40.302: Adv Optimization

Guo Ziqi - 1000905
Liu Sidian - 1000909
Wang Yiran - 1000906

IP Formulation **Minimize Makespan Cmax**

$$Min \qquad C_{max}$$

One position has at most one job	$\sum_{i=1}^{m}\sum_{j=1}^{n} x_{i,j,k} \leq 1 \quad \forall k$
One job must be assigned to one position	$\sum_{i=1}^{m}\sum_{k=1}^{n} x_{i,j,k} = 1 \quad \forall j$
Job in position k is completed its release and processing	$h_{i,k} \geq \sum_{j=1}^{n}(p_{i,j} + r_j)\, x_{i,j,k} \quad \forall i, j, k$
Job in position k is completed after job in position k-1	$h_{i,k} \geq h_{i,k-1} + \sum_{j=1}^{n} x_{i,j,k} * p_{i,j}$
Lower bound of Makespan	$C_{max} \geq h_{i,k} \quad \forall i, k$
Position starts from 1	$\sum_{j=1}^{n} x_{i,j,k} \leq \sum_{j=1}^{n} x_{i,j,k-1} \quad \forall i, \forall k > 1$
Binary constraint	$x_{i,j,k} \in \{0,1\}$
** Improve the calculation speed*	$x_{i,j,k} = 0 \quad \forall i, j, p_{i,j} = 2000$

$x_{i,j,k}$ — Binary Indicator of whether job j is in position k on machine i

$h_{i,k}$ — Completion time of job in position k on machine i

Completion time of job i on machine j removed under this objective function

Minimize Makespan Cmax **Result**

Cmax* = 442.12

Computation Time = 1.78 sec

M1

1	3	2	11	40	10	19	37	46	29	41	34	47	35	42	12	33	8	23	31	32	9	22

M2

4	5	6	28	7	17	36	27	14	25	30	16	13	21	24	15	26	45	39	44	38	18	20	43

Extension: Fairness in Scheduling Problems

Motivation

1. People feel better when they know why they are waiting and the schedule caters to their perceived "fairness".
2. Fairness could potentially go hand-in-hand with other societal objectives

Objective

1. With IP, compare the efficacy of various definitions of fairness from administrator's perspective
2. Assess the feasibility of heuristics in achieving fairness

Data Preparation

1. Reduce job quantity to 15 for manageability
2. Replace 2000 with realistic values

IP Formulation 1 : Minimize Maximum Waiting Time

1 New variables and constraints

W(i,j,k):
Waiting Time of Job j in position k on machine i

$$w_{i,j,k} \geq h_{i,k-1} - r_j - M \times (1 - x_{i,j,k}) \qquad \forall\, i, j, \forall k > 1$$

$$w_{max} \geq w_{i,j,k} \qquad \forall i, j, k$$

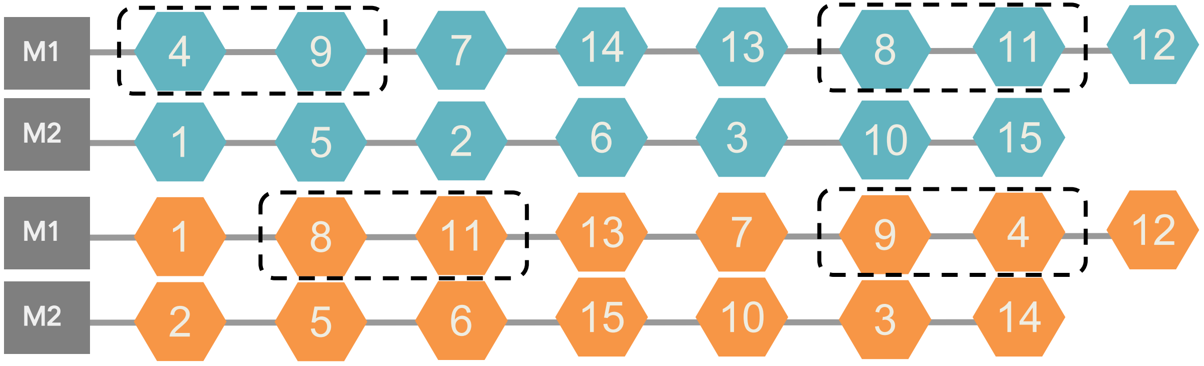
$$w_{i,j,k} \leq M \times x_{i,j,k} \qquad \forall i, j, k$$

$$w_{i,j,k} \geq 0 \qquad \forall i, j, k$$

2 Results comparison

Obj Func	Wmax	Cmax	Time (s)
min Wmax	65.62	149.1	28.8
min Cmax	81	146.2	0.5

3 Optimal sequence & analysis



Rationale

Allow more waiting time for jobs that require long processing time.

1 New variables and constraints

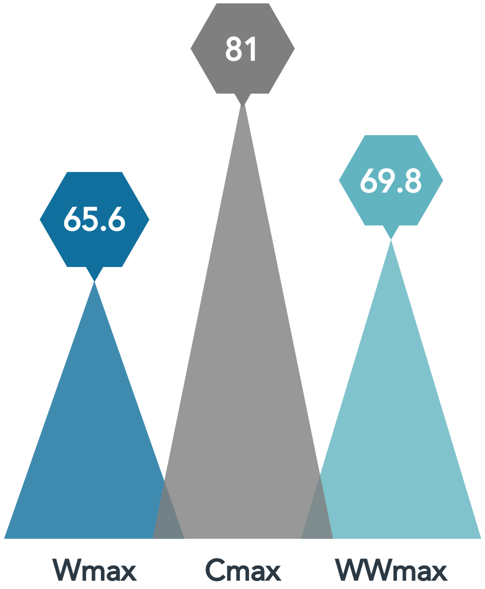
$$ww_{i,j} = \frac{\sum_{k=1}^{m} w_{i,j,k}}{p_{i,j}} \quad \forall i,j$$

$$ww_{max} \geq ww_{i,j} \quad \forall i,j$$

2 Results comparison

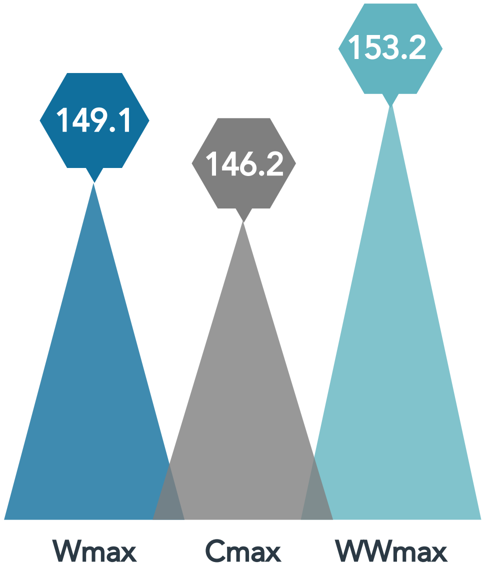
Obj Func	Wmax	Cmax	Time (s)
min WWmax	69.78	153.2	50.75
Min Cmax	81	146.2	0.5

IP Formulation: Wmax OR WWmax



Maximum Waiting Time

WWmax although has slightly higher Wmax, as expected, it still greatly reduces Wmax from Cmax as an objective.

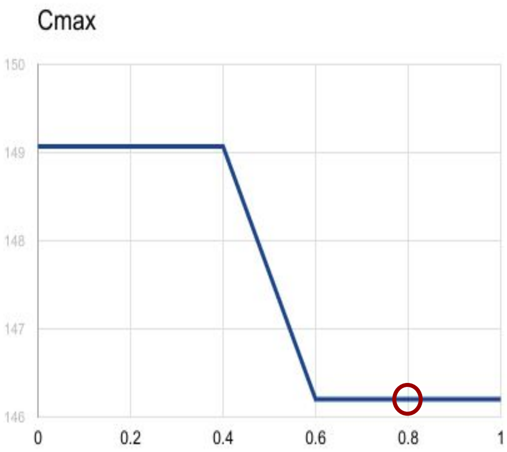


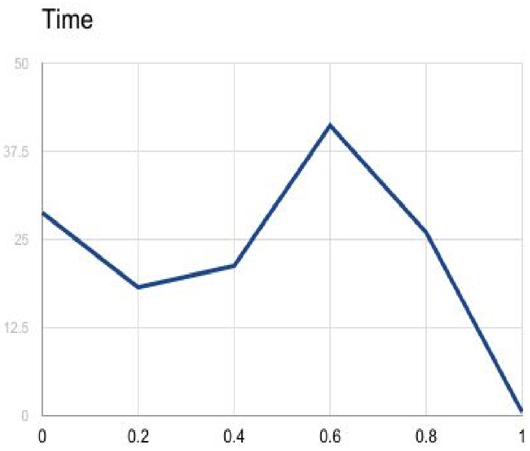
Makespan

Wmax is relatively close to the optimal makespan achieved by Cmax, followed by WWmax

IP Formulation 3 : Hybrid Wmax Model

Objective function: k * Cmax + Wmax





Heuristic Approach: Greedy Algorithm

1

First Come First Serve

When one job is completed, process the job that has been released for the longest time.

- No machine idle time
- Maximum waiting time: 68.7
- Makespan: 157.1

2

First Complete First Serve

When one job is completed, process the job with the smallest expected completion time.

- Potential machine idle time
- Maximum waiting time: 69.7
- Makespan: 158.1

Putting It All Together

Key Takeaway

Speed

It takes significantly longer time to solve an IP with fairness as the objective

Compatibility

Minimal sacrifice in makespan for huge reduction in waiting time and improvement in perceived services

Adaptability

Fairness can be achieved with varying definitions that are tailored for different scenarios

Heuristic

"First Come First Serve" principle is a simple yet effective method in achieving fairness