Introduction to Optimization
Term Project

Single product unrelated machine maintenance scheduling problem

r09725006 劉心鈺

r09725023 姚佳妤

r09725041 鄭宇伶

2021/1/5

Contents

Abstract

Facing the production scheduling problem with the objective of minimizing the order tardiness, make-up work could increase the production. However, as the machine yield decreases, the proportion of defective products become higher, therefore the total production time must be extended, and this machine will take the lower yield rate. The machine yield would recover if preventive maintenance is implemented at the appropriate time, but production would temporarily stop during the process and may even cause order delays.

In order to trade off the yield declining caused by continuous production and the opportunity cost of the machine maintenance, this study considers the maintenance cost of the machine and the due period of each order. Under the conditions of multiple machines, multiple orders, different production rates, and the yield declining rates of each machine, we develop a Mixed Integer Program to find the best downtime for maintenance that could minimize the tardiness.

Keywords: *Yield of the machine, Preventive maintenance, Machine scheduling problem, Mixed Integer Programming*

1 Motivation and Background

Production and maintenance scheduling are two areas that have received tremendous attention in the manufacturing industry. They are closely related because machines need to be maintained, and hence it will interrupt the production. However, most machine scheduling models assume that the machines are available all of the time which is not realistic. Without considering the declining yield rate and the probability of machine failures, it would underestimate the production cost. Besides, unexpected machine breakdowns would mess up the schedule. Consequently, production and maintenance activities should be scheduled simultaneously when we deal with real-world problem.

In fact, job scheduling is a classic optimization problem which has been widely studied in the past decades. Nevertheless, there are less literature coordinating the production and maintenance scheduling. Some previous literature decides a certain time period of preventive maintenance and consider the random failure as a cost of production. Those researches may overlook the trade-offs between maintenance and production. Maintenance affect available production time and may cause the order tardiness, though production would become inefficient without maintenance. Consequently, there is a gap in the literature as to how joint preventive maintenance and production to make scheduling decisions.

Moreover, scheduling problems are an NP problem Ullman, 1975 which is believed that there is no polynomial time algorithm for solving NP-hard problems. Hence, the maintenance scheduling is a worthwhile area of study. This is the reason why we chose this problem as final project and devoted our time and energy to study. Although most of NP-hard problem would be studied by proposing a heuristic algorithm, for term project requirements, we proposed an MIP model to solve the maintenance scheduling problem and discussed how large problem size can be addressed by Gurobi within an appropriate period of time.

2 Literature Review

Planning and scheduling are two fields of research for different decision-making levels in most manufacturing industries. Production planning problem is dedicated to get a midterm tactical plan. By contrast, production scheduling problem is to find an optimal short term execution plan. Both these two level of decision-making for production need to consider the impact of machines maintenance. As for our research, we would focus on how to joint production and maintenance to get an appropriate schedule.

Lee and Chen, 2000 study the maintenance scheduling problem for parallel machines. They assume that each machine must be maintained once during the planning horizon. The objective is to arrange production and maintenance activities such that total weighted completion time is minimized. Besides, they also consider whether resources are sufficient for machines and thus they can be maintained simultaneously. In the paper, they proved that both case as to whether maintenance resources are limited or not are NP-hard problems. Hence, they proposed branch and bound algorithms based on the column generation approach for solving both cases of the problem.

Cassady and Kutanoglu, 2005 proposed an integrated model that coordinates preventive maintenance planning decisions with single-machine scheduling decisions. The objective is to minimize the total expected weighted completion time of jobs. They compared the performance of scheduling problem whether preventive maintenance and production are integrated. By conducting an experimental study, they proved that integrating the two decision-making processes resulted in significant improvement. Besides, by their research, they recommended the integrated scheduling should be focused on bottleneck machines.

In this study, we deal with the maintenance scheduling for unrelated machines such that total tardiness is minimized. Unrelated machines mean the processing time of the job can differ on different machines. Besides, we assume that the yields are declined in fixed rate but different machines could have different yield rates. We also take limitation of maintenance resources into consideration. For this term project, we propose an MIP model to solve this problem and to discussed how large problem size can be addressed by Gurobi within an appropriate period of time.

3 Problem Statement

In our project, we consider a single product job scheduling problem with maintenance. There are orders with different demand levels required to be satisfied. A decision maker will decide the production and the maintenance schedule in each machine to minimize the total tardiness.

We are given a planning horizon $T = \{1, 2, ..., |T|\}$ including $|T|$ periods of fixed length. Let $N = \{1, 2, ..., |N|\}$ denote the set of orders we need to fulfill and $M = \{1, 2, ..., |M|\}$ represent the set of machines. Each order requires certain demand quantity

Q_j at its given due time D_j. The ideal production rates of machines are different, and we denote A_i as the ideal production rate of machine i. Because the yield rate decreases as time progresses, we denote r_{it} as the yield rate of machine i at time period t.

The yield rate declines as a function of time. We formulate the function as a linear function with constant decline rate to describe the deteriorating production system. Let B_i denote the yield declining rate of machine i, which means that the yield of machine i decreases B_i units every time period.Both yield and yield declining rate are in the range of 0 to 1. Here is a simple example. Given that machine 1 produces 100 units of products per time period, and that the current yield and the yield declining rate are 90% and 5% respectively, we may derive that $A_1 = 100$, $r_{1,t} = 0.9$, $B_1 = 0.05$ and the yield at next time period $r_{1,t+1}$ is $0.9 - 0.05 = 0.85$. The production quantity of machine i at time period $t + 1$ is $100 \cdot 0.85 = 85$ units.

The yield may not drop to zero and there is a lower bound of yield for each machine, which is denoted as L_i. Without maintenance, the yield would decline incrementally until it reaches L_i. As the yield falls at L_i, it remains constant without maintenance. There is a limitation on the total number of machines under maintenance within a time period. We may arrange a maintenance on any machine at any time period if the number of machines under maintenance at the time period does not exceed H. Machine i consumes F_i units of consecutive time periods to take a maintenance. The maintenance process is not allowed to be suspended and the production of the machine must be ceased during the maintenance. The yield of machine i would recover and rise to 1 immediately at the next time period of the completion of its maintenance, and the yield would proceed declining at a constant rate of B_i from 1 later on.

We may assign each order to exactly one machine and preemption is not enabled. To satisfy an order, the assigned machine would consume several consecutive time periods in which the sum of production quantity is larger or equal to the order quantity. Note that once an order is processed on a machine, any other order or maintenance may not inserted until the machine completes the current order production.

There are three essential decision variables y_{ijt}, x_{it} and w_j in this model. The decision variable y_{ijt} is 1 if the machine i is processing order j at time period t. The decision variable x_{it} is 1 if the machine i is in the maintenance process. The decision variable w_j denotes the completion time of order j.

4 Research Model

We formulate a Mixed Integer Program (MIP) to find optimal solutions for the problem at hand. The problem is formulated as

$$
\begin{aligned}
\min \quad & \sum_{j \in J} p_j && && (1)\\
\text{s.t.} \quad & \sum_{j \in N} y_{ijt} \leq 1 && \forall i \in M, \quad \forall t \in T && (2)\\
& \sum_{i \in M} y_{ijt} \leq 1 && \forall j \in N, \quad \forall t \in T && (3)\\
& \sum_{i \in M} z_{ij} = 1 && \forall j \in N && (4)\\
& M_1 z_{ij} \geq \sum_{t \in T} y_{ijt} && \forall i \in M, \quad \forall j \in N && (5)\\
& \sum_{j \in N} y_{ijt} \leq (1 - x_{it}) && \forall i \in M, \quad \forall t \in T && (6)\\
& \sum_{i \in M} x_{it} \leq H && \forall t \in T && (7)\\
& v_{it} = A_i r_{it} && \forall i \in M, \quad \forall t \in T && (8)\\
& \sum_{t \in T} \sum_{i \in M} g_{ijt} \geq Q_j && \forall j \in N && (9)\\
& g_{ijt} \leq v_{it} && \forall i \in M, \quad \forall j \in N, \quad \forall t \in T && (10)\\
& g_{ijt} \leq M_2 y_{ijt} && \forall i \in M, \quad \forall j \in N, \quad \forall t \in T && (11)\\
& 2e_{it} \leq x_{i,t-1} - x_{it} + 1 && \forall i \in M, \quad \forall t \in T && (12)\\
& r_{it} \leq M_3 e_{it} + M_4 s_{it} + r_{i,t-1} - B_i && \forall i \in M, \quad \forall t \in T && (13)\\
& r_{it} \leq M_3 e_{it} + M_5 (1 - s_{it}) + L_i && \forall i \in M, \quad \forall t \in T && (14)\\
& r_{it} \leq 1 && \forall i \in M, \quad \forall t \in T && (15)\\
& u_{ijt} \geq y_{ijt} - y_{ij,t+1} && \forall i \in M, \quad \forall j \in N, \quad \forall t \in T && (16)\\
& y_{ij,T+1} = 0 && \forall i \in M, \quad \forall j \in N && (17)\\
& \sum_{i \in M} \sum_{t \in T} u_{ijt} = 1 && \forall j \in N && (18)\\
& w_j \geq \sum_{i \in M} \sum_{t \in T} t u_{ijt} && \forall j \in N && (19)
\end{aligned}
$$

$$F_i - \sum_{k=t-F_i}^{t-1} x_{ik} \leq M_7(1 - e_{it}) \qquad \forall i \in M, \quad \forall t = F_i + 1, \cdots, |T| \qquad (20)$$

$$x_{i,0} = 0 \qquad \forall i \in M \qquad (21)$$

$$p_j \geq w_j - D_j \qquad \forall j \in N \qquad (22)$$

$$p_j \geq 0 \qquad \forall j \in N \qquad (23)$$

$$r_{it} \geq 0 \qquad \forall i \in M, \quad \forall t \in T \qquad (24)$$

$$v_{it} \geq 0 \qquad \forall i \in M, \quad \forall t \in T \qquad (25)$$

$$w_j \geq 0 \qquad \forall j \in N \qquad (26)$$

$$p_j \geq 0 \qquad \forall j \in N \qquad (27)$$

$$g_{ijt} \geq 0 \qquad \forall i \in M, \quad \forall j \in N, \quad \forall t \in T \qquad (28)$$

$$x_{it} \in \{0, 1\} \qquad \forall i \in M, \quad \forall t \in T \qquad (29)$$

$$z_{ij} \in \{0, 1\} \qquad \forall i \in M, \quad \forall j \in N \qquad (30)$$

$$y_{ijt} \in \{0, 1\} \qquad \forall i \in M, \quad \forall j \in N, \quad \forall t \in T \qquad (31)$$

$$s_{it} \in \{0, 1\} \qquad \forall i \in M, \quad \forall t \in T \qquad (32)$$

$$u_{ijt} \in \{0, 1\} \qquad \forall i \in M, \quad \forall j \in N, \quad \forall t \in T \qquad (33)$$

$$e_{it} \in \{0, 1\} \qquad \forall i \in M, \quad \forall t \in T \qquad (34)$$

where $M1$, $M2$, $M3$, $M4$, $M5$, $M6$, and $M7$ are seven large numbers.

The objective (1) is to minimize the total tardiness in a scheduling problem. Constraints (2) ensure that each machine can only deal with one order at a time. Constraints (3) ensure that each order is assigned to one machine at a time. Constraints (4) and (5) ensure that each order is assigned to one machine. Constraints (6) ensure that each machine should stop producing during maintenance. Constraints (7) define the upper bound of the number of machines under maintenance in the same period. Constraints (8) define the production rate after considering the yield of each machine in every period. Constraints (9) ensure that each order should be fulfilled. Constraints (10) and (11) define the produced quantity for each order in every single period. Constraints (12) to (15) define that the yield of each machine would recover and rise to 1 immediately at the next period of the completion of its maintenance; otherwise, the yield would decrease, but it should never lower than the lower bound. Constraints (16) define the completion point for each order, constraints (17) state that the completion point would never in the dummy time period $T + 1$, and constraints (18) ensure that there is one and only one completion point at each machine. These three constraints could make sure it is non-preemptive scheduling. Constraints (19) define the completion time for each order. Constraints (20) ensure that the maintenance for each machine should be continuous. Constraints (21) state that the machine would never be maintained at the dummy time period 0. Constraints (22) and (23) define the tardiness of each order. Constraints (24) to (28) are the non-negative restrictions. Constraints (24) to (28) specify that the decision variable x_{it}, z_{ij}, y_{ijt}, s_{it}, u_{ijt} and e_{it} are binary over all domains. Table 1 introduce all the notations mentioned above.

This formulation total has $3 \cdot |M| \cdot |N| \cdot |T| + |M| \cdot |N| + 5 \cdot |M| \cdot |T| + 2 \cdot |N|$ variables and $3 \cdot |M| \cdot |N| \cdot |T| + 2 \cdot |M| \cdot |N| + 8 \cdot |M| \cdot |T| + |N| \cdot |T| + |M| + 5 \cdot |N| + |T|$ constraints. In the next section, we use this formulation to do some numerical analysis.

Indices	
i	$1, ..., \|M\|$: indices of machines.
j, k	$1, ..., \|N\|$: indices of orders.
t	$1, ..., \|T\|$: indices of periods.
Parameters	
A_i	The ideal production rate of machine i.
B_i	The yield declining rate of machine i, $B_i \in [0, 1]$.
F_i	The units of consecutive time periods to take a maintenance of machine i.
L_i	The yield lower bound of machine i, $L_i \in [0, 1]$.
Q_j	The demand quantity for order j.
D_j	The due time of order j.
H	The upper bound of the number of machines under maintenance in the same period.
Variables	
r_{it}	The yield of machine i at time period t, $r_{it} \in [0, 1]$.
v_{it}	Consider the yield, the production rate of the machine i at the time period t.
w_j	The completion time of order j.
p_j	The tardiness of order j.
x_{it}	1 if machine i is under maintenance at the time point t and 0 otherwise.
z_{ij}	1 if machine i is processing order j and 0 otherwise.
y_{ijt}	1 if machine i is processing order j at time period t and 0 otherwise.
g_{ijt}	The production rate of machine i if machine i is processing order j at time period t and 0 otherwise.
s_{it}	1 if the yield is lower than L_i at the time period $t + 1$ and 0 otherwise.
u_{ijt}	1 if order j is completed at machine i at time period t and 0 otherwise.
e_{it}	1 if time period t is the first period after machine i finished maintenance and 0 otherwise.

Table 1: List of Notations

5 Research Results

For the research results, we used Gurobi in Python to solve the model. The computation was made on IBM T60 laptop with Windows 10 with Intel® Core™ i7-8700 CPU @ 3.20GHz $\times$ 12 and 15.5 Gibibyte memory. We examined the model by two-machines seven-orders problem (i.e. $|M| = 2$, $|N| = 7$). The detailed parameter setting for this problem is show in Table 2. We set $|T| = \frac{\sum_{j \in N} Q_j}{\sum_{i \in M}(A_i \times r')}$ where $r' = \frac{1+L_i}{3}$.

M	A_i	B_i	F_i	$r_{i,0}$	L_i
machine 1	100	5	3	50	20
machine 2	90	10	2	30	10
N	Q_j	D_j			
order 1	150	7			
order 2	200	6			
order 3	300	8			
order 4	280	5			
order 5	280	1			
order 6	200	3			
order 7	90	10			

Table 2: parameter setting for two-machines seven-orders problem

After the computation, we got the optimal production and maintenance schedule which was conclude as Gantt chart in Figure 1. The blue timelines mean that the machine is under production and the gray timelines mean that the machine is under maintenance. The tardiness time for each order is shown in Table 3. The optimal objective value is 40 and the makespan is 17. The computation time for this problem is 878.426 seconds. It is worth noting that machines are allowed to be maintenance in the initial. For example, in this problem, machine 2 needs to be maintenance at the beginning because we set the initial yield in the lower bound of yield for machine 2.

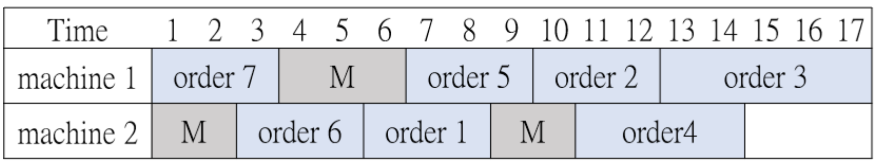

Figure 1: the schedule of computation result

Order	1	2	3	4	5	6	7
Tardiness	1	6	9	9	8	2	0

Table 3: The tardiness of each order

6 Discussion and Analysis

Because the scheduling problem is an NP-hard problem, we would like to know how large problem size can be addressed by Gurobi within an appropriate period of time. Hence, we designed two experiments for our model. The first experiment is to compare the optimality gap of different problem sizes within a given appropriate period of time. The second experiment is to calculate the declining rate of the optimality gap and tries to find an appropriate stopping time for different problem sizes.

The experiments were examined by solving 90 different problems. To generate the problem data, the parameters were set as follows and a uniform distribution was employed. We set the number of machines $|M| = 1,\ 3,\ 5$ and the number of orders $|N| = 5,\ 10,\ 15$. Thus, there are 9 problem sizes to be studied. For each problem size, 10 tests were generated.

Parameter setting

- $Q_j \sim \text{Uni}(500,\ 700)$
- $A_{ij} = 200$
- $r_{i,0} = 1$
- $B_i = 0.05$
- $L_i = 0.7$
- $D_j \sim \text{Uni}(0.8\eta,\ 1.2\eta)$ where $\eta = \dfrac{|N| \times \frac{600}{200\times0.85}}{|M|}$
- $H = |M|$
- $|T| = \dfrac{\sum_{j\in N} Q_j}{\sum_{i\in M}(A_i \times r')}$ where $r' = \dfrac{1+L_i}{3}$

First experiment

In the first experiment, we run Gurobi to solve the each problem and recorded the optimality gap. Table 4 and Table 5 are the average optimality gap of 10 tests at 10 minutes and 30 minutes, respectively.

10 min			
$m\backslash n$	5	10	15
1	0	0.49884	0.72776
3	0	0.429365	0.48173
5	0	0.104444	0.493384

Table 4: the average optimality gap for 10min

30 min			
$m\backslash n$	5	10	15
1	0	0.481115	0.677687
3	0	0.39246	0.462128
5	0	0.030556	0.461142

Table 5: the average optimality gap for 30min

According to the Table 4 and Table 5, we observed that the average optimality gap over 10 tests for a certain number of orders decreased as the number of machines increased. A possible reason might be that the workload for each machine decreased, thus, it became easier to schedule. Also, the average optimality gap for a certain number of machines increased as the number of orders increased. It is straightforward that scheduling became complicated if the number of orders increased.

Second experiment

After the first experiment, we noticed that the optimality gap declined rapidly at the beginning, but it slowed down and needed more time to improve solutions. It may be worthless to spend such long time to get a small improvement in some cases. Hence, in the second experiment, we observed the declining rate of the optimality gap and tried to find an appropriate stopping time for solving different problem sizes by Gurobi.

For the fixed number of machines, the declining rate of the optimality gap would rapidly convergence according to the number of orders. The smaller the number of orders, the higher the decline rate of the optimality gap. On the other hand, for the fixed number of orders, the declining rate of the optimality gap would rapidly convergence according to the number of machines as well.

According to Figure 2 to 7, we observed that for calculation speed, the number of machines have a greater impact than the number of orders. We also found out that with the problem size between 1 to 5 machines and 5 to 15 orders, 10 minutes might be an appropriate stopping criterion.

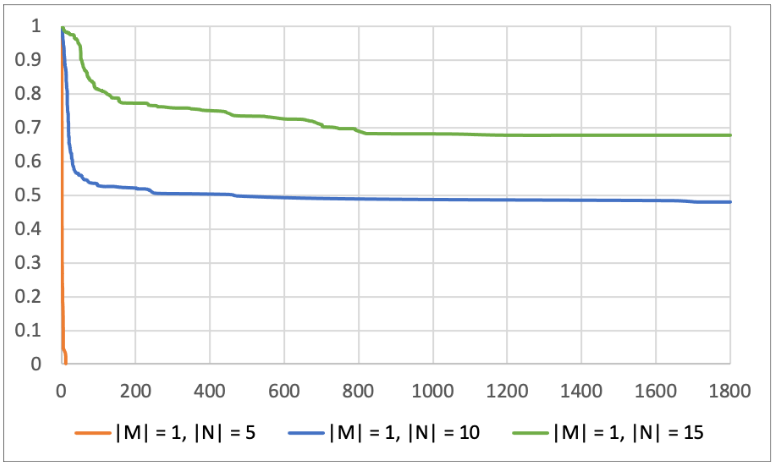

Figure 2: The optimality gap where $|M| = 1$

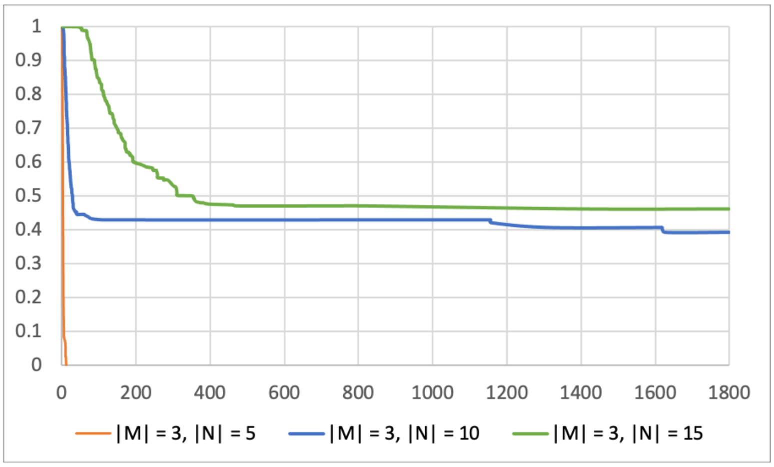

Figure 3: The optimality gap where $|M| = 3$

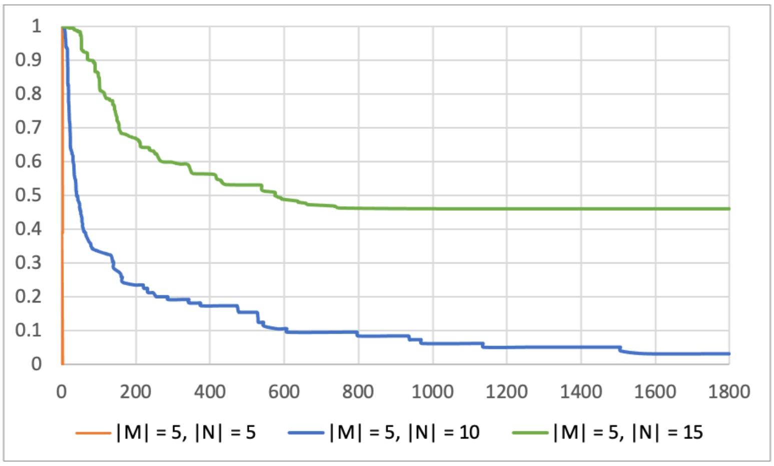

Figure 4: The optimality gap where $|M| = 5$

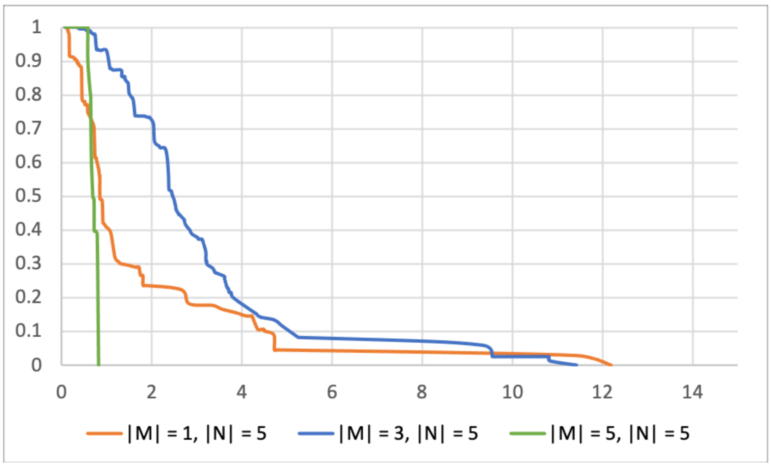

Figure 5: The optimality gap where $|N| = 5$

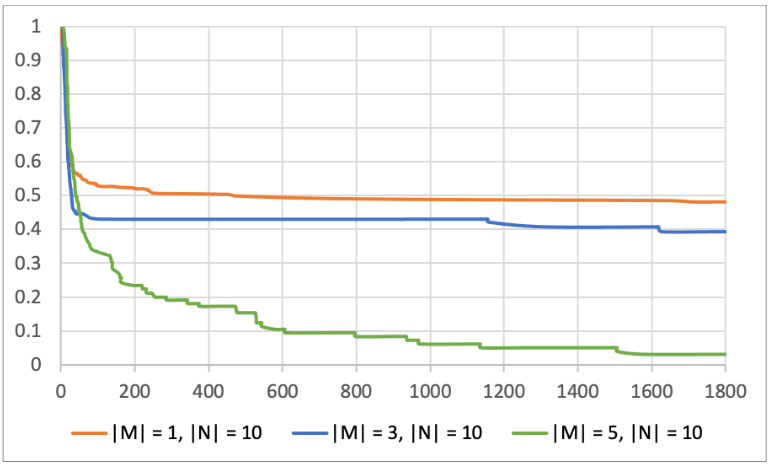

Figure 6: The optimality gap where $|N| = 10$

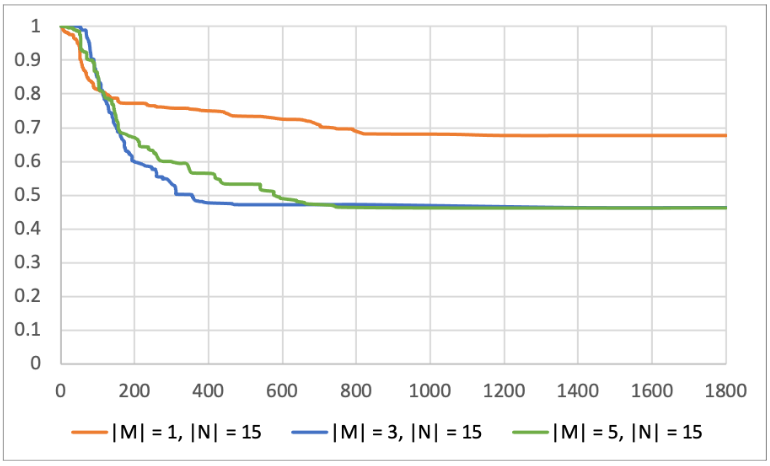

Figure 7: The optimality gap where $|N| = 15$

7 Conclusion

We have proposed an MIP model for the single-product unrelated-machine maintenance scheduling problem. It have considered the yield declining rate and maintenance resources limitation. Our objective is to generate a schedule with predictive maintenance which minimize the total order tardiness.

By the MIP model, we could get the optimal solution for maintenance scheduling problem, but it spent much time. Therefore, we have conducted the two experiments to discuss how the problem size impact the performance of MIP. The experiment have been conducted under 9 different problem sizes and for each problem size there are 10 tests. We observed that the optimality gap became large as number of orders increased or the number of machines decreased. Besides, by the experiments, We found that 10 minutes might be an appropriate stopping criterion for the Gurobi to solve our MIP model.

Because the scheduling problem is an NP-hard problem, solving by Gurobi seems impracticable for the large size problem. Therefore, future research directions involve proposing a heuristic algorithm to solve this problem more efficiently.

References

Cassady, C. R., & Kutanoglu, E. (2005). Integrating preventive maintenance planning and production scheduling for a single machine. *IEEE Transactions on Reliability*, *54*(2), 304–309. https://doi.org/10.1109/TR.2005.845967

Lee, C.-Y., & Chen, Z.-L. (2000). Scheduling jobs and maintenance activities on parallel machines. *Naval Research Logistics (NRL)*, *47*(2), 145–165.

Ullman, J. D. (1975). Np-complete scheduling problems. *Journal of Computer and System sciences*, *10*(3), 384–393.