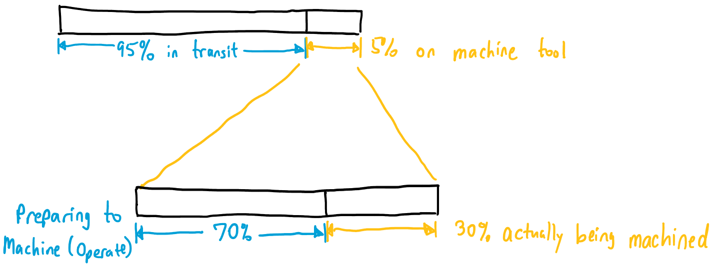
Sequencing_and_Scheduling

Tuesday, June 11, 2024 5:32 PM

Sequencing & Scheduling

- Sequencing → Sequence : A, C, D, F, E, B
- Scheduling → Schedule : Item A at 2:00
  Item C at 2:15
  Item D at 2:35
  etc ⋮

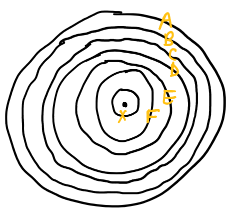
Time of Parts in the System (Production)



→ 30% of 5% = 1.5% = %1.5

- So if typical part spends 10 hours in factory floor,
  10 hr × 1.5% = 0.15 hr → 9 minutes on actual machining
- So, limited time available → Machine
  → Part

→ Sequencing & Scheduling makes available time be used more efficiently



— From outside to target:
A: Infinite number of possible schedules
B: Reduced number of schedule by stating objective function(s)
C: " " " " " making simplifying assumptions
D: Separate smaller schedules by breaking down work into groups
E: Reduced number of schedule by using algorithm
F: " " " " " " heuristic rules

D: Separate smaller schedules by breaking down work into groups
E: Reduced number of schedule by using algorithm
F: " " " " " " heuristic rules
x: achieved target with manipulation
•: target

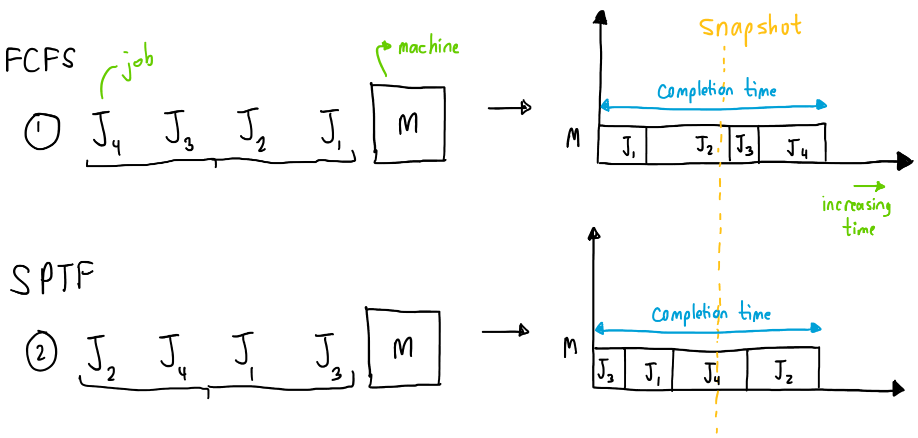
> 50 Parts → One schedule might be difficult to control

> 20 Parts → much easier to control

- Simplifying Assumptions
  > Machines never break down

- Algorithms
  → Simple Mathematical or non-mathematical rules & procedures that facilitate scheduling of operations

- Heuristics Rules
  → Very simple rules such as:
    > Shortest processing time first (SPTF)
    > First Come First Serve (FCFS)



Snapshot:

Sequence ①: Job J_1 is completed. J_2 is being processed

Sequence ②: Job J_3 and Job J_1 are completed (& out of the system).

Sequence ①: Job J_1 is completed. J_2 is being processed

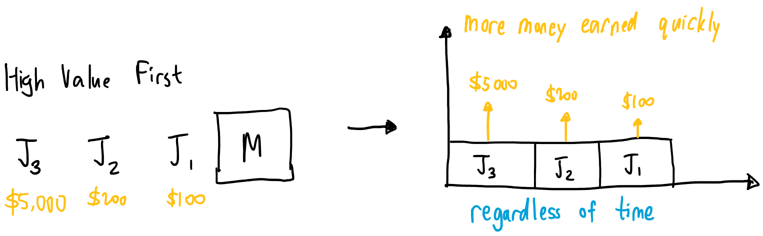
Sequence ②: Job J_3 and Job J_1 are completed (& out of the system).

J_4 is being processed

> High Value First
→ $

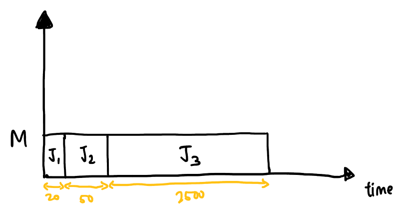
> Low Value First
→ hospitals when it comes to children

High Value First



> Job J_1 ($100) takes $20^{=5}$ units of time to process
> Job J_2 ($200) " $50^{=4}$ " " " " "
> Job J_3 ($5000) " $2500^{=2}$ " " " " "

High value first
(with new info)



Classification

> Naming the Sequencing & scheduling problems

$N / M / A / B$

- N → # of Jobs
- M → # of machines
- A → Shop type
- B → Criteria for Optimality

> Shop Types:

- F flow shop
- P permutation shop
- G general shop

▸ To be covered:

A is left blank

n / 1 / ◯ / Minimize "mean flow time"

n / 2 / G, P / Minimize "completion time"

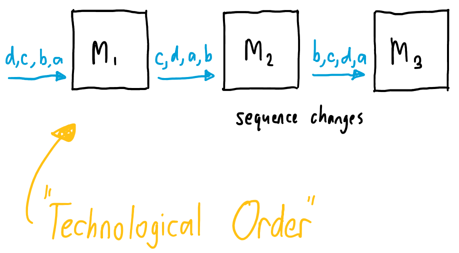
n / 3 / P / Minimize "completion time"

2 / M / F, P / Minimize "completion time"

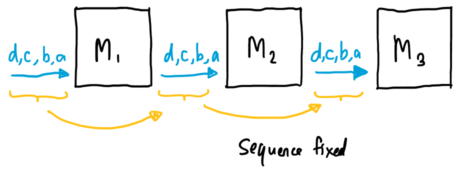
one machine doesn't form job type shop

▸ completion time can't be minimized

• Shop Types

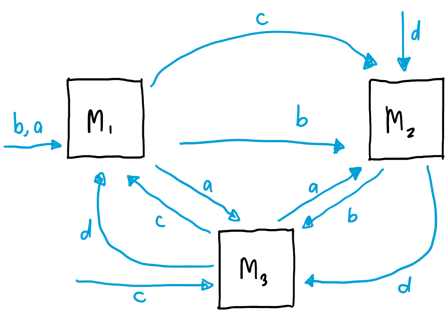


F = flow shop



P = permutation shop

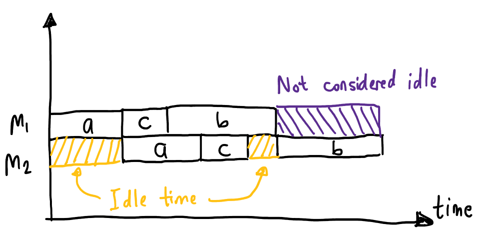
permutation repeated for all machines



G = general shop

No technological order

Another example:



$M_1 \quad M_2 \quad M_3 \rightarrow$

one machine order

Shop Type : P

a b c d | M1 | M2 | M3

a b c d | M 2 | M1 | M3

$\underbrace{\text{a b c d}}_{4! = 24}$ $\underbrace{\text{M2 M1 M3}}_{\text{all machine orders}}$

3! = 6

144

All Permutation Job-Shop

options

| a | e | c | d | b | M1 | M2 |

Steps

Processing Times

front of line

back of line

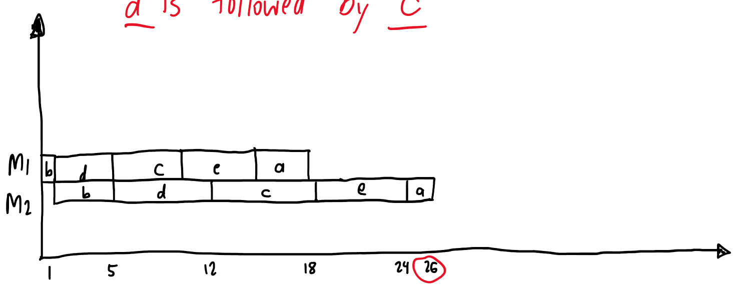
	M1	M2
a	3	2
b	1	4
c	5	6
d	4	7
e	5	6

2 1 4 3

Shortest time first

$\rightarrow 5/2/P/C_{max} \rightarrow$ minimize maximum completion time

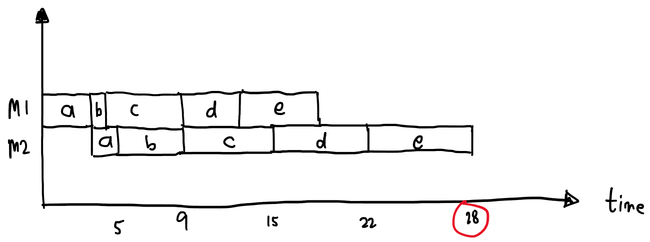
If in practice c and d use same size used (as an example), d is followed by c

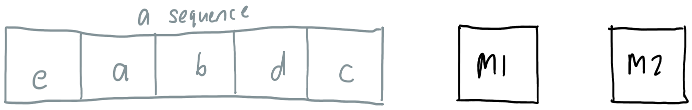


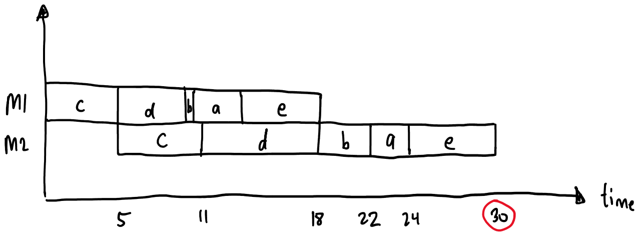
Compare to:

as given in table

| e | d | c | b | a | M1 | M2 |





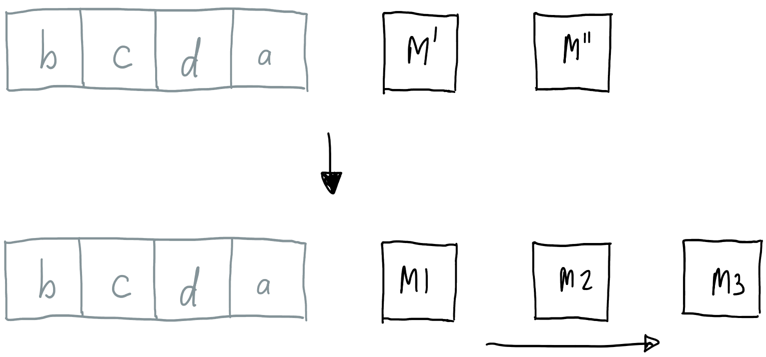


$n/3/P/C_{max}$ ← minimize maximum completion time

Job ID	M1	M2	M3
a	3	1	6
b	5	4	5
c	4	2	8
d	2	3	7

→

	Dummy M'	Dummy M''
Job ID	M1+M2	M2+M3
a	4	7
b	9	9
c	6	10
d	5	10



But there are conditions!

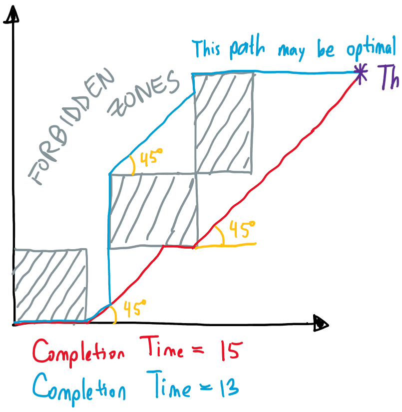
- Either:
  - > Smallest processing time on M1 must be greater than (or equal to the) processing times on M2 $Min\ t_{M1} \geq Max\ t_{M2}$
  - > Smallest processing time on M3 must be greater than (or equal to the) processing times on M2 $Min\ t_{M3} \geq Max\ t_{M2}$

Means either M1 dominates M2 or M3 dominates M2

• If the conditions don't exist, the solution may be sub-optimal, but still good

Aker's 2/m/z/ C_{max} OR F_{max}

• All the jobs (2 jobs) are available @ time zero. Then ready time (r) = 0 so $C_{max} = F_{max}$



A random job graph

Completion or equally Flow Time = Total Processing Time of Job 1 + While Job 1 waits and Job 2 is processed (+ vertical lines)

OR

Total Processing Time of Job 2 + While Job 2 waits and Job 1 is processed (sum of horizontal lines)

Either way, the results are the same