Gantta



Rysunek 2: Diagram Gantta dla optymalnego harmonogramu ($C_{\max} = 237$)

Profil zasobów

Zużycie zasobu (maksymalnie 30 jednostek) w kolejnych przedziałach czasu, w których wykorzystanie pozostaje stałe:

- $t \in [0, 50)$: 9 / 30,
- $t \in [50, 96)$: 15 / 30,
- $t \in [96, 105)$: 28 / 30,
- $t \in [105, 143)$: 24 / 30,
- $t \in [143, 158)$: 27 / 30,
- $t \in [158, 162)$: 20 / 30,
- $t \in [162, 175)$: 13 / 30,
- $t \in [175, 237)$: 17 / 30.