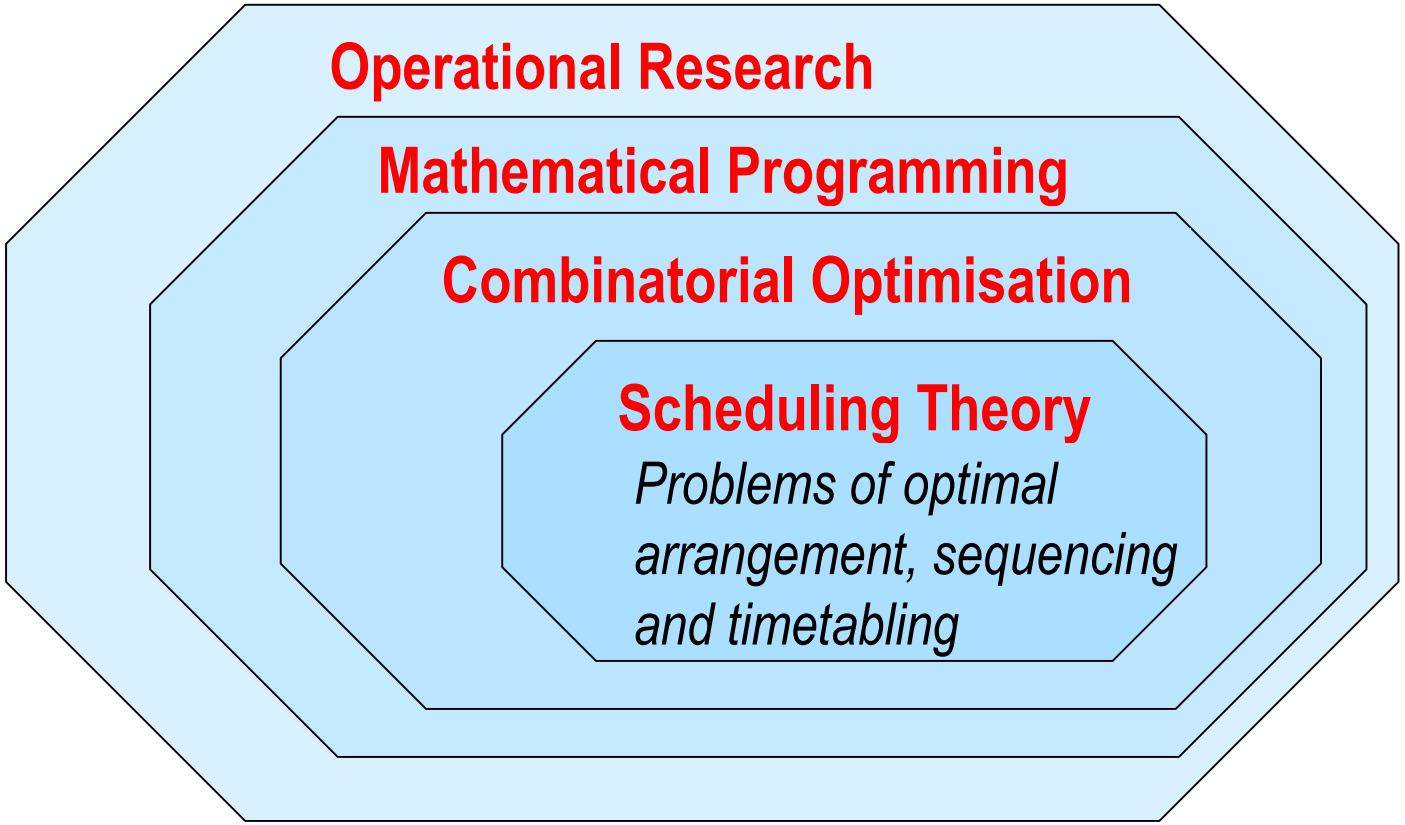
Scheduling:
Models and Algorithms

Natasha Shakhlevich

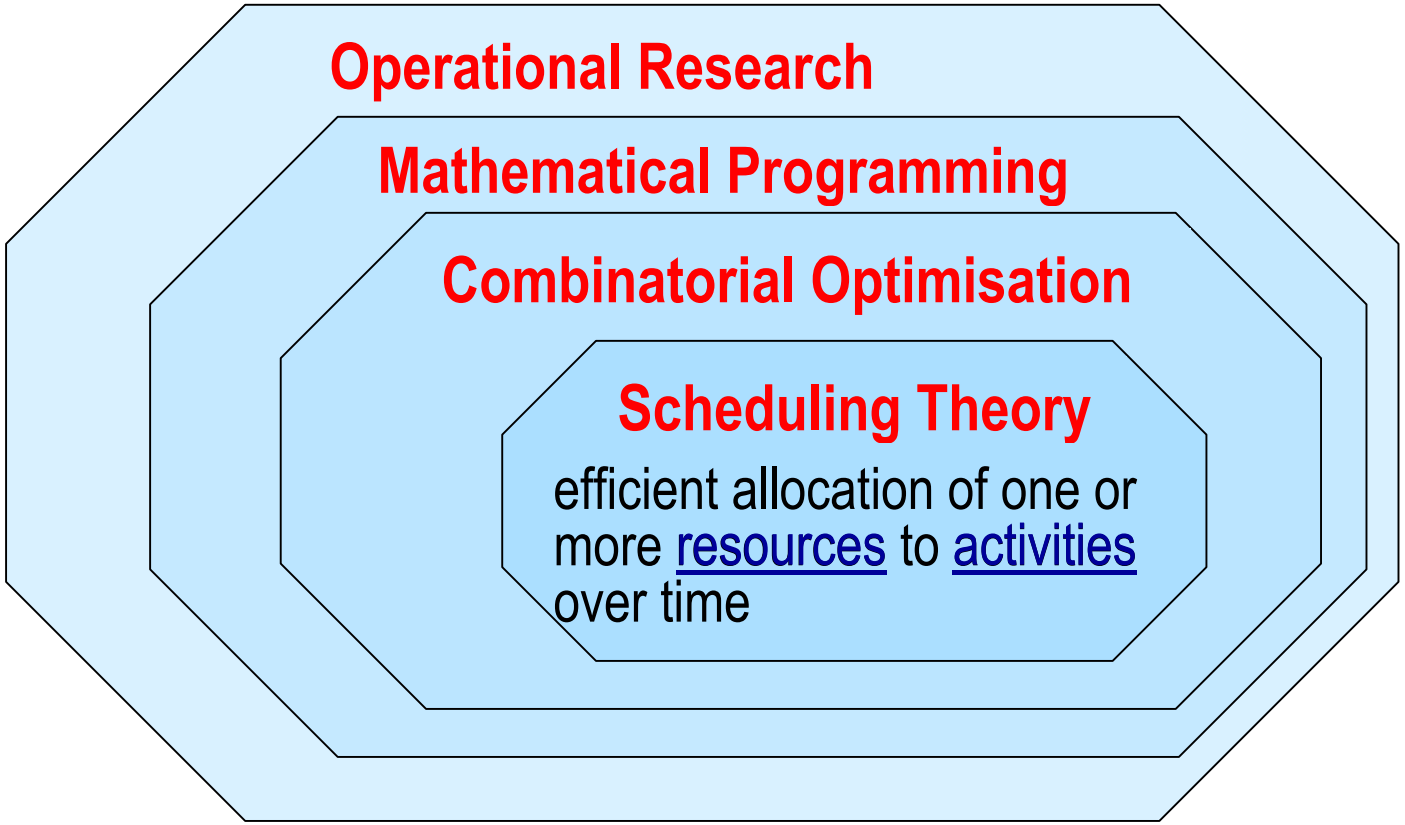
Consultation hour	Friday 3-4 pm
School of Computing	9.10o

(1)	Introduction
(1)	Classification of scheduling models
(2)	Basic scheduling algorithms for single machine problems
(2)	Basic scheduling algorithms for single machine problems
(3)	Computational complexity
(3)	Computational complexity
(4)	Exercises
(4)	Single machine problems: complexity and approximation
(5)	Parallel machine models
(5)	Flow shop
(6)	Flow shop
(6)	Open shop
(7)	-
(7)	Job shop
(8)	Scheduling techniques: dispatching rules, composite rules
(8)	Scheduling techniques: branch and bound algorithms, beam search
(9)	Scheduling techniques: the shifted bottleneck heuristic
(9)	Local search methods
(10)	Local search methods
(10)	Genetic Algorithms
(11)	Revision
(11)	Revision

Introduction



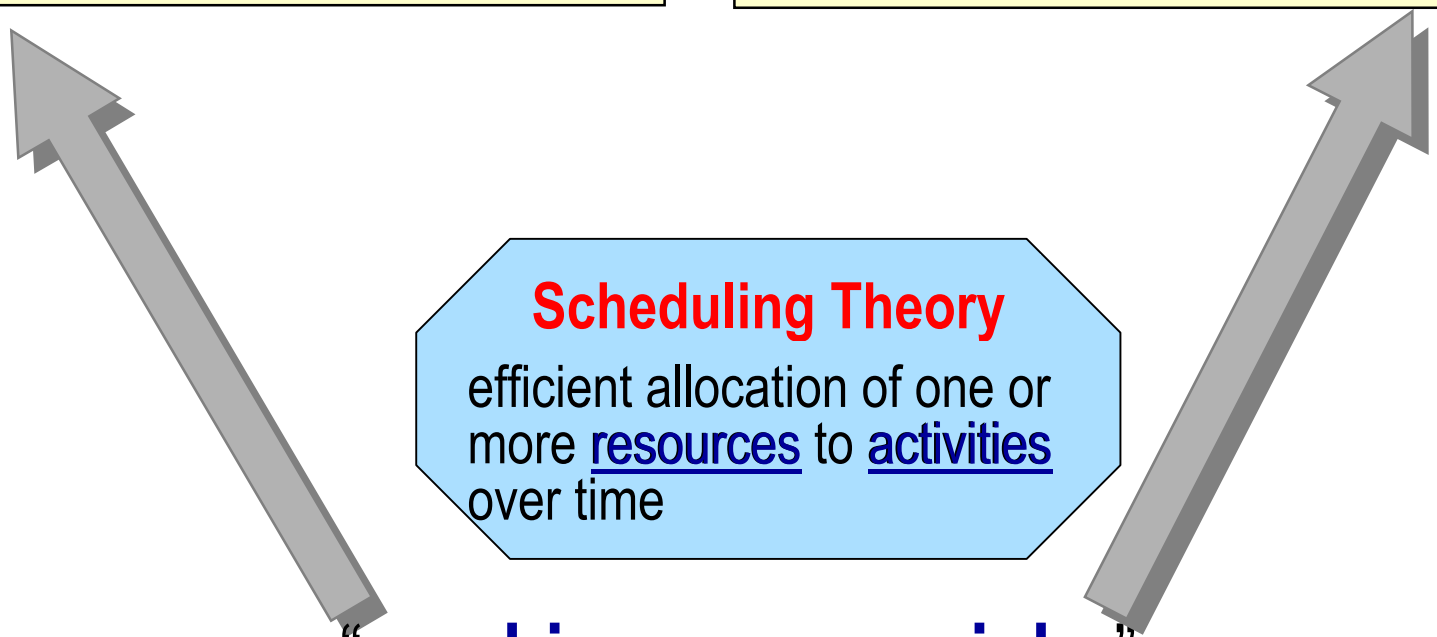
Introduction



Introduction

Resources (**machines**):
- machines at a workshop,
- runways at an airport,
- crews at a construction site,
- processing units in a computing environment.

Tasks (**jobs**):
- operations in a workshop,
- takeoffs and landings,
- stages at a construction project,
- computer programs.



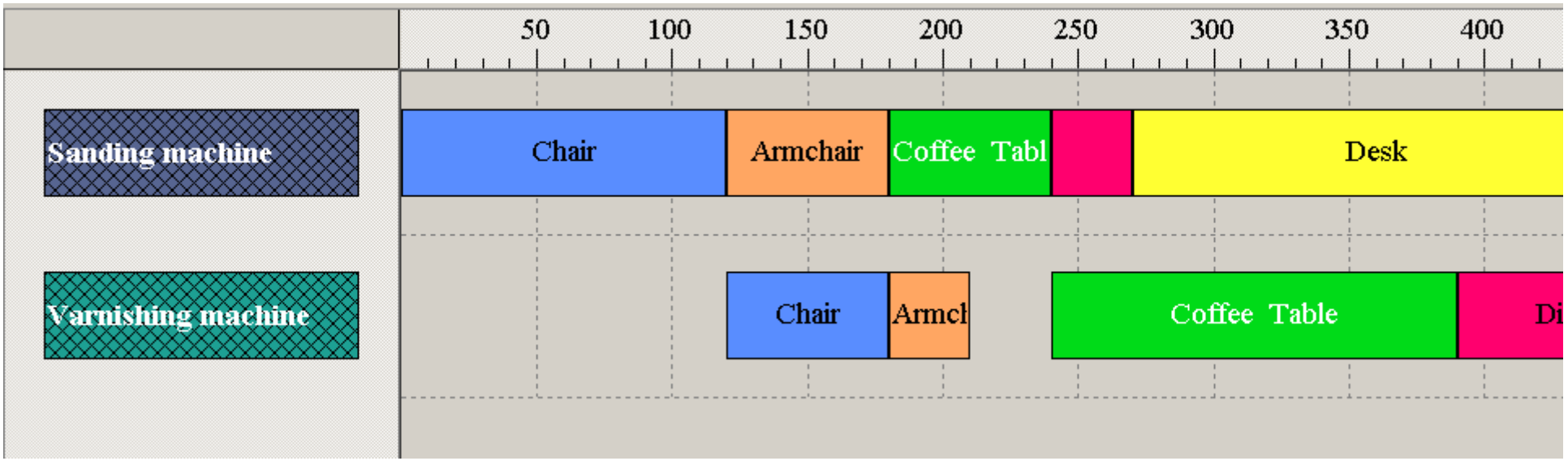
Examples of Scheduling Problems

Publishing industry**:** typesetting, actual printing, binding, packaging.	
Clothing industry: cutting, sewing, pressing, packing.	
Steel mills**:** different rods or girders pass through the set of rollers in their own orders with their own temperatures and pressure settings.	
Repair of cars in a garage	
Production planning for FMS	
Scheduling different programs on the computer	
University timetable	

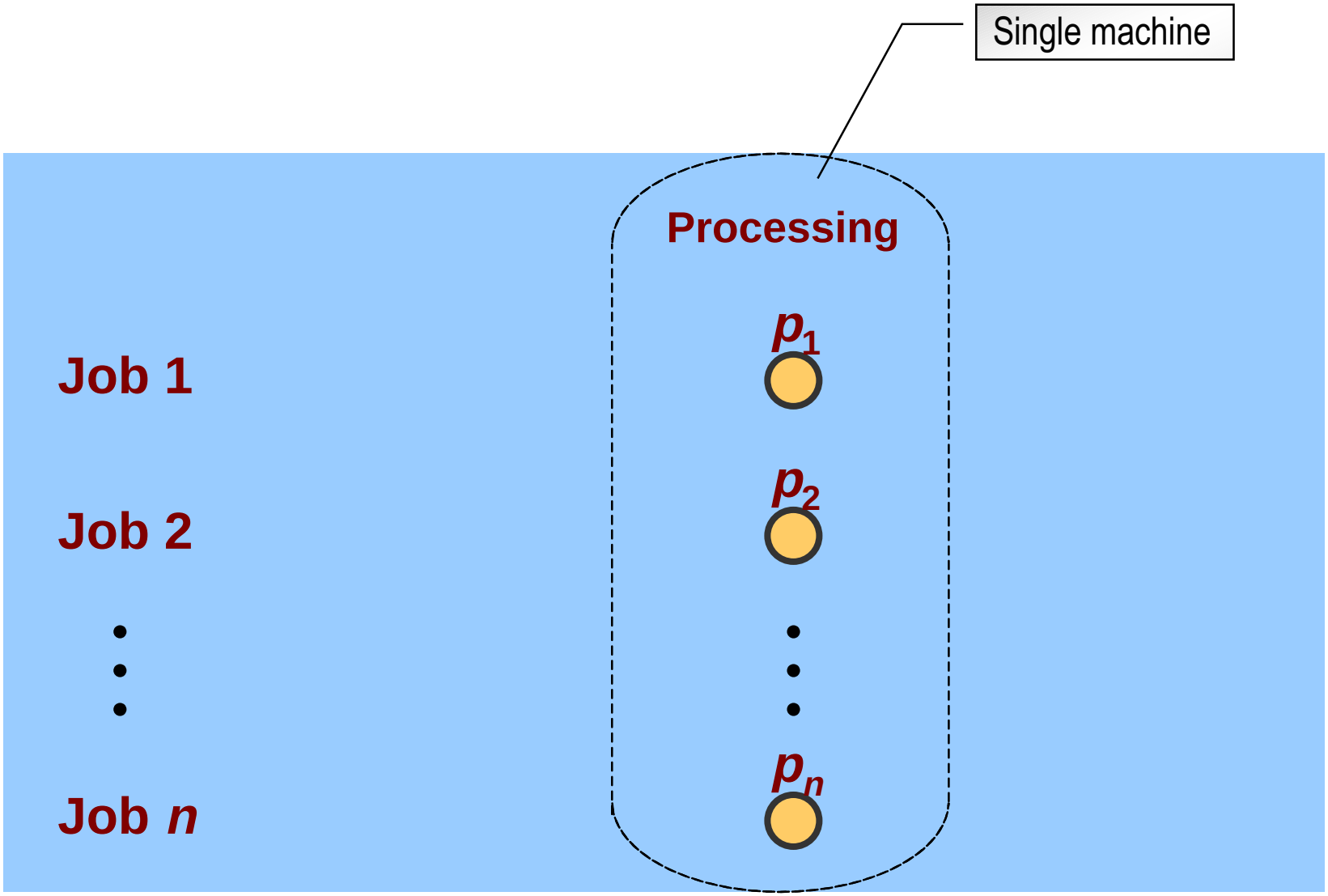
Gantt charts

Henry Laurence Gantt (1861-1919)

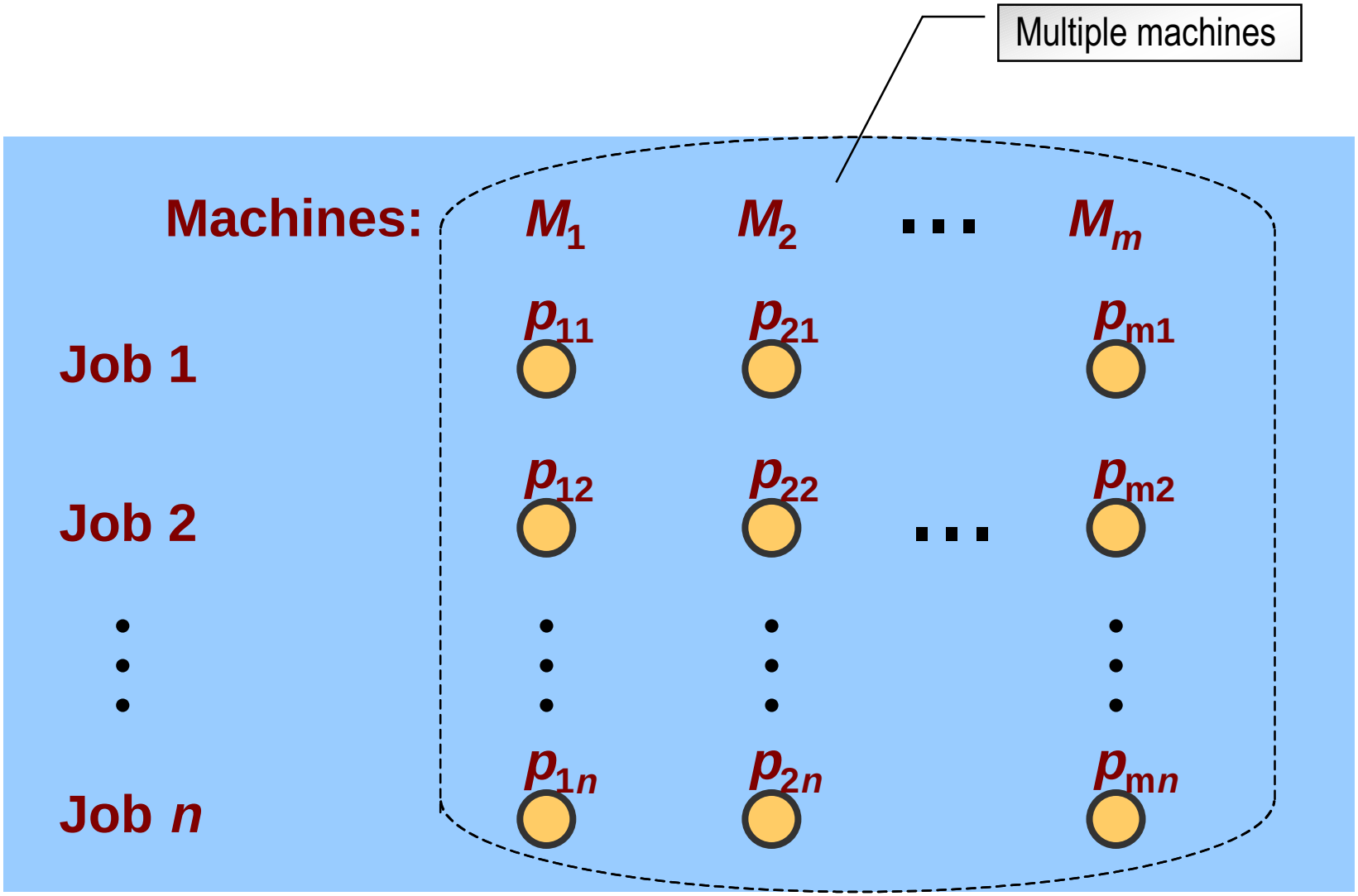
Gantt chart is the horizontal bar chart, with the x-axis representing the time and the y-axis representing machines. A colour and/or pattern code may be used to indicate operations of the same job.



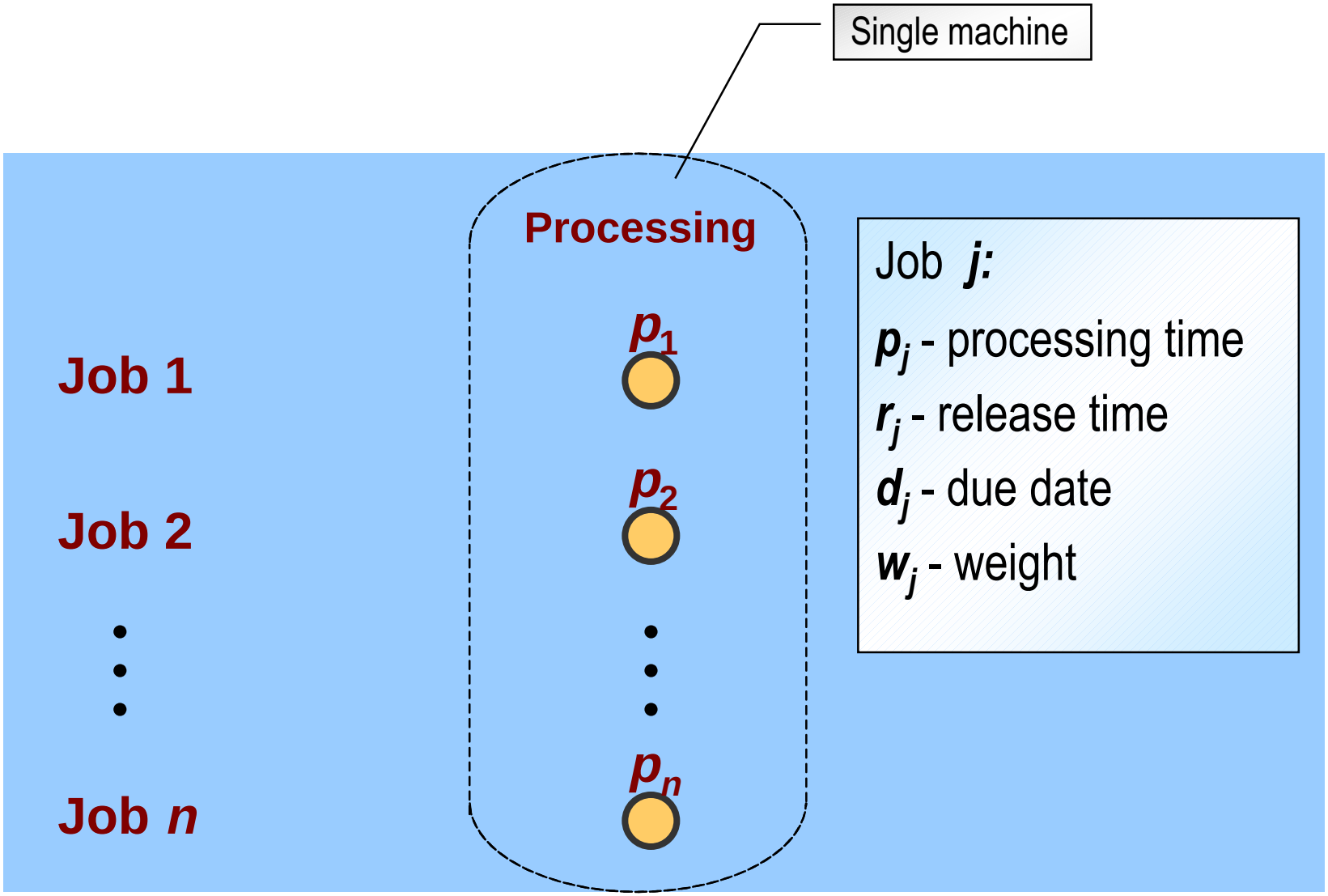
Classification of Scheduling Models



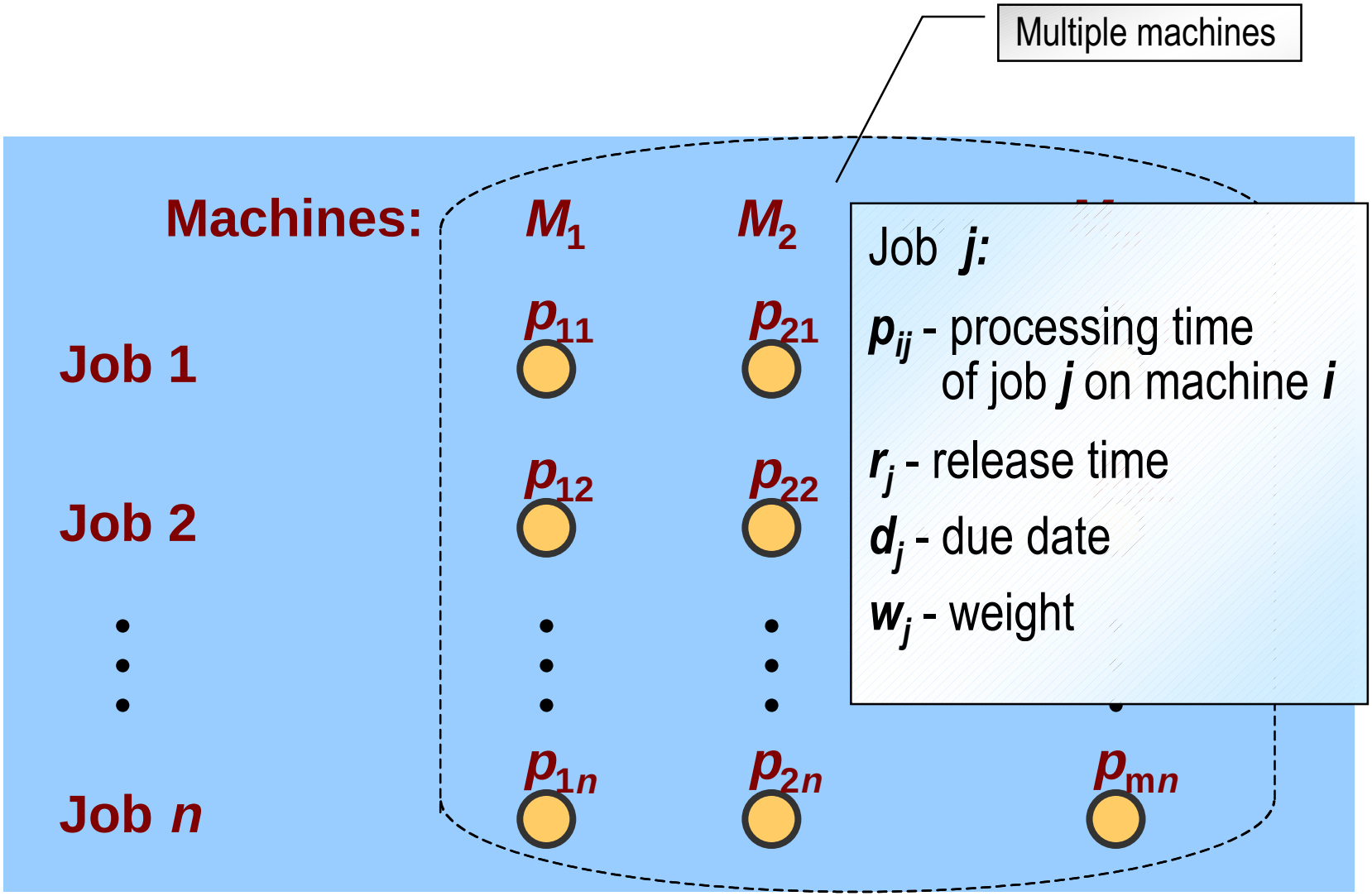
Classification of Scheduling Models



Classification of Scheduling Models



Classification of Scheduling Models



Classification of Scheduling Models

Single stage systems:

- If there is a single machine (m=1), each job should be processed by that machine exactly once.
- If there are several parallel machines, each job can be processed by any machine from the set $\{M_1, M_2, \dots, M_m\}$.

Multistage systems:

- Each job should be processed on each machine from the set $\{M_1, M_2, \dots, M_m\}$.
- Each machine can process no more than one job at a time, each job can be processed by no more than one machine at a time.

Classification of Scheduling Models

Machine environment

$\alpha|\beta|\gamma$

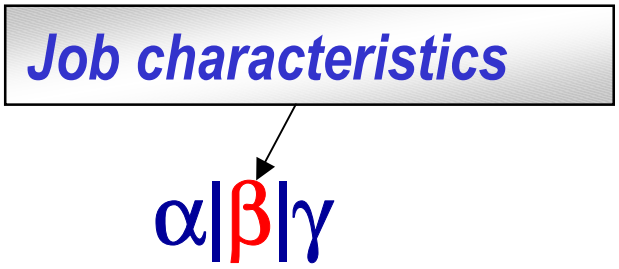
Single stage systems

$\alpha =$
- 1 – **single (dedicated) machine :** p_j - processing time of job j
- P – **identical parallel machines :** $p_{ij} = p_j$ - processing time of job j on machine i

Multistage systems

$\alpha =$
- F – **flow shop :** job j is processed first on machine 1, then on machine 2, … , and finally on machine m.
- J – **job shop :** each job has its own route to follow
- O – **open shop :** each job can be processed by the machines in an arbitrary order

Classification of Scheduling Models



There are ***n*** jobs ***N***={1, … , ***n***}

Job ***j:***

p_{ij} - processing time of job ***j*** on machine ***i***

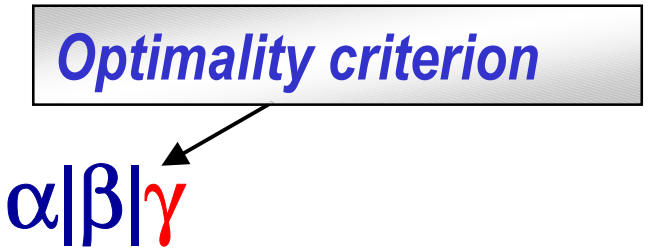
r_j - release time

d_j - due date

w_j - weight

pmtn - preemption implies the processing of any job can be interrupted and resumed later.

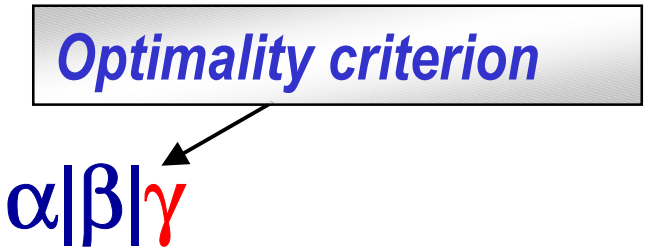
Classification of Scheduling Models



C_j - completion time of job j,
i.e., the completion time of the last operation of job j

Makespan	$C_{max} = \max \{C_j \mid j=1,\ldots,n\}$
Total completion time	$\sum C_j = \sum_{j=1}^{n} C_j$
Total weighted completion time	$\sum w_j C_j = \sum_{j=1}^{n} w_j C_j$

Classification of Scheduling Models



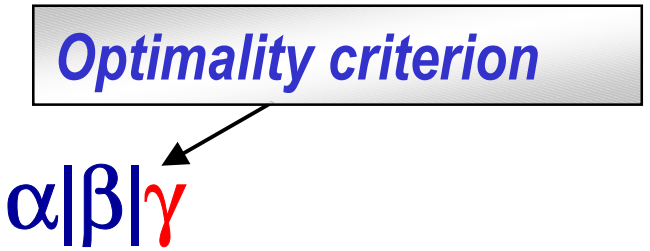
C_j - completion time of job ***j***,
i.e., the completion time of the last operation of job ***j***

$L_j = C_j - d_j$ lateness

$T_j = \max\{0, C_j - d_j\}$ tardiness

$U_j = \begin{cases} 0 \text{ if } C_j \le d_j \\ 1 \text{ otherwise} \end{cases}$ unit penalty

Classification of Scheduling Models



Maximum lateness	$L_{max} = \max \{L_j \mid j=1,\ldots,n\}$
Total tardiness	$\sum T_j = \sum_{j=1}^{n} T_j$
Total weighted tardiness	$\sum w_j T_j = \sum_{j=1}^{n} w_j T_j$
Total number of late jobs	$\sum U_j = \sum_{j=1}^{n} U_j$
Total weighted number of late jobs	$\sum w_j U_j = \sum_{j=1}^{n} w_j U_j$

Examples of Scheduling Problems

$1|r_j|L_{max}$

$1|r_j, \text{Pmtn}|L_{max}$

$P \mid p_j=1 \mid C_{max}$

$J3|p_{ij}=1|C_{max}$

Examples of Scheduling Problems

In-class exercise 1: Consider a scheduling problem with *n* readers and two books. Classify the following scheduling models:

Machines	Jobs	Objective	$\alpha\|\beta\|\gamma$
Two volumes of one book	Readers	Finish reading a.s.a.p.	?
Two volumes of one book	Readers	Minimise the cost of late book return	
Two different (independent) books	Readers	Finish reading a.s.a.p.	
Two different (independent) books	Readers (each reader has his own “reading sequence”)	Finish reading a.s.a.p.	

Examples of Scheduling Problems

Publishing industry**:** typesetting, actual printing, binding, packaging.	
Clothing industry**:** cutting, sewing, pressing, packing.	
Steel mills**:** different rods or girders pass through the set of rollers in their own orders with their own temperatures and pressure settings.	
Repair of cars in a garage**:** replace tires, repair gear box, check brakes, repair headlights, etc.	
Completing several pieces of CW so that the maximum lateness is minimised. CW j is released at time r_j, requires p_j days for completion and has a due date d_j.	
Revision schedule: starting on 13/12/2004, revise the material of 5 modules by their exam dates. Revision time for module j is p_j.	
Literature review for FYP should be based on n library books. Book j can be read in p_j days and it should be returned by its due date d_j. The library charges 30p per day on each overdue book. The objective is to minimise the total fine.	

Complexity Hierarchy