U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

SAAD Practical Assignment

Group 1A

José Pedro Evans de Carvalho Nobre João – 202108818

Ricardo Jorge Correia Pinto - 201202477

Vitor Souza Piña - 202400084

2024/2025

Agenda

- Introduction
- Instances
- Open Shop problem & Results
- Job Shop problem & Results
- Conclusions

Introduction

Operational research

Combinatorial optimization problems

Machine scheduling problems

NP-hard

Processing a set of **jobs** on a set of **machines**, with the ultimate goal of optimizing the **completion time** of all jobs

Introduction

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Shop scheduling problems

- Set M of **machines**
- Set J of **jobs**, where each job j comprises a sequence of n_j **operations or tasks** performed on specific machines
  - processing time p_{ij} on a specific machine m_{ij}
  - processed on only one machine at a time
  - each machine can execute only one operation at any given moment
- **Ordering** of operation processing

Job Shop	Open Shop
• precedence constraint: processing in a predetermined sequence on specific machine	• order in which operations are carried out is not fixed or predetermined

Introduction

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Constraint satisfaction problems

Mixed Integer Programming (MIP)

- linear programming formulations
- integer or binary variables
- optimization techniques: branch-and-bound

Constraint Programming (CP)

- logical and global constraints
- adept at addressing precedence relations or temporal constraints
- optimization techniques: domain reduction and constraint propagation

Instances

U.PORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Job Shop	Open Shop
Taillard's benchmark instances	
j x *m*	*j* x *m*
5 x 15 (10 instances)	4 x 4 (10 instances)
20 x 15 (10 instances)	5 x 5 (10 instances)
20 x 20 (10 instances)	7 x 7 (10 instances)
30 x 15 (10 instances)	10 x 10 (10 instances)
30 x 20 (10 instances)	15 x 15 (10 instances)
50 x 15 (10 instances)	20 x 20
50 x 20	
100 x 20	

Time limits of 100 and 200 seconds

Open Shop problem | MIP

- Similar formulation as the Job-shop problem, but tasks don not need to be executed in order. **One of the constraints is relaxed**.
- Adapted the Job-shop model by removing the precedence constraint and introducing constraints 7 and 8.

$$
\begin{aligned}
\min\ & C_{\max} && \\
\text{s.t.}\ & x_{i_2 j} \geq x_{i_1 j} + p_{i_1 j} - \theta \cdot (1 - z_{ijk}), && \forall i_1, i_2 \in M,\ j \in J,\ i_1 \neq i_2, && (7)\\
& x_{i_1 j} \geq x_{i_2 j} + p_{i_2 j} - \theta \cdot z_{ijk}, && \forall i_1, i_2 \in M,\ j \in J,\ i_1 \neq i_2, && (8)\\
& x_{ij} \geq x_{ik} + p_{ik} - \theta \cdot z_{ijk}, && \forall j, k \in J,\ j < k,\ i \in M, && (9)\\
& x_{ik} \geq x_{ij} + p_{ij} - \theta \cdot (1 - z_{ijk}), && \forall j, k \in J,\ j < k,\ i \in M, && (10)\\
& C_{\max} \geq x_{\sigma^j_m, j} + p_{\sigma^j_m, j}, && \forall j \in J, && (11)\\
& z_{ijk} \in [0, 1], && \forall j, k \in J,\ i \in M, && (12)\\
& x_{ij} \geq 0, && \forall j \in J,\ i \in M. && (13)
\end{aligned}
$$

- J: A finite set of n jobs.
- M: A finite set of m machines.
- σ_{hj}: The h-th operation of job j, representing the sequence of machines through which j must be processed.
- σ_{mj}: The last operation of job j.
- p_{ij}: A non-negative integer representing the processing time of job j on machine i.
- x_{ij}: An integer representing the start time of job j on machine i.
- z_{ijk}: A binary variable equal to 1 if job j precedes job k on machine i, and 0 otherwise.

Open Shop problem | CP

- Relaxation of the strict sequencing constraint

$$
\begin{aligned}
&\min C_{\max} \\
&\text{s.t.} \\
&\qquad \text{start}_{t_1} = \text{end}_{t_2}, \quad \forall t \in \mathcal{T}, t_1 \neq t_2 && (18) \\
&\qquad \text{NoOverlap}(S_m), \quad \forall m \in \mathcal{M}. && (19) \\
&\qquad \text{NoOverlap}(S_j), \quad \forall j \in \mathcal{J}. && (20) \\
&\qquad \text{makespan} \geq \text{end}_t, \quad \forall t \in \mathcal{T}. && (21)
\end{aligned}
$$

where:

- start-end dependency: ensures the start time of a dependent task aligns with the end time of its predecessor
- machine capacity constraint (*NoOverlap*): prevents tasks from overlapping on the same machine

Results

	Open Shop CP	Open Shop MIP
100 s & 200 s	• CP demonstrates a markedly **superior** performance compared to the MIP for instances where *m* and *j* > 7. • CP achieved an optimality rate of **100%.** • CP model was consistently **faster** in achieving optimality 0.30 s vs 1.70 s (MIP) – 100 s 0.26 s vs 3.85 s (MIP) – 200 s	• MIP achieved **100%** optimality for 4x4 and 5x5 instances and only 10% (100s) / 20% (200s) for 7x7 instances. • **Failed** to find any solutions for **larger** problems. • Maximum **gap** of 56.6% (100s) and 75.1% (200s). • For **larger** instances (15x15 and 20x20), the solver could not identify any feasible solution

Table 1 – CP optimality percentage for various instance sets sizes in the JS problem with 100s time limit.

Job Shop problem | MIP

- The **MIP** model utilizes the Big θ method;
- Based on the disjunctive model, proposed by Wen-Yang [1]

$$\min C_{\max}$$

$$\text{s.t.}$$

$$
\begin{aligned}
& x_{\sigma_h^j, j} \geq x_{\sigma_{h-1}^j, j} + p_{\sigma_{h-1}^j, j}, && \forall j \in J, h = 2, \ldots, m && (1) \\
& x_{ij} \geq x_{ik} + p_{ik} - \theta \cdot z_{ijk}, && \forall j, k \in J, j < k, i \in M && (2) \\
& x_{ik} \geq x_{ij} + p_{ij} - \theta \cdot (1 - z_{ijk}), && \forall j, k \in J, j < k, i \in M && (3) \\
& C_{\max} \geq x_{\sigma_m^j, j} + p_{\sigma_m^j, j}, && \forall j \in J && (4) \\
& z_{ijk} \in [0, 1], && \forall j, k \in J, i \in M && (5) \\
& x_{ij} \geq 0, && \forall j \in J, i \in M && (6)
\end{aligned}
$$

where:

- J: A finite set of n jobs.
- M: A finite set of m machines.
- σ_{hj}: The h-th operation of job j, representing the sequence of machines through which j must be processed.
- σ_{mj}: The last operation of job j.
- p_{ij}: A non-negative integer representing the processing time of job j on machine i.
- x_{ij}: An integer representing the start time of job j on machine i.
- z_{ijk}: A binary variable equal to 1 if job j precedes job k on machine i, and 0 otherwise.

[1] Wen-Yang Ku and J. Christopher Beck, Mixed Integer Programming Models for JobShop Scheduling: A Computational Analysis, Computers & Operations Research,vol. 73, Sept. 2016

Job Shop problem | CP

- The **CP** model uses global constraints to simplify the model
- Elimination of the Big θ method

$$\min C_{\max}$$
$$\text{s.t.}$$
$$\text{start}_{t'} = \text{end}_t, \quad \forall t \in \mathcal{T}, t \neq t' \tag{14}$$
$$\text{NoOverlap}(S_m), \quad \forall m \in \mathcal{M}. \tag{15}$$
$$\text{Sequencevar}(\sigma_{ij}) \tag{16}$$
$$\text{makespan} \geq \text{end}_t, \quad \forall t \in \mathcal{T}. \tag{17}$$

where:

- start-end dependency: ensures the start time of a dependent task aligns with the end time of its predecessor
- machine capacity constraint (*NoOverlap*): prevents tasks from overlapping on the same machine
- task sequencing (*SequenceVar*): maintains the proper sequence of tasks on machine

Results

Job Shop CP vs MIP

100 s

- **CP** significantly outperformed the MIP
- **Optimality** (time limit): 23.8% CP vs 0.0% MIP
- Time limit reached: **CP** has **smaller** gaps (maximum 12.1%) vs MIP (maximum 98.5%; minimum 11.9%)
- CP maintained a **consistent gap** across all problem sizes; MIP's gap **increased proportionally** with problem size
- CP demonstrated **superior** performance in minimizing makespan

Instance Set	Percentage Optimal (%)
15x15	50
20x15	20
20x20	0
30x15	10
30x20	0
50x15	100
50x20	10
100x20	0

Table 1 – CP optimality percentage for various instance sets sizes in the JS problem with 100s time limit.

Results

UPORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Job Shop CP vs MIP

200 s

- **MIP** still failed to achieve optimality for any instance.
- **MIP's** gap values showed little improvement, particularly for larger problems
- **CP** showed a modest improvement, solving 33.8% of instances optimally
- **CP** consistently delivered **smaller** gaps, even for the largest instances

Instance Set	Percentage Optimal (%)
15x15	80
20x15	20
20x20	0
30x15	10
30x20	0
50x15	100
50x20	30
100x20	30

Table 2 – CP optimality percentage for various instance sets sizes in the JS problem with 200s time limit.

Conclusions

- CP consistently **outperformed** MIP in both proving optimality and achieving higher-quality solutions, regardless of the instance set size in the **Job Shop** problem.
- For the **Open Shop** problem, due to the absence of precedence constraints, both models performed comparably well in finding optimality for smaller problem instances.

However, as the problem size increased, CP demonstrated superior performance

CP's robust handling of combinatorial complexity makes it a preferred method for tackling real-world scheduling challenges

Thank you

Suggestions and comments are welcome