An introduction to scheduling: The case of astronomical observations

Maher Mallem[1], Frédéric Vivien[1], Anne Benoit[2]

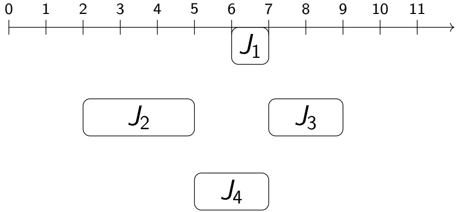
[1]Inria, CNRS, ENS de Lyon, Université Claude Bernard Lyon 1, LIP, UMR 5668, 69342, Lyon cedex 07, France,

[2]ENS de Lyon, CNRS, Inria, Université Claude Bernard Lyon 1, LIP, UMR 5668, 69342, Lyon cedex 07, France

20 November 2025

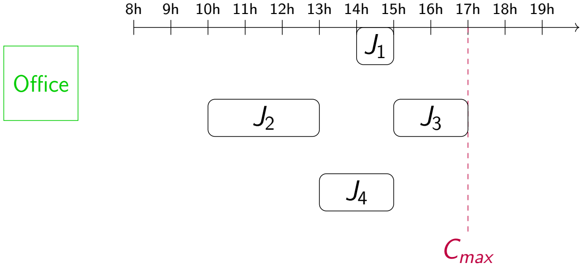
Outline

Outline

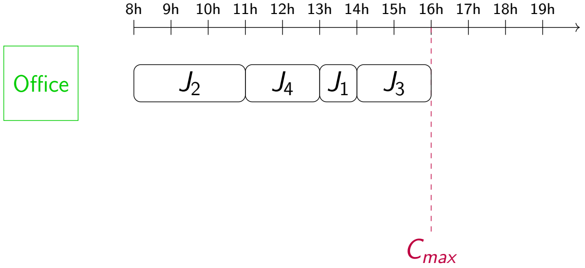


$\mathcal{J} = \{J_1, \ldots, J_n\}$ set of n jobs of processing time $p_1, \ldots, p_n$

Introduction to scheduling



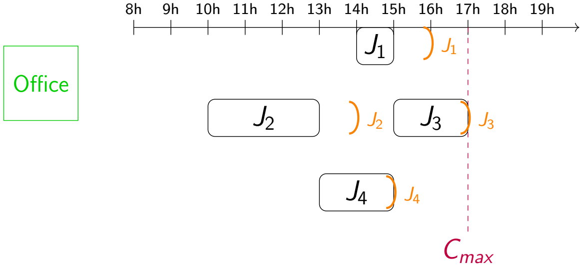
$1||C_{max}$



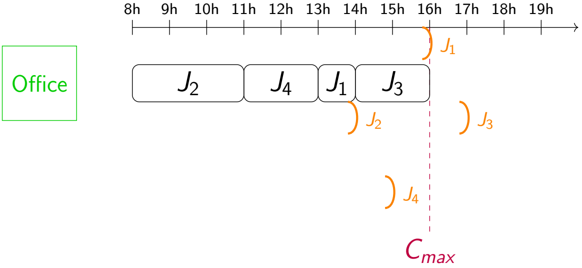
$1||C_{max}$

Any job order. $\mathcal{O}(n)$ time.

Introduction to scheduling



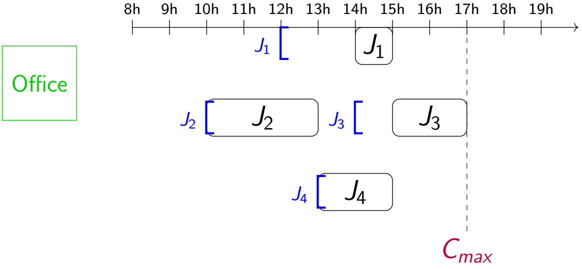
$1|d_j|C_{max}$



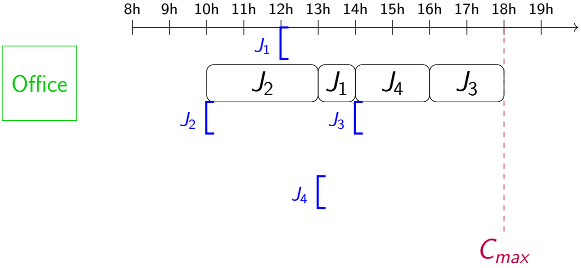
$1|d_j|C_{max}$

"Earliest Deadline First" (EDF) job order. $\mathcal{O}(n \cdot \log(n))$ time.

Introduction to scheduling

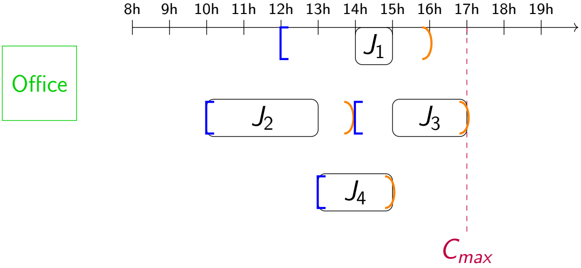


$$1|r_j|C_{max}$$



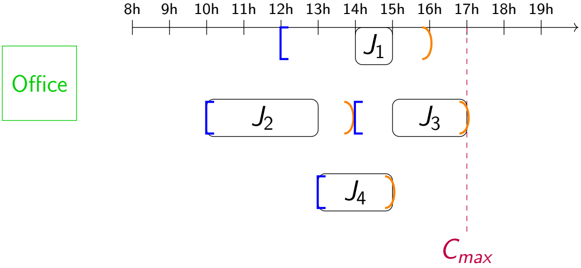
$1|r_j|C_{max}$

"Earliest Start Time" (EST) job order. $\mathcal{O}(n \cdot \log(n))$ time.


8h
9h
10h
11h
12h
13h
14h
15h
16h
17h
18h
19h
Office
J_1
J_2
J_3
J_4
C_{max}

$$1|r_j, d_j|C_{max}$$

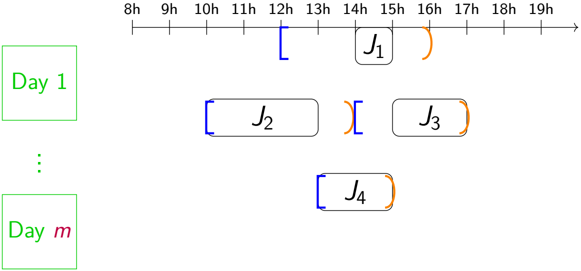
Introduction to scheduling



$1|r_j, d_j|C_{max}$

Strongly NP-complete [Lenstra et al., 1977].

Introduction to scheduling



$$P|r_j, d_j|min(m)$$

Strongly NP-complete [Lenstra et al., 1977].

Outline

Why?

Fixed daily start times

Why?

- (Maybe) easier to solve.

Fixed daily start times

Why?

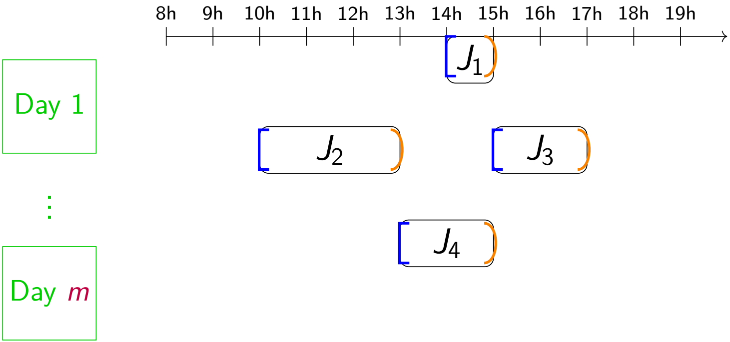
- (Maybe) easier to solve.
- Upper bound.

Fixed daily start times

Why?

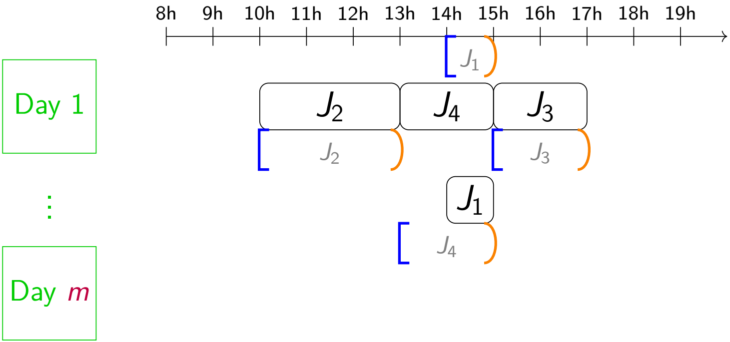
- (Maybe) easier to solve.
- Upper bound.
- Set around altitude peaks.
  $\longrightarrow$ Best observation quality.

Back to meeting planning



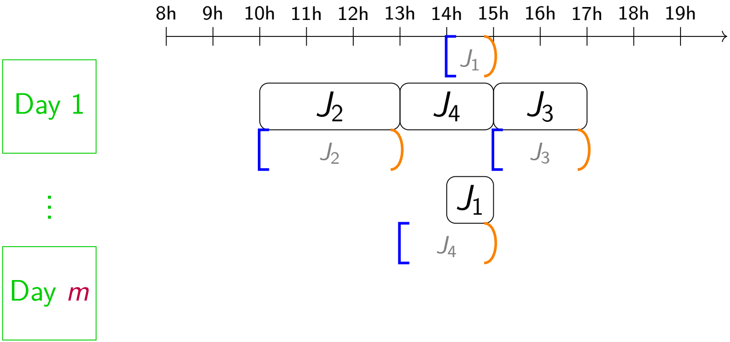
$$P|r_j, d_j, d_j - r_j = p_j|min(m)$$

Back to meeting planning



$$P|r_j, d_j, d_j - r_j = p_j|min(m)$$

Back to meeting planning



$$P|r_j, d_j, d_j - r_j = p_j|min(m)$$

Interval graph coloring: $\mathcal{O}(n \cdot \log(n))$ time.

Outline

No daily start time restriction.

- Back to the NP-complete general problem.

No daily start time restriction.

- Back to the NP-complete general problem.
- No (2-ε)-approximation algorithm.

More flexible settings

No daily start time restriction.

- Back to the NP-complete general problem.
- No (2-ε)-approximation algorithm.

Fixed arbitrary start times (no more periodicity).

- a.k.a. Interval Scheduling.

More flexible settings

No daily start time restriction.

- Back to the NP-complete general problem.
- No $(2\text{-}\varepsilon)$-approximation algorithm.

Fixed arbitrary start times (no more periodicity).

- a.k.a. Interval Scheduling.
- NP-complete even with at most 3 possible start times per job.

No daily start time restriction.

- Back to the NP-complete general problem.
- No $(2\text{-}\varepsilon)$-approximation algorithm.

Fixed arbitrary start times (no more periodicity).

- a.k.a. Interval Scheduling.
- NP-complete even with at most 3 possible start times per job.

Maximize the number of scheduled jobs.

- a.k.a. Interval Selection.

More flexible settings

No daily start time restriction.

- Back to the NP-complete general problem.
- No (2-ε)-approximation algorithm.

Fixed arbitrary start times (no more periodicity).

- a.k.a. Interval Scheduling.
- NP-complete even with at most 3 possible start times per job.

Maximize the number of scheduled jobs.

- a.k.a. Interval Selection.
- 1/2-approximation: "Earliest Completion Time" (ECT).

More flexible settings

No daily start time restriction.

- Back to the NP-complete general problem.
- No (2-ε)-approximation algorithm.

Fixed arbitrary start times (no more periodicity).

- a.k.a. Interval Scheduling.
- NP-complete even with at most 3 possible start times per job.

Maximize the number of scheduled jobs.

- a.k.a. Interval Selection.
- 1/2-approximation: "Earliest Completion Time" (ECT).
  $\rightarrow$ $\mathcal{O}(\log(n))$-approximation algorithm for observation scheduling.

Personalize day choices.

- a.k.a. Parallel scheduling with machine subsets $\mathcal{M}_j \subseteq \{1, \ldots, m\}$.

Personalize day choices.

- a.k.a. Parallel scheduling with machine subsets $\mathcal{M}_j \subseteq \{1, \ldots, m\}$.
- No $o(\log(n))$-approximation algorithm.

Personalize day choices.

- a.k.a. Parallel scheduling with machine subsets $\mathcal{M}_j \subseteq \{1, \ldots, m\}$.
- No $o(\log(n))$-approximation algorithm.

Parameterized complexity.

- Parameters: m number of days, σ slack.

More flexible settings (cont.)

Personalize day choices.

- a.k.a. Parallel scheduling with machine subsets $\mathcal{M}_j \subseteq \{1, \ldots, m\}$.
- No $o(\log(n))$-approximation algorithm.

Parameterized complexity.

- Parameters: m number of days, σ slack.
- Fixed-parameter algorithm parameterized by $(m + \sigma)$.

More flexible settings (cont.)

Personalize day choices.

- a.k.a. Parallel scheduling with machine subsets $\mathcal{M}_j \subseteq \{1, \ldots, m\}$.
- No $o(\log(n))$-approximation algorithm.

Parameterized complexity.

- Parameters: m number of days, σ slack.
- Fixed-parameter algorithm parameterized by $(m + \sigma)$. $\mathcal{O}([(m\sigma^2)^{3m-1} \cdot 2^{3m} \cdot m^{m\sigma} \cdot m\sigma] \cdot n + n \cdot log(n))$ time.

More flexible settings (cont.)

Personalize day choices.

- a.k.a. Parallel scheduling with machine subsets $\mathcal{M}_j \subseteq \{1, \ldots, m\}$.
- No $o(\log(n))$-approximation algorithm.

Parameterized complexity.

- Parameters: m number of days, σ slack.
- Fixed-parameter algorithm parameterized by $(m + \sigma)$. $\mathcal{O}([(m\sigma^2)^{3m-1} \cdot 2^{3m} \cdot m^{m\sigma} \cdot m\sigma] \cdot n + n \cdot log(n))$ time.

Open questions

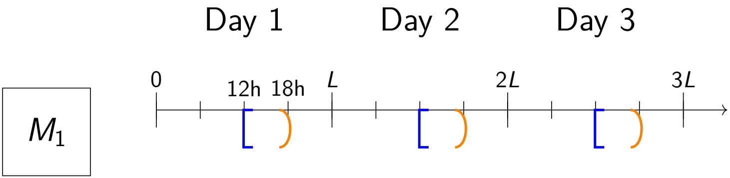
- Constant factor approximation algorithm?
- More efficient fixed-parameter algorithms?

Outline

Postdoc: daily time windows

LIP, ENS de Lyon Nov. 2024 - Apr. 2026

- *Topic:* Scheduling radio-astronomical observations.
  - Joint work with Frédéric Vivien and Anne Benoit.
  - Partnership with the ECLAT laboratory (HPC & AI for astronomy & astrophysics).
  - Access to NenuFAR radio-telescope usage data.



References I

Lenstra, J., Rinnooy Kan, A., and Brucker, P. (1977).
Complexity of machine scheduling problems.
Ann. of Discrete Math., 1:343–362.