SMART
INDUSTRY
LABORATORY

Scheduling Algorithms (8)

- Production Scheduling Algorithm (2) -

Graduate School of Information,
Production and Systems

Shigeru FUJIMURA

Scheduling Problem Example

Schedule 4 jobs

that have 3 operations for each

Job	Machine / Processing Time		
	Operation 1	Operation 2	Operation 3
1	M1 / 14	M2 / 4	M3 / 8
2	M1 / 4	M3 / 10	M2 / 10
3	M2 / 6	M3 / 16	M1 / 4
4	M2 / 10	M1 / 8	M3 / 4

Scheduling Procedure

```
E : Eligible Operation Set
A : Unfinished Operation Set

initialize E (add first operations for all jobs into E);
A = ϕ;
time = 0;
while E != ϕ
      for each Machine
            randomly select one operation from operations
                  which can start at time using the Machine;
            if (the selected operation != null)
                  set starting time of the selected operation to time;
                  add the selected operation to A;
                  remove the selected operation from E;
                  add successors of the selected operation to E;
            end if
      end for
      set earliest finishing time of operations in A to time;
      remove operations which finish at time from A;
end while
```

Dispatching Rule Method

What is Dispatching Rule?

a rule to select one operation from candidates that can be allocated

Examples

- Earliest Due Date
  give priority to the operation that the delivery time is getting close most
- Critical Ratio
  give priority to the operation that the slack time ratio is the minimum
  slack time ratio = (delivery time - current time) / remained lead time
- Most Work Remaining (MWR)
  give priority to the operation that a total of remained lead time is the maximum
- Slack
  give priority to the operation that the slack time for the delivery time is the minimum

slack time = delivery time – current time – remained lead time

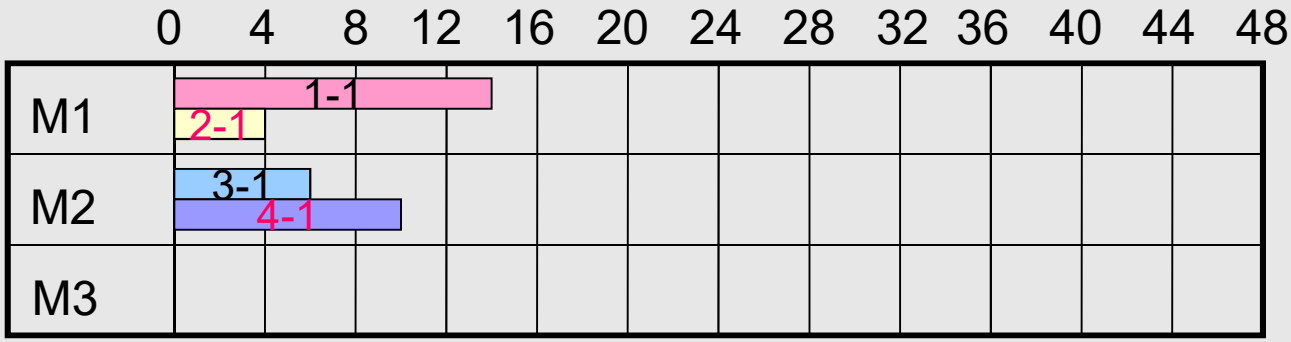
etc.

Scheduling Procedure for Dispatching Rule

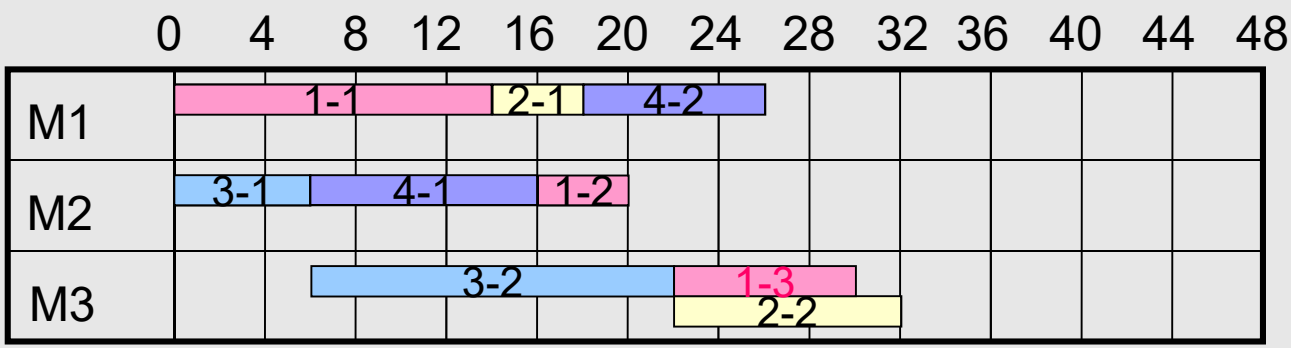
```
E : Eligible Operation Set
A : Unfinished Operation Set

initialize E (add first operations for all jobs into E);
A = ϕ;
time = 0;
while E != ϕ
    for each Machine
        pick up operations as candidates from operations
            which can start at time using the Machine;
        if (size of operations as candidates == 0)
            continue;
        if (size of operations as candidates > 1)
            select one operation using a dispatching rule;
        set starting time of the selected operation to time;
        add the selected operation to A;
        remove the selected operation from E;
        add successors of the selected operation to E;
    end for
    set earliest finishing time of operations in A to time;
    remove operations which finish at time from A;
end while
```

Most Work Remaining: MWR Rule (1)

Job	Machine / Processing Time			Delivery Time
	Ope 1	Oper 2	Ope 3	
1	M1 / 14	M2 / 4	M3 / 8	48
2	M1 / 4	M3 / 10	M2 / 10	44
3	M2 / 6	M3 / 16	M1 / 4	44
4	M2 / 10	M1 / 8	M3 / 4	38



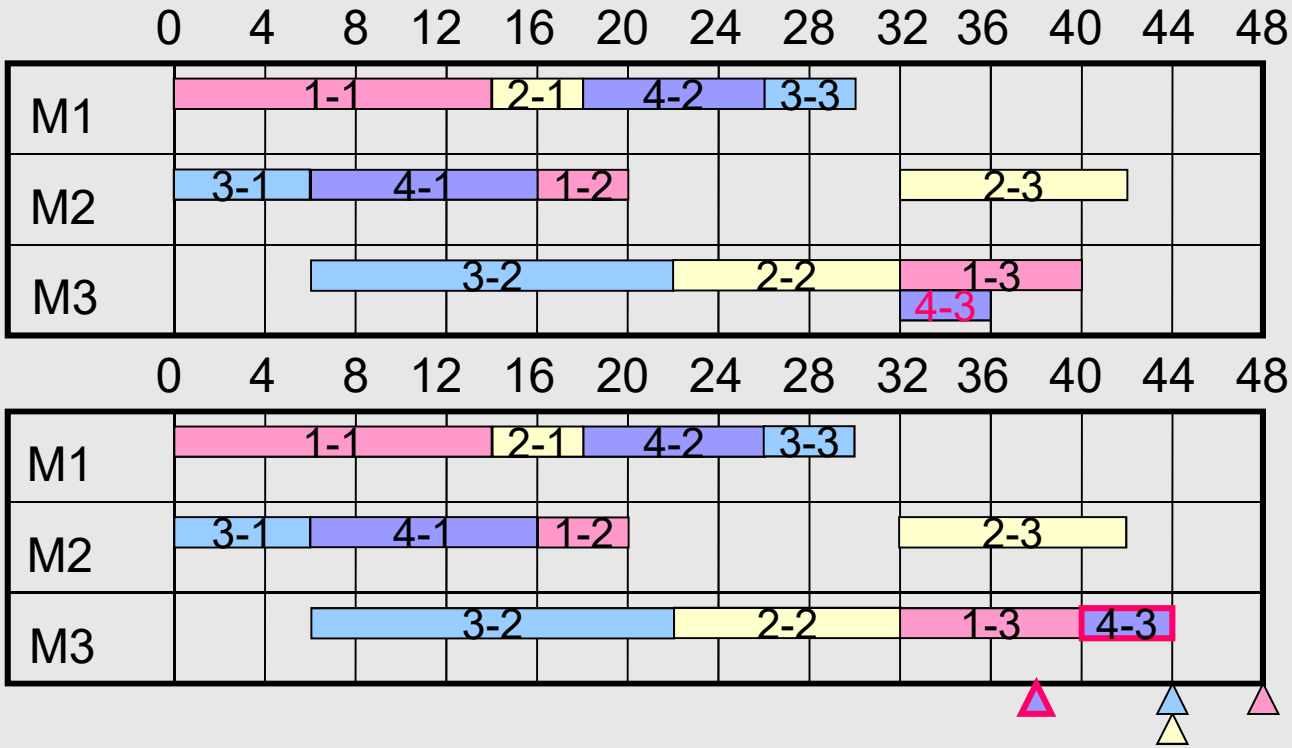
M1: 1-1: 14+4+8=26
2-1: 4+10+10=24
M2: 3-1: 6+16+4=26
4-1: 10+8+4=22



M3: 1-3: 8
2-2: 10+10=20

Most Work Remaining: MWR Rule (2)

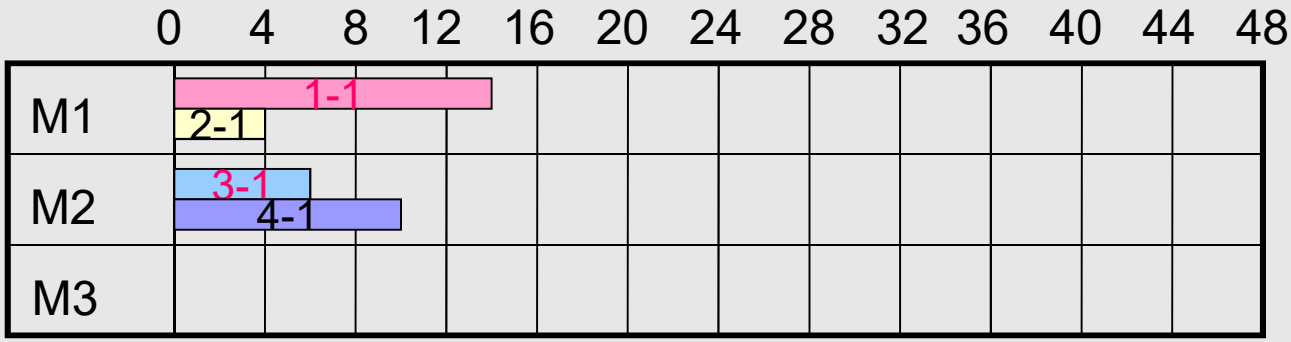
Job	Machine / Processing Time			Delivery Time
	Ope 1	Oper 2	Ope 3	
1	M1 / 14	M2 / 4	M3 / 8	48
2	M1 / 4	M3 / 10	M2 / 10	44
3	M2 / 6	M3 / 16	M1 / 4	44
4	M2 / 10	M1 / 8	M3 / 4	38



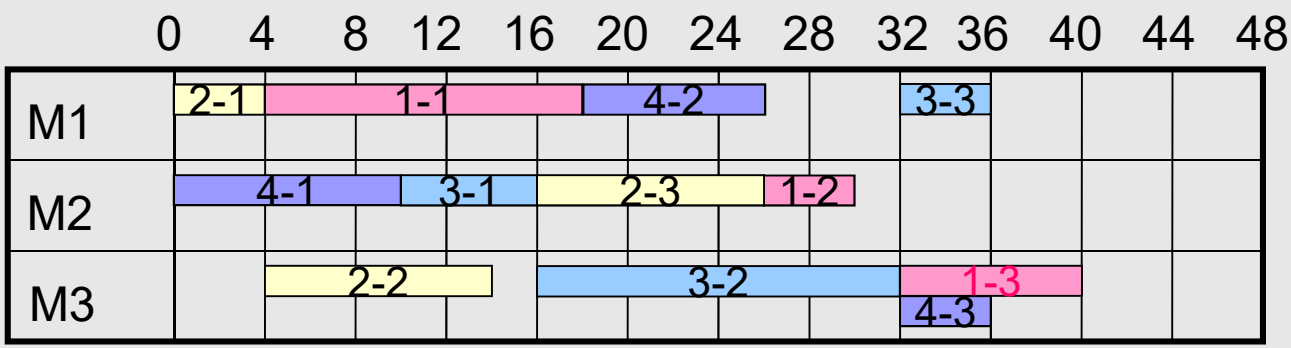
M3: 1-3: 8
4-3: 4

Slack Rule (1)

Job	Machine / Processing Time			Delivery Time
	Ope 1	Oper 2	Ope 3	
1	M1 / 14	M2 / 4	M3 / 8	48
2	M1 / 4	M3 / 10	M2 / 10	44
3	M2 / 6	M3 / 16	M1 / 4	44
4	M2 / 10	M1 / 8	M3 / 4	38



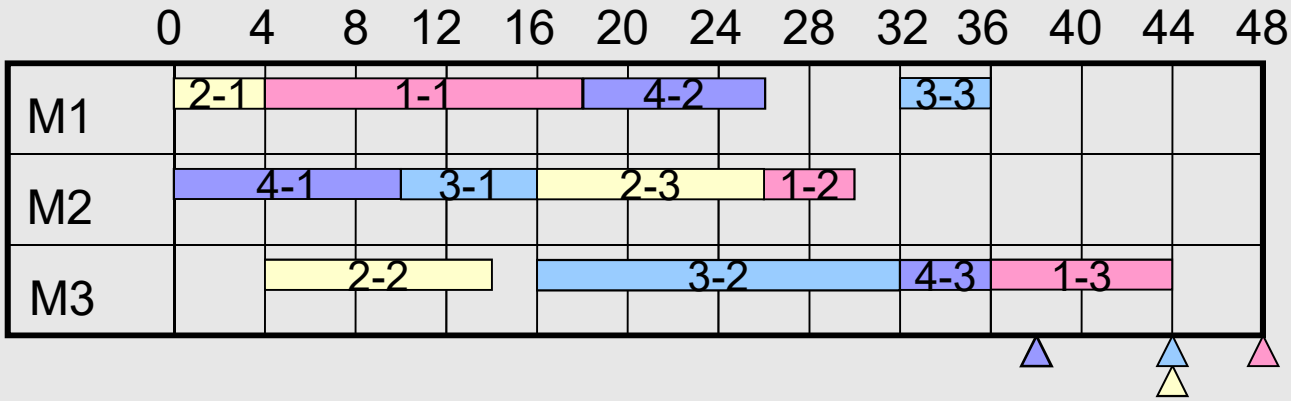
M1: 1-1: 48-0-26=22
2-1: 44-0-24=20
M2: 3-1: 44-0-26=18
4-1: 38-0-22=16



M3: 1-3: 48-32-8=8
4-3: 38-32-4=2

Slack Rule (2)

Job	Machine / Processing Time			Delivery Time
	Ope 1	Oper 2	Ope 3	
1	M1 / 14	M2 / 4	M3 / 8	48
2	M1 / 4	M3 / 10	M2 / 10	44
3	M2 / 6	M3 / 16	M1 / 4	44
4	M2 / 10	M1 / 8	M3 / 4	38



M3: 1-3: 48-32-8=8

4-3: 38-32-4=2

SMART
INDUSTRY
LABORATORY

Thank you