



DEFINIÇÃO DO TRABALHO

Alberto Francisco Kummer Neto

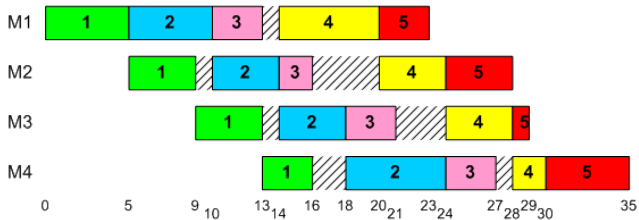
INF05010 - Otimização Combinatória — Maio, 2019

Problemas propostos

1. PFSP (Permutational Flowshop Scheduling Problem)
2. PMSP (Parallel Machines Scheduling Problem)
3. TSP-DL (Traveling Salesman Problem with Draft Limits)

PFSP

Permutational Flowshop Scheduling Problem



(Fonte: [StackOverflow](#))

Conjuntos e parâmetros:

- N tarefas a serem processadas
- M máquinas disponíveis para processamento
- Tempos de processamento $T_{ir} > 0$, para $1 \leq i \leq N, 1 \leq r \leq M$

Restrições:

- Tarefas devem ser processadas em sequência
- Cada tarefa deve ser processada em todas as máquinas
- Processamento sem preempção

Objetivo: Minimizar o *makespan*

Solução: Ordem alocação das tarefas às máquinas

$$\text{Minimiza } C_{\max} \quad (1)$$

Sujeito a:

$$C_{1i} \geq T_{1i} \quad 1 \leq i \leq N \quad (2)$$

$$C_{ri} - C_{r-1,i} \geq T_{ri} \quad 1 \leq i \leq N, 2 \leq r \leq M \quad (3)$$

$$C_{ri} - C_{rk} + PD_{ik} \geq T_{ri} \quad 1 \leq i < k \leq N, 1 \leq r \leq M \quad (4)$$

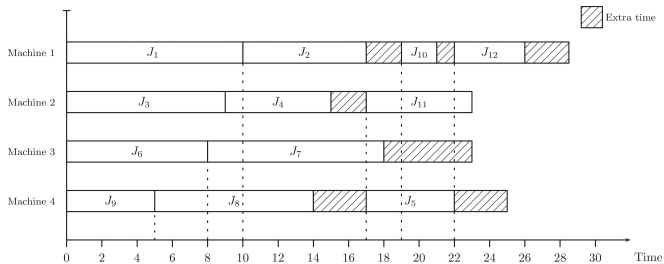
$$C_{ri} - C_{rk} + PD_{ik} \leq P - T_{rk} \quad 1 \leq i < k \leq N, 1 \leq r \leq M \quad (5)$$

$$C_{\max} \geq C_{ri} \quad 1 \leq i \leq N, 1 \leq r \leq M \quad (6)$$

(Fonte: Tseng et al. (2009))

PMSP

Parallel Machines Scheduling Problem



(Fonte: Lalla-Ruiz e Voß (2016))

Conjuntos e parâmetros:

- N tarefas a serem processadas
- M máquinas disponíveis para processamento
- Tempo de preparo S_{ijk} para $1 \leq i \leq N, 1 \leq j \leq N, 1 \leq k \leq M$
- Tempo de processamento P_{jk} para $1 \leq j \leq N, 1 \leq k \leq M$
- Tempo total $G_{ijk} = S_{ijk} + P_{jk}$

Restrições:

- Cada tarefa deve ser processada em alguma máquina
- Processamento sem preempção
- Tempos de preparação para todos os pares de tarefas, para cada máquina
- Tempos de processamento de cada tarefa, em cada máquina

Objetivo: Minimizar o *makespan*

Solução: Ordem alocação das tarefas às máquinas

$$\text{Minimiza } C_{\max} \quad (7)$$

Sujeito a:

$$\sum_{\substack{i=0 \\ i \neq j}}^N \sum_{k=0}^M x_{ijk} = 1 \quad 1 \leq j \leq N \quad (8)$$

$$\sum_{\substack{i=0 \\ i \neq h}}^N x_{ijk} - \sum_{\substack{j=0 \\ j \neq h}}^N x_{hjk} = 0 \quad \forall 1 \leq h \leq N \quad (9)$$

$$C_j \geq C_i + \sum_{k=1}^M G_{ijk} x_{ijk} + V \left(\sum_{k=1}^M x_{ijk} - 1 \right) \quad 0 \leq i \leq N \quad (10)$$

(Continua no próximo slide)

$$C_j \leq C_{\max} \quad 1 \leq j \leq N \quad (11)$$

$$\sum_{j=0}^N x_{0jk} = 1 \quad 1 \leq k \leq M \quad (12)$$

$$C_j \leq 0 \quad 1 \leq j \leq N \quad (13)$$

$$C_0 = 0 \quad (14)$$

$$x_{ijk} \in \{0, 1\} \quad 0 \leq i \leq N, 0 \leq j \leq N, 1 \leq k \leq M \quad (15)$$

(Fonte: Ezugwu (2019))

TSPDL

Traveling Salesman Problem with Draft Limits



(Fonte: Rakke et al. (2012))

Conjuntos e parâmetros:

- Uma embarcação
- Conjunto V de portos, com $|V| - 1$ portos a serem atendidos
- Garagem no porto $1 \in V$
- Cada porto com uma demanda $d_j \geq 0$ e limite de profundidade $l_j \geq 0$
- Garagem com oferta suficientemente grande
- Custo c_{ij} da viagem entre $(i, j) \in V \times V$

Restrições:

- A embarcação não pode fazer múltiplos retornos à garagem
- As demandas dos portos devem ser atendidas

- A embarcação não pode atracar caso sua carga extraple o limite de profundidade do porto

Objetivo: Rota de menor custo

Solução: Ordem de visitação dos portos

$$\text{Minimiza } c_{ij}x_{ij} \quad (16)$$

Sujeito a:

$$\sum_{i \in V} x_{ij} = 1 \quad \forall j \in V \quad (17)$$

$$\sum_{j \in V} x_{ij} = 1 \quad \forall i \in V \quad (18)$$

$$\sum_{i \in V} y_{ij} - \sum_{i \in V} y_{ji} = d_j \quad \forall j \in V \setminus \{1\} \quad (19)$$

(Continua no próximo slide)

$$\sum_{j \in V} y_{1j} = \sum_{i \in V} d_i \quad (20)$$

$$\sum_{i \in V} y_{i1} = 0 \quad (21)$$

$$0 \leq y_{ij} \leq l_j x_{ij} \quad \forall (i, j) \in V \times V \quad (22)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in V \times V \quad (23)$$

(Fonte: Rakke et al. (2012))