1. Single machine problems

- Exact algorithms (SPT, SRPT, EDD)
- Approximation algorithms
- Branch and Bound

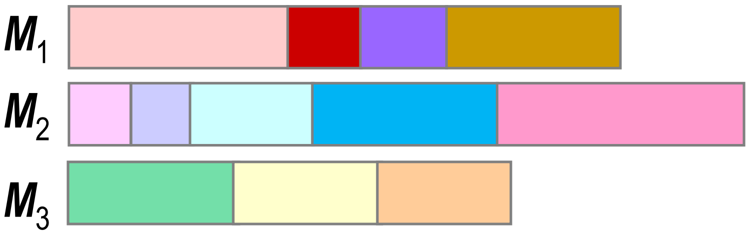
2. Parallel machine problems

- Exact algorithms $P|\ pmtn\ |\ C_{\max}$
- Approximation algorithms $P||\ C_{\max}$
- Branch and Bound

3. Shop problems

- Exact algorithms
- Approximation algorithms
- Branch and Bound

Parallel Machines: Minimising C_{max}



<u>Lower Bounds</u>:

Job-based bound

$$C_{\max} \geq p, \quad \text{where } p = \max\{p_1, p_2, \ldots, p_n\}$$

(we cannot complete all jobs earlier than we complete one job)

Machine-based bound

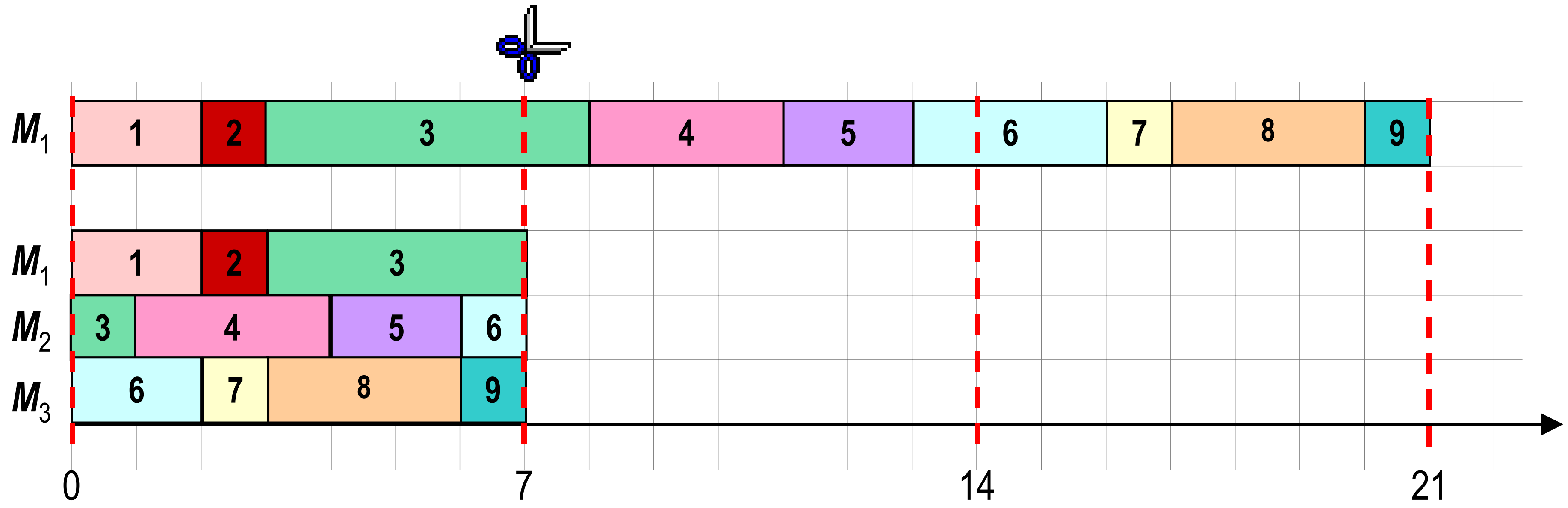
$$C_{\max} \geq \sum_{j=1}^{n} p_j / m$$

(*makespan cannot be smaller than the average machine load*)

$$C_{\max} \geq \max\left\{p, \sum_{j=1}^{n} p_j / m\right\}$$

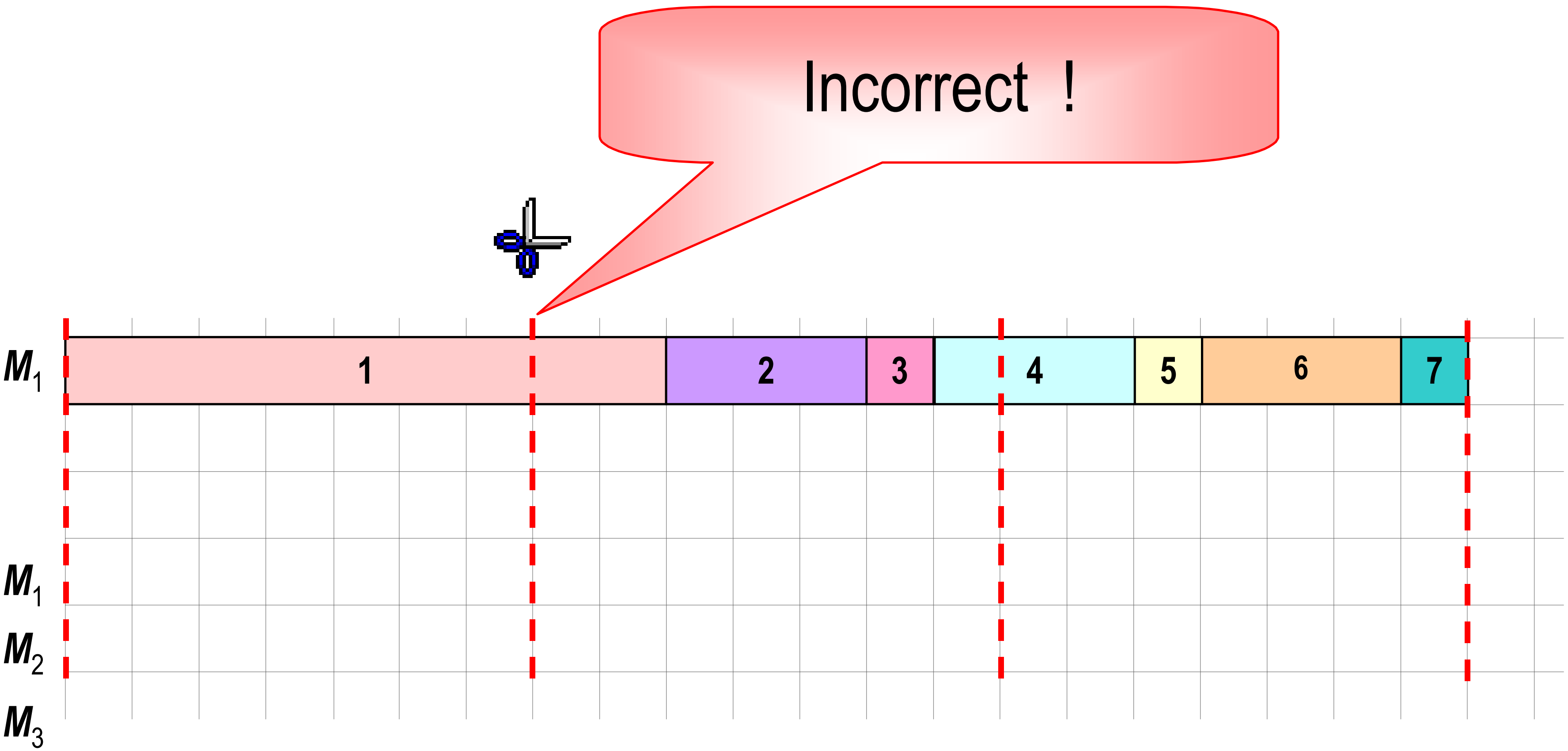
$P \mid Pmtn \mid C_{max}$

1. Calculate the optimal makespan value $$C_{\max}^{OPT} = \max\left\{p, \sum_{j=1}^{n} p_j / m\right\}$$
2. Construct a single-machine nonpreemptive schedule
   (assign ***n*** jobs to a single machine in an arbitrary order starting with the longest job)
3. Cut this single-machine schedule into ***m*** parts of length $\boxed{C_{\max}^{OPT}}$



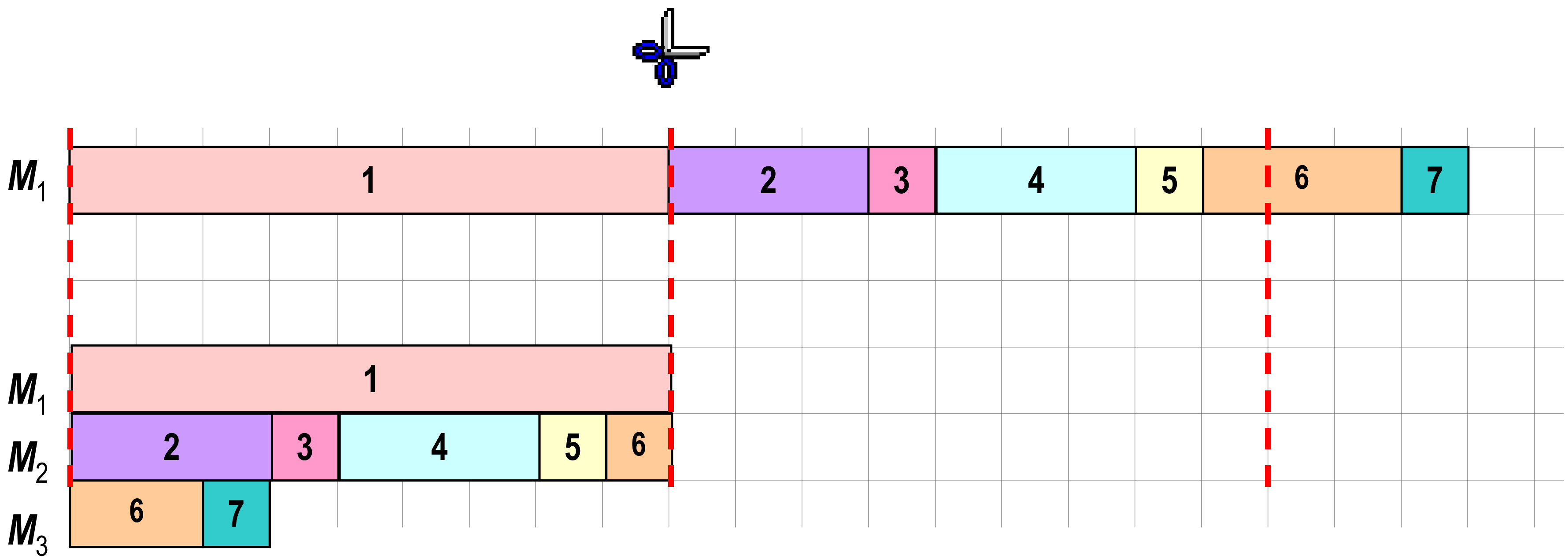
$P \mid Pmtn \mid C_{max}$

$$C^*_{\max} = \max\left\{p_j, \sum_{j=1}^{n} p_j / m\right\}$$



$P \mid Pmtn \mid C_{max}$

$$C^*_{\max} = \max\left\{p_j, \sum_{j=1}^{n} p_j / m\right\}$$



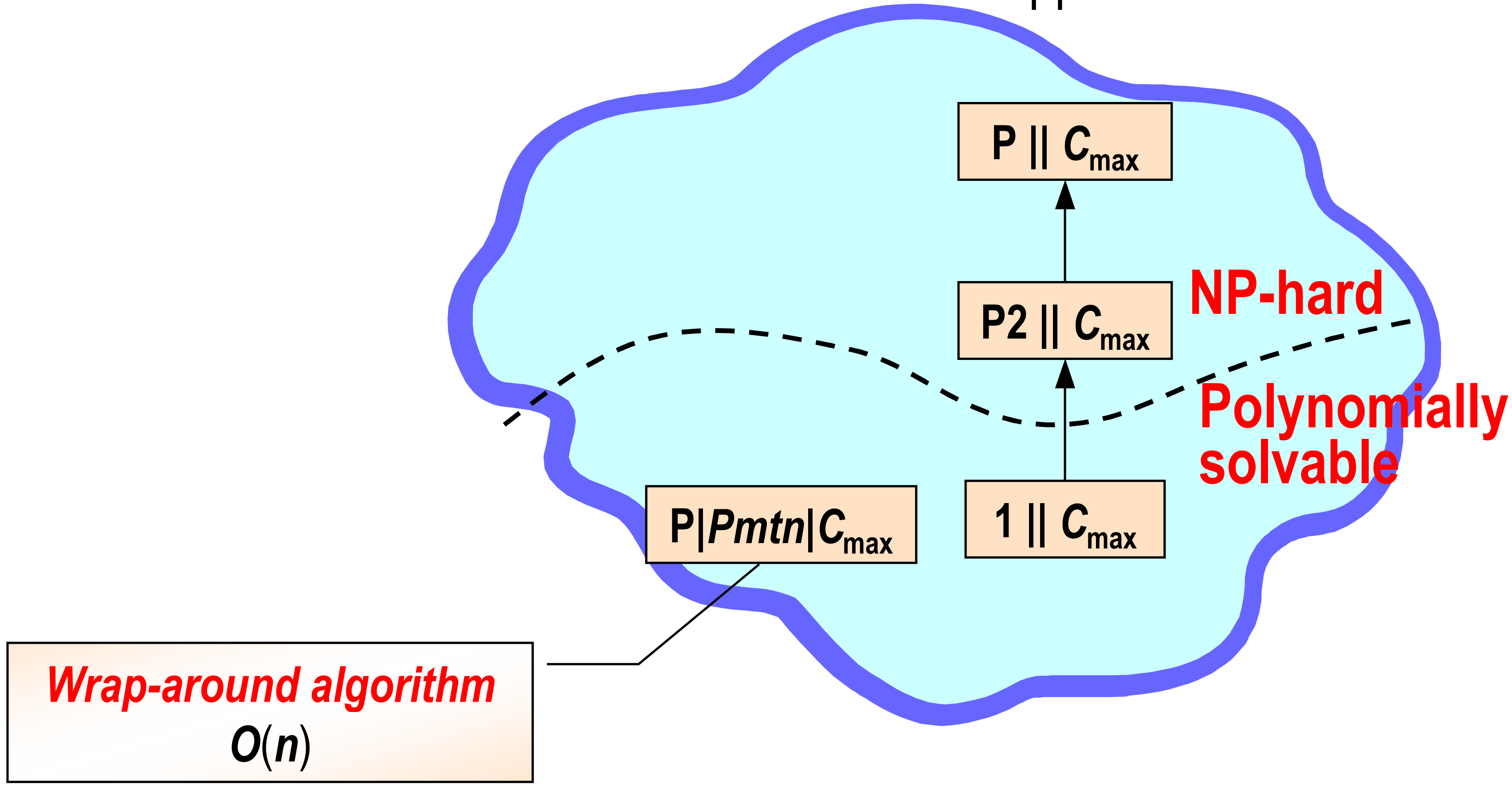
Complexity

Algorithms:

-B&B

-Heuristics

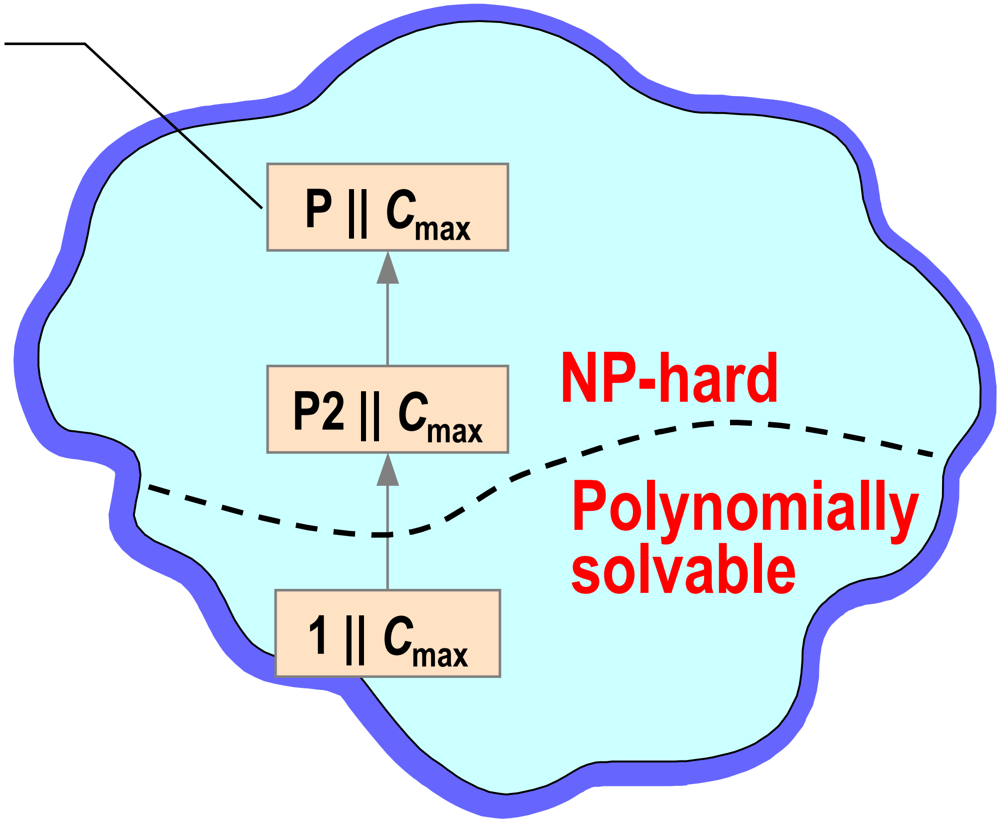
-Approximations



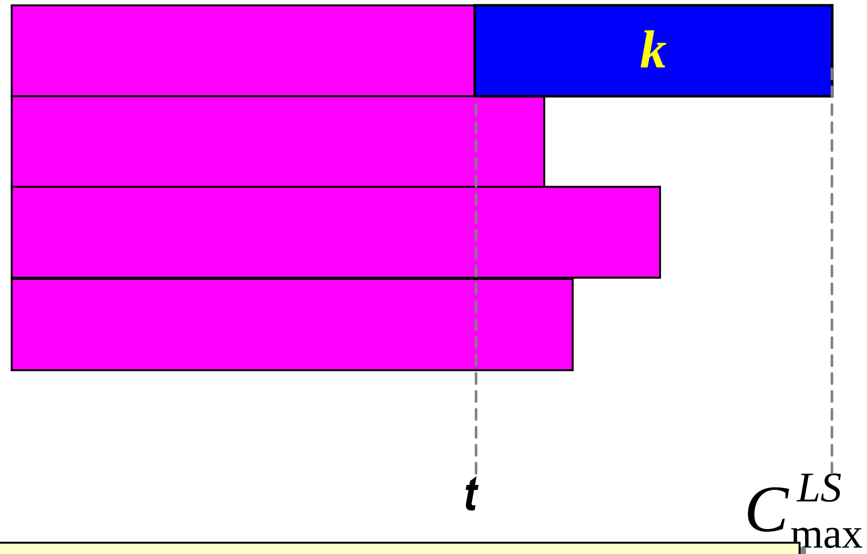
Minimising C_{max}

List Scheduling (2-approximation):

- place jobs into a list
- schedule the first available job on the list of unscheduled jobs whenever a machine becomes idle



For $P \| C_{\max}$, $\dfrac{C_{\max}^{LS}}{C_{\max}^{OPT}} \leq 2$

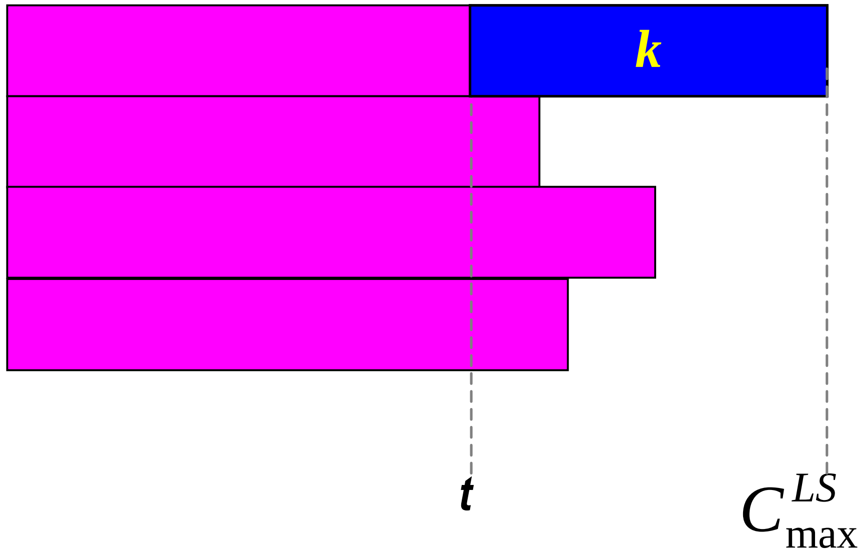


List Scheduling

- historically the first **optimisation** algorithm that was subject to worst-case analysis

- Due to Graham (1965)

For $P \| C_{\max}$, $\quad \dfrac{C_{\max}^{LS}}{C_{\max}^{OPT}} \le 2$



- ***k*** is the last job
- ***t*** is its start time

$$C_{\max}^{LS} = t + p_k \le \frac{1}{m}\sum_{j=1}^{n} p_j + p_k$$

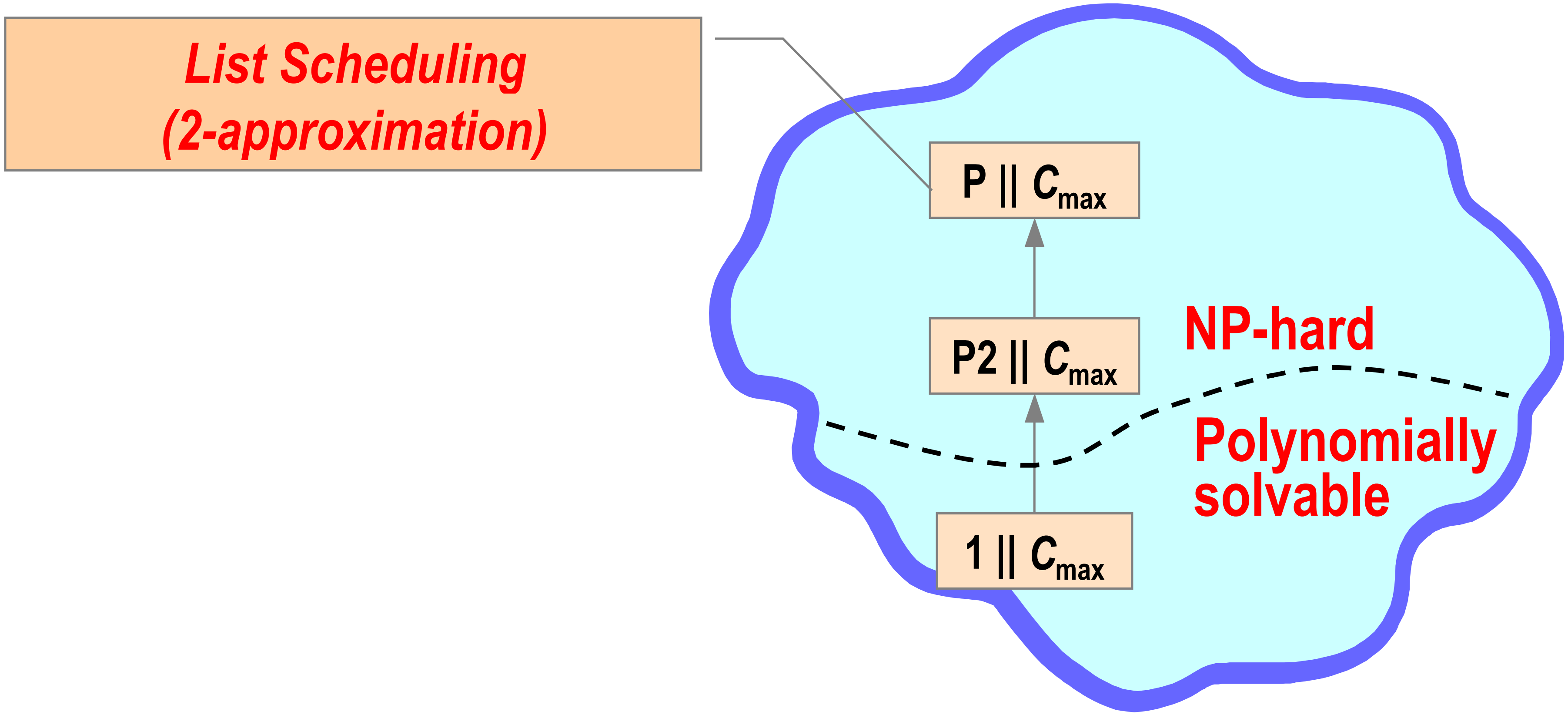
$$\frac{1}{m}\sum_{j=1}^{n} p_j \le C_{\max}^{OPT},$$

$$\boldsymbol{p_k \le p \le C_{\max}^{OPT}}$$

$\Rightarrow$

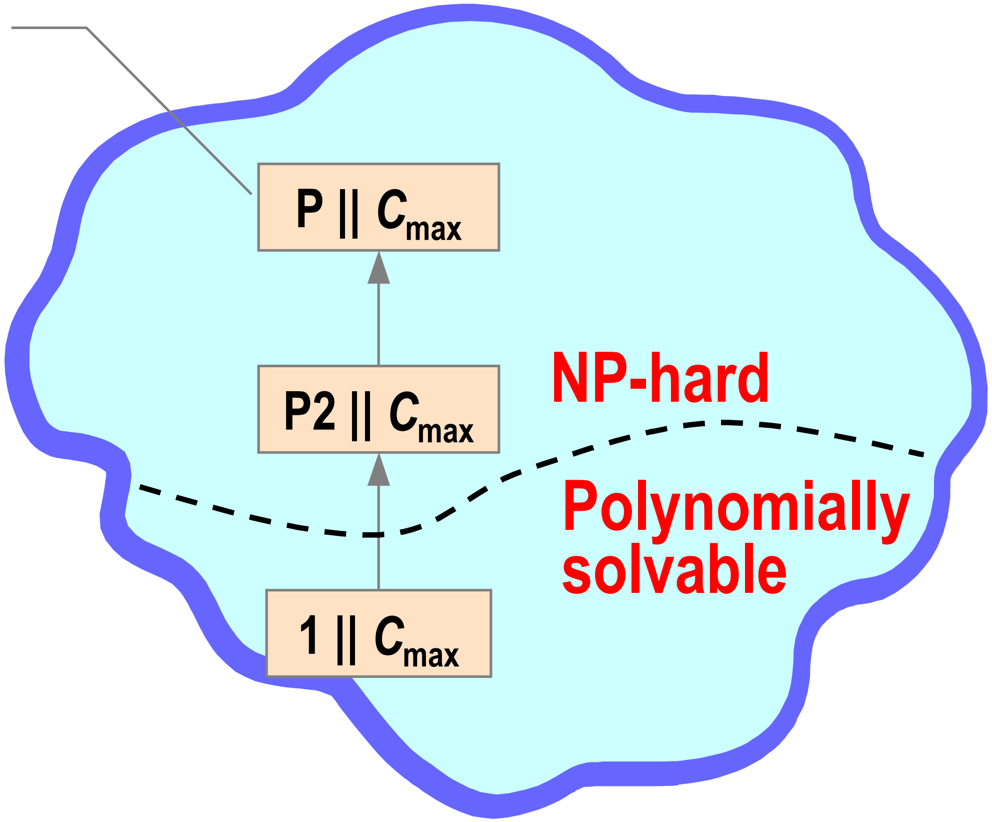
$$\boxed{C_{\max}^{LS} \le 2C_{\max}^{OPT}}$$

Minimising C_{max}

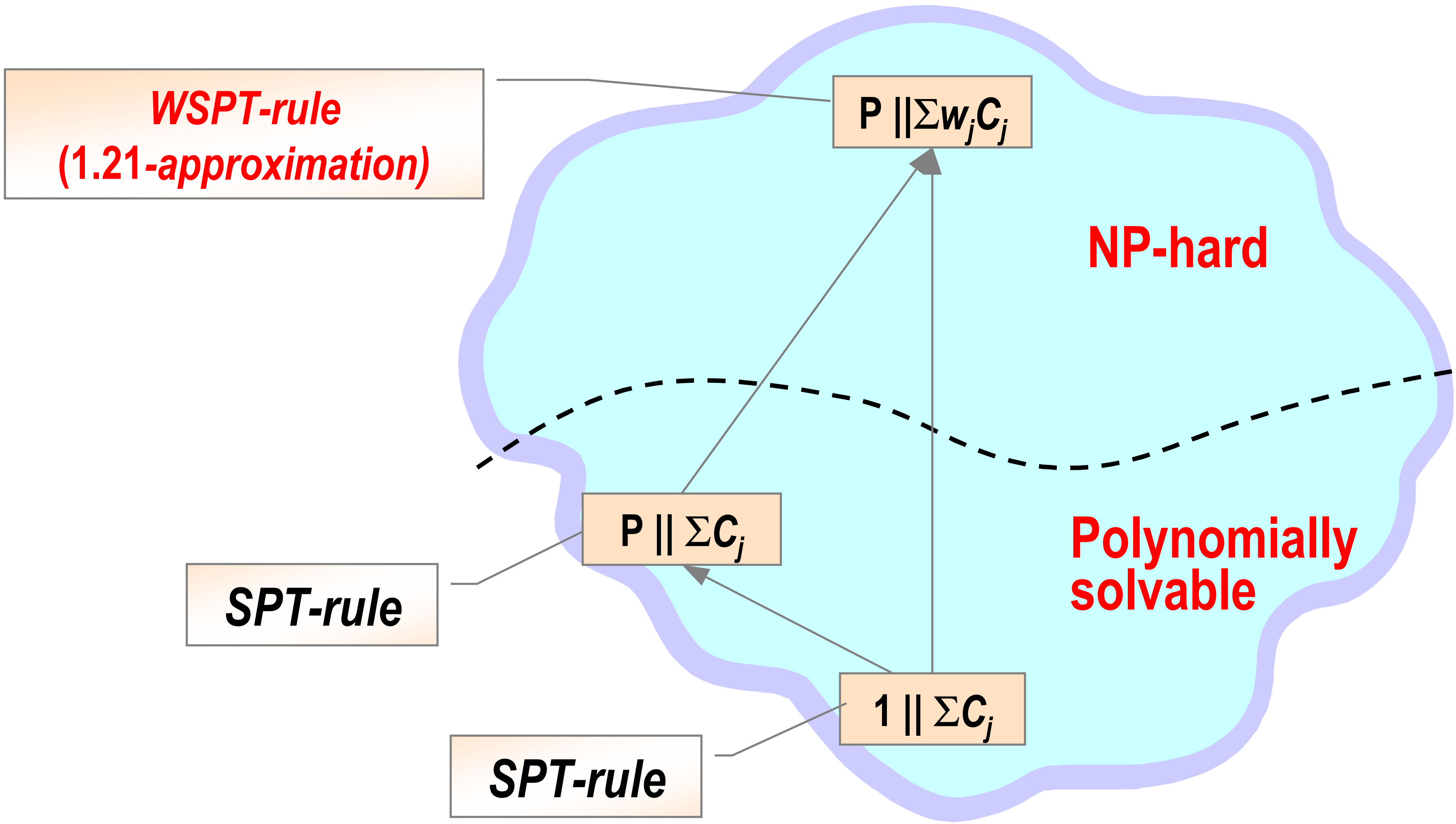


Minimising C_{max}

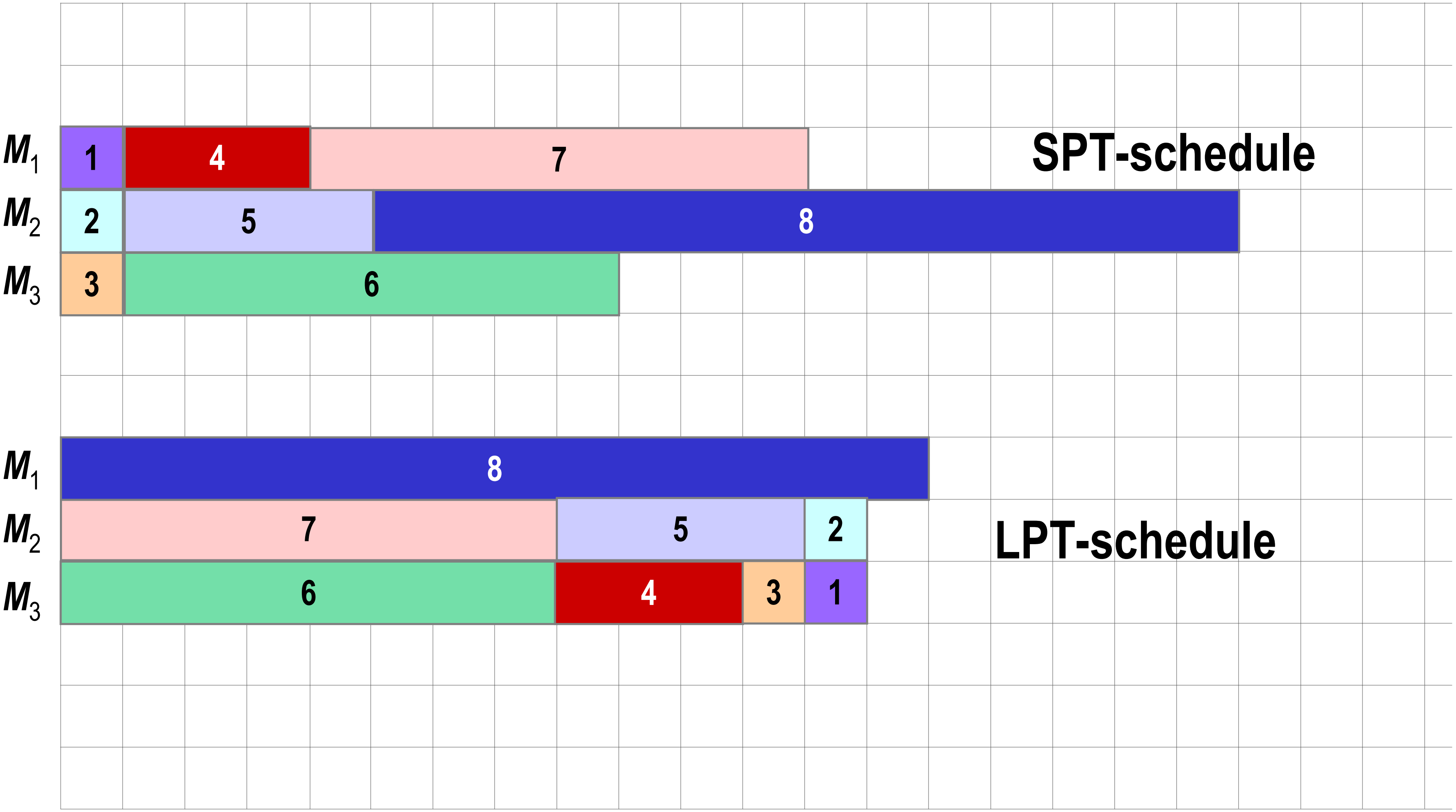
LPT-rule (4/3-approximation):
(Longest Processing Time)
schedule jobs in order of nonincreasing processing times



Total processing time



SPT vs LPT


M_1
M_2
M_3
1
4
7
2
5
8
3
6
SPT-schedule
M_1
M_2
M_3
8
7
5
2
6
4
3
1
LPT-schedule

List of results

Problem	Rule
$P\|\|C_{max}$	LPT (1.33-approximation)
$P\|Pmtn\|C_{max}$	Wrap-around (exact)
$P\|\|\Sigma C_j$	SPT (exact)
$P\|\|\Sigma w_j C_j$	WSPT (1.21-approximation)