Preventive Maintenance Scheduling in Semiconductor Manufacturing

IE3100M Systems Design Project | Department of Industrial Systems Engineering & Management

NUS Supervisor: Professor Ye Zhi Sheng

Industry Supervisors: Nikhil Mukkawar | Hui Xin Xu | Yang Hao

Group Members: Clement Soo U-Jie | Teo Jia En Jewel | Muhammad Alisudais Bin Suhaiemi | Shirely Phua Meng Geok

National University of Singapore

Problem Overview

Preventive Maintenance (PM) :

In semiconductor manufacturing, the reliability of equipment has significant implications on production yield, production time reduction and cost reduction. Regular maintenance of equipment is important as it keeps equipment running and prevents any costly unplanned downtime from unexpected equipment failure.

High Costs:

Fabrication equipment is extremely sophisticated and expensive. They require extensive calibration and maintenance, thus underscoring the importance of preventive maintenance

Good Scheduling:

A good PM schedule can thus increase equipment availability by trading off between "planned" unproductive downtime due to PM and the risk of much costlier "unplanned" downtime due to equipment failure, which can cause major disruptions in the manufacturing process.

Objective:

The objective of this systems design project is to develop a PM scheduler that recommends a PM schedule for the next 14 days according to Micron's given constraints and requirements.

Scope:

The scope of this project is limited to the fabrication equipment in the area in Fab 10N at Micron's Woodland's compound, as determined by Micron. The constraints and requirements that the PM scheduler needs to meet are as following.

Requirements

1. Scheduling Preventive Maintenance for Equipment Before Reaching Critical Limit
2. Ensuring Wafer Target Production is Met For All Stations
3. Accounting for Equipment Green-To-Green Time
4. Manpower Constraint for Maintenance
5. Checking for Availability to Undergo Maintenance
6. Exception for Equipments Required for Priority Lots
7. Account for Wafer Production Process Constraints

Methodologies

Mixed Integer Programming Method

The PM scheduling problem can be modelled as a mixed-integer programming (MIP) problem. The objective function of the MIP model is to maximize wafer production.Python library PuLP was studied and used to implement the MIP problem.

Objective Function:

$$\text{Max wafer production} = \sum_{t=1}^{T}\sum_{j=1}^{N}\sum_{i=1}^{n_j} \lambda_{i,j}(t)\, C_{i,j}(t) + \mu_j(t)\, M_j(t)$$

Indices:

i	chamber
j	mainframe
t	Time period (t= 1, …, T)

Decision Variables:

Variable	Variable Definition
Binary decision variable for PM on chamber i (in mainframe j) in period t	$a_{i,j}(t)$ = {1: do PM in period t, 0: don't do PM in period t}
Binary decision variable for PM on mainframe j in period t	$b_j(t)$ = {1: do PM in period t, 0: don't do PM in period t}
Availability of chamber i (in mainframe j) in time period t	$C_{i,j}(t)$ = {1: available, 0: unavailable}
Mainframe j's availability in time period t	$M_j(t)$ = {1: available, 0: unavailable}

Parameters:

T	Number of time periods in planning horizon
N	Number of mainframes considered
n_j	Number of chambers in mainframe j
$w_{i,j}$, $u_{i,j}$	Time window [min, max] associated with PM on chamber i in mainframe j
w_j , u_j	Time window [min, max] associated with PM on mainframe j
$v(t)$	Number of equipment that can undergo PM in time period t
$\lambda_{i,j}(t)$	Wafer production of chamber i in mainframe j in time period t
$\mu_j(t)$	Wafer production of mainframe j in time period t
G_j	Aggregate G2G time for mainframe j
$H_{i,j}$	Aggregate G2G time for chamber i in mainframe j
Z	Wafer production target to meet for the next 14 days

Constraints

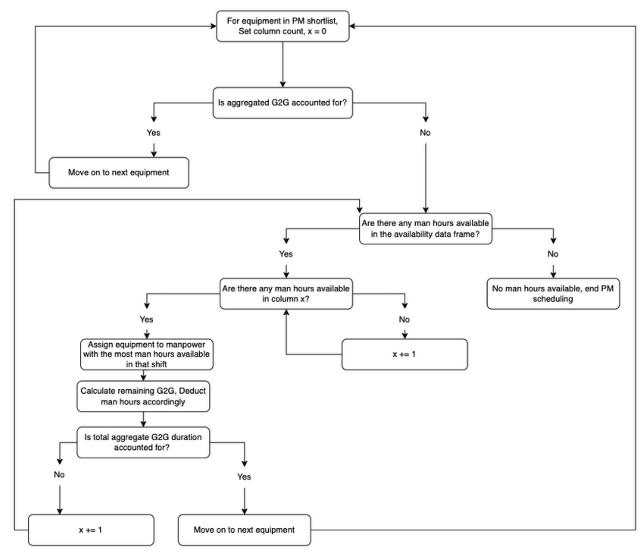
Constraint	Constraint Formulation in Model
PM to be completed for chamber i in mainframe j	$\sum_{t=w_{i,j}}^{u_{i,j}} a_{i,j}(t) = 1 \quad \forall\, i,j$
Availability of chamber i in time period t	If any $a_{i,j}(t) = 1$, then $C_{i,j}(t), C_{i,j}(t+1), \ldots, C_{i,j}(t+H_{i,j}) = 0$
If entire mainframe is unavailable, then all chambers in that mainframe will also be unavailable	$C_{i,j}(t) \le M_j(t) \quad \forall j,t,i$
PM to be completed for mainframe j	$\sum_{t=w_j}^{u_j} b_j(t) = 1 \quad \forall\, j$
Availability of mainframe j in time period t	If any $b_j(t) = 1$, then $M_j(t), M_j(t+1), \ldots, M_j(t+G_j) = 0$
Wafer production target to meet for the next 14 days	$\sum_{t=1}^{T}\sum_{j=1}^{N}\sum_{i=1}^{n_j} \lambda_{i,j}(t)\, C_{i,j}(t) + \mu_j(t)\, M_j(t) \ge Z$
Manpower constraint	$\sum_{j=1}^{N}\sum_{i=1}^{n_j} a_{i,j}(t) + b_j(t) \le v(t) \quad \forall\, t$

Automated PM Scheduling Via Heuristic Algorithm

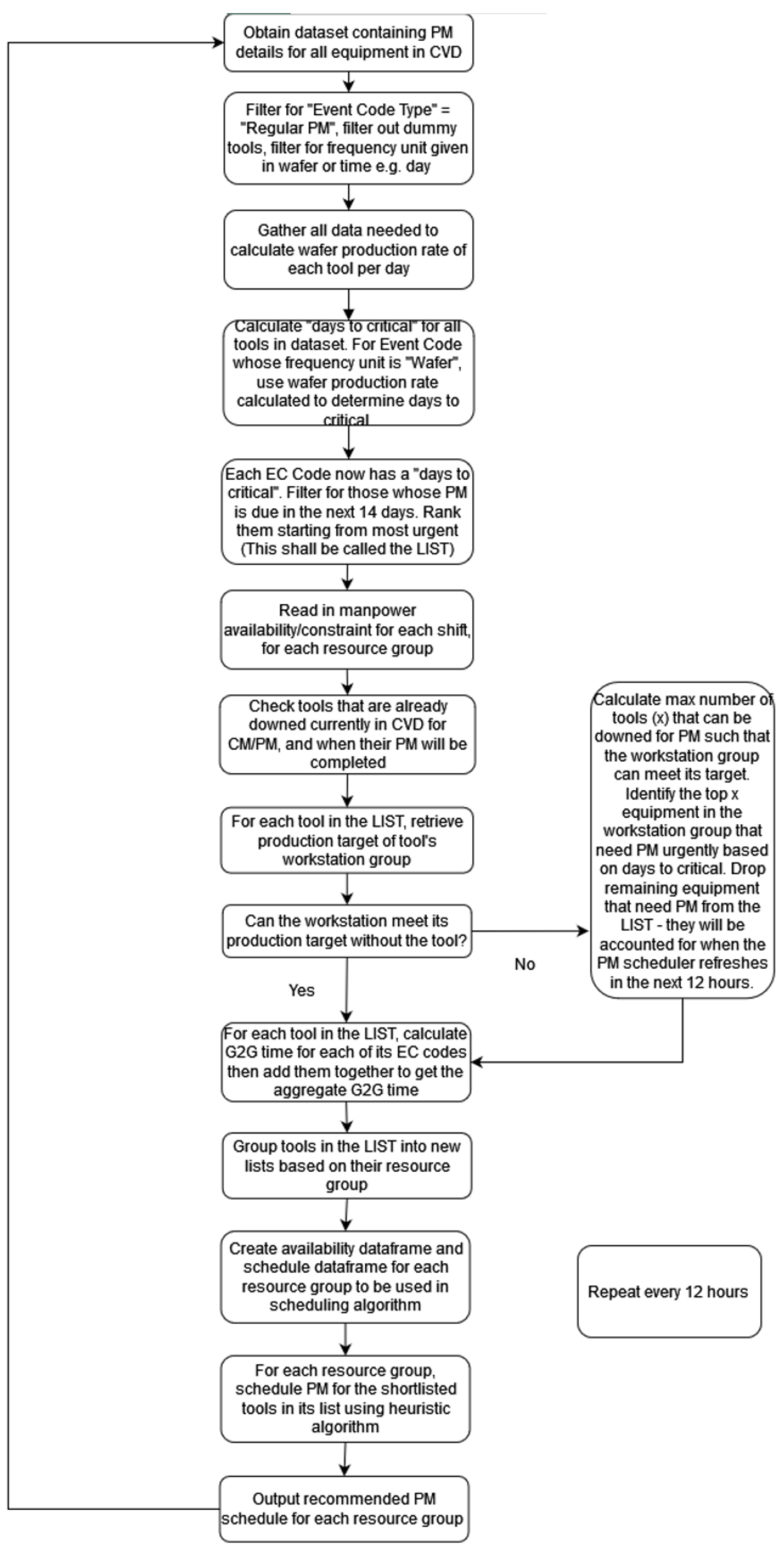
The second approach develops the PM scheduler by automating requirement checks and scheduling PM using a heuristic algorithm. It is also implemented in Python using the Python libraries pandas, numpy, math and random. The proposed heuristic algorithm is given below. For each resource group, it iterates through all the shortlisted PM equipment by urgency

An availability data frame is constructed for each resource group. To illustrate, the manpower available, the maximum number of manpower in a shift is 4, a total of 4 rows will be constructed for the resource group. Next, since there are 2 shifts a day for the next 14 days, 28 columns are constructed. Each cell in the data frame contains the number of man-hours available. Even Column represents Day Shifts. Odd Column represents Night Shifts

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12
1	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12
2	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12
3	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12	0	12



Implementation



1.Data Gathering

Data would first have to be collected and processed before input into the scheduler.

The data used was extracted from Micron's database and processed to calculate the remaining days to critical for each workstation.

Required data	Remark
EQUIP ID	Equipment's ID that is unique to each equipment and is needed as a primary key to retrieve its Workstation Group Name from another dataset
WS NAME	Equipment's workstation name that is needed as a primary key to retrieve G2G data
EVENT CODE	Name of Event Code that is needed to retrieve its G2G duration from G2G dataset
EVENT CODE TYPE	Only Event Codes with Event Code Type set to "Regular PM" are considered for PM scheduling
FREQ UNIT	Only Event Codes with frequency unit set to "Hour", "Day", "Week", "Month", "Year" or "Wafer" are considered for PM scheduling
REMAINING COUNT TO CRITICAL	Remaining count to critical is needed to calculate the number of days for the equipment to reach the Critical Limit for that Event Code. The unit for remaining count to critical is the same as the frequency unit stated. It is given directly in *Datanav* but can also be calculated by subtracting Current Count from Critical Limit, both of which are also given in *Datanav* and are terms explained earlier in Section 3.2.1.

2.Shortlisting Equipment that need PM

A list of workstations that are critical within 14 days will then be obtained. This list is then sorted based on urgency, allowing the scheduler to down more immediate cases first.

3.Meeting Wafer Production Targets

Each workstation group has a wafer production target that should still be met despite some tools being down. A check has to be carried out to ensure a tool can be down for maintenance without compromising the production target.

Firstly, the wafer production target for the next 14 days is calculated for each workstation group.

$$\text{Wafer production target for next 14 days} = (\text{“MOVES REQUIRED 24 HR” of workstation group}) * 14$$

Before calculating the output capacity of the workstation group after downing some of the equipment.

$$\text{Wafer production for next 14 days without shortlisted equipment} = \left(\frac{\text{Daily wafer production of workstation group}}{\text{Number of In Service Equipment}}\right) * (\text{Number of In Service Equipment} - \text{Count of shortlisted Equipment marked for PM}) * 14$$

Thus, the maximum number of equipment that can undergo maintenance is

$$\text{Maximum number of equipment that can undergo PM} = \text{Number of In Service Equipment} - \frac{\text{Wafer production target} * \text{Number of In Service Equipment}}{\text{Wafer production of workstation group without all shortlisted equipment}}$$

4.Calculating Aggregate G2G

As there are different event codes indicating differing maintenance types, an aggregated G2G was calculated based on every event code that is available for each workstation.

5.Accounting for Equipment that are Currently Down

The scheduler needs to take into account equipment that are currently down to ensure that manpower availability is accurate for each shift. A list of workstations that are currently down is first obtained and their respective G2G times stored for input into the scheduler.

6.Scheduling Preventive Maintenance

The PM scheduler will schedule PM for each resource group separately as each resource group is responsible for different workstation groups.

Firstly, each shortlisted equipment is grouped by their respective resource group. Next, the equipment that are currently down, are stored in a new dictionary with the shortlisted equipment. The equipment that are currently down is added infront of the shortlisted equipment such that the PM scheduler will account for them first.

An availability dataframe is then created for each resource group to be used by the heuristic algorithmn. The dataframe is created based on the manpower availability for each shift and resource group. A scheduling dataframe is also created for each resource group that is initially empty, but will be populated by the PM scheduler. Thus with both dataframes ready, the PM scheduler can now use the heuristic algorithm to plan a maintenance schedule for each respective resource group.

Conclusion

Automated PM Scheduling Via Heuristic Algorithm:

The most important benefit offered by Automated PM Scheduling is that it is not constrained by any feasibility condition that the MIP approach is known to have and will always recommend a PM schedule. For the MIP approach, whenever the scheduling problem formulated is infeasible or cannot be solved by the optimization algorithm and no optimal solution can be found, the PM scheduler cannot output a PM schedule at all. When this happens, the PM scheduler provides no practical benefit and is no longer useful to Micron.

The second approach was selected due to the limitations of the first approach, being a MIP model, being prone to infeasible solutions. This allows a scheduling solution that is able to cater for each requirement as it automates the checks for each requirement defined by Micron for PM scheduling as well as schedules PM using a heuristic algorithm.

Future Directions

As the model is static in nature, outputting a recommended schedule given data, work could be done such that is dynamic and automated. APIs can be created that have live access to necessary datasets and output a schedule that is able to refresh themselves every 12 hours. A dashboard can also be implemented that can display the recommended schedule for convenience