Optimizacijski problem za fleksibilni "Job Shop" sustav

April 4, 2025

1 Optimizacijski model

Zadan je skup zadataka J, pri čemu svaki zadatak $i \in J$ odgovara jednom predmetu na traci. Pozicija predmeta i u trenutku razmaranja je (x_i, y_i). Zadan je skup robota R. Za robota $k \in R$ definiraju se granice njegovog radnog prostora W_{k-1} i W_k. Također, definira se i za svakog robota k njegova pozicija (x^k, y^k). Brzine trake je v, a koordinatni sustav postavljen je prema slici (dodati sliku).

Sustav se razmatra za trenutak $t = 0$. Definiraju se sljedeće vrijednosti:

tr_i^k - vrijeme kad posao $i \in J$ postaje dostupan robotu $k \in R$ (engl. release time). Ta vrijednost određena je radnim prostorom robota k i pozicijom predmeta i na traci.

$$tr_i^k = \frac{x_i - W_k}{v}$$

td_i^k - vrijeme kad posao $i \in J$ postaje nedostupan robotu $k \in R$ (engl. deadline time), jer izlazi iz njegovog radnog prostota.

$$tr_i^k = \frac{x_i - W_{k-1}}{v}$$

t_i - trenutak u kojem robot uzima predmet i. Ako robot k uzima predmet, vrijedi:

$$tr_i^k \leq t_i \leq td_i^k$$

T_i^k - vrijeme obrade posla i na stroju k, koje se sastoji od uzimanja predmeta i odlaganja prededmeta. Ova vrijednost ovisi o položaju predmeta u trenutku kad se uzima s trake. Neka je t_i vrijeme u kojem robot k uzima predmet i s trake. Ako je u trenutku $t = 0$ predmet bio na poziciji (x_i, y_i), pozicija predmeta u trenutku t_i je $(x_i - v \cdot t_i, y_i)$, a T_i^k se može izračunati kao:

$$T_i^k = \frac{\sqrt((x^k - x_i + v \cdot t_i)^2 + (y^k - y_i)^2)}{v} = f(t_i)$$

Aproksimirana vrijednost je:

$$T_i^k = \frac{x^k - x_i + v \cdot t_i}{v} + \frac{y^k - y_i}{v}$$

Optimizacijski algoritam treba kao rezultat odrediti sljedeće vrijednosti:

- $z_{ij} = \begin{cases} 1, & \text{ako se predmet uzima s trake tj. zadatak se izvršava} \\ 0, & \text{ako se predmet preskače} \end{cases}$
- t_i - trenutak u kojem se uzima predmet i
- $\alpha_i^k = \begin{cases} 1, & \text{ako je zadatak } i \text{ dodijeljen robotu } k \\ 0, & \text{inače} \end{cases}$
- $\beta_{ij} = \begin{cases} 1, & \text{ako se predmet } i \text{ uzima prije predmeta } j \\ 0, & \text{inače} \end{cases}$

Cilj optimizacije je odrediti prethodne varijable, tako da se minimizira kriterij optimizacije, npr. ukupan broj predmeta koji se uzimaju:

$$\min \sum_{i \in J} z_i,$$

uz sljedeća ograničenja koja moraju vrijediti za svako rješenje (H je oznaka za jako veliki broj):

- $\sum\limits_{k \in R} \alpha_i^k = z_i$ - svaki zadatak koji se izvršava dodijeljen je najviše jednom robotu
- $t_i \geq t_j + T_j^k - (4 - z_i - z_j - \alpha_i^k - \alpha_j^k + \beta_{ij})H$ - ako se zadaci izvršavaju (ako je $z_i = 1$ i $z_j = 1$) i ako su dodijeljeni istom robotu (ako je $\alpha_i^k = 1$ i $\alpha_j^k = 1$), pazi da se ne izvršavaju istovremeno tj. da nema preklapanja
- $t_j \geq t_i + T_i^k - (5 - z_i - z_j - \alpha_i^k - \alpha_j^k - \beta_{ij})H$ - ako se zadaci izvršavaju (ako je $z_i = 1$ i $z_j = 1$) i ako su dodijeljeni istom robotu (ako je $\alpha_i^k = 1$ i $\alpha_j^k = 1$), pazi da se ne izvršavaju istovremeno tj. da nema preklapanja
- $tr_i^k \cdot z_i \cdot \alpha_i^k \leq t_i \leq td_i^k \cdot z_i \cdot \alpha_i^k \leq H \cdot z_i$ - ako se zadatak izvodi ($z_i = 1$), i ako se izvodi na robotu k ($\alpha_i^k = 1$) treba paziti da se izvodi unutar dozvoljenog vremena. A ako se ne izvodi $t_i = 0$.
- $T_i^k = \frac{x^k - x_i + v \cdot t_i}{v} + \frac{y^k - y_i}{v}$ - ovo treba vrijediti samo ako se izvodi na k

2 Primjer

Neka je zadano da u sustavu postoje dva robota $R = \{1, 2\}$. Neka u sustavu postoji pet zadataka $J = \{1, 2, 3, 4, 5\}$.

Ako se npr. predmet 5 preskače, a prva dva predmeta radi robot 1, a druga dva robot 2, rješenje će biti:

$$z_1 = 1, z_2 = 1, z_3 = 1, z_4 = 1, z_5 = 0$$

$$\alpha_1^1 = 1, \alpha_1^2 = 0$$

$$\alpha_2^1 = 1, \alpha_2^2 = 0$$

$$\alpha_3^1 = 0, \alpha_3^2 = 1$$

$$\alpha_4^1 = 0, \alpha_4^2 = 1$$

$$\alpha_5^1 = 0, \alpha_5^2 = 0$$

Ako se na robotu 1 prvo izvršava zadatak 1 pa zadatak 2, a na robotu prvo predmet 4, pa 3, vrijedit će:

$$\beta_{1,2} = 1, \beta_{2,1} = 0$$

$$\beta_{3,4} = 0, \beta_{4,3} = 1$$

Trebaju također biti određene vrijednosti t_1, t_2, t_3, t_4 koje su neki cijeli brojevi. Vrijednost $t_5 = 0$ jer se predmet ne uzima s trake.

Rješenje koje nije moguće prekršit će barem jedan od napisanih uvjeta tj. ograničenja. Potrebno je napisati optimizacijski algoritam koji će za zadane ulazne vrijednosti, od svih mogućih rješenja koja postoje, odrediti najbolje rješenje.