COSMOS Intelligent Layout Planning Solution
Development of Facilities
Capex Optimization Module

Department of Industrial Systems Engineering and Management

AY 2021/2022 Systems Design Project (Group 12) Team Members:

Elle Lee Xin Yu | Rahul Gupta | Tan De Ming | Zhang Lizheng

Department Supervisor: A/P Chew Ek Peng **Industry Supervisor**: Mr Michael Agung Pradhana

Problem Overview

Add a module in Micron's intelligent layout planning solution **(COSMOS)** to comprehend facilities' basebuild availability to determine the optimal placement of tools when they arrive in the fab. The **project scope** can be broken down into:

1. Liaising with the facilities team to **create a data pipeline** for facilities basebuild availability and tool utility requirements
2. **Develop an optimisation algorithm** to minimise facilities capital expenditure (CAPEX) or lead time by using utility cost functions and lead time to build laterals for utilities

Current Situation

✘ Current database is **not easily readable**

✘ **No automated system** to plan factory layout. Planning is currently done **manually** by Micron engineers

Time is spent **inefficiently** to interpret data and **inconsistencies** may arise due to human error

Key Objectives

✔ Create a **data pipeline** to transform data to be used in Python

✔ Develop an **optimisation model** that fits Micron's needs

✔ **Enhance accuracy** of layout planning

✔ Allow **scalability** of the proposed model

Key Skillsets

✱ Systems Thinking

✱ Operations Research

✱ Linear Programming

✱ Project Management

✱ Python and Excel (data cleaning and creation of model)

✱ Gurobi Optimiser

Methodology

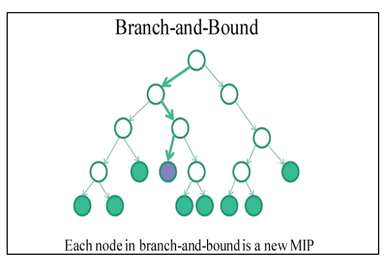
Data Pipelining

Before building the optimisation algorithm, **extensive preliminary work** was done. It involved data cleaning and rigorous data transformations. From these datasets, the available number of utilities at each location and required utilities for each tool are found. The cost of each utility is assumed to be constant.

Name of Column	Column Details
Name of Utility	Type of Utility
Lateral/VMB	Fab location - Level - Gridline - Service - Type of EQ - Index
LPOC	Fab location - Level - Gridline - Service - Type of EQ - Index - Connection Index
VMP	Lateral of Utility used to link the Utility to Tool
MiCap	Index of Tool
Workstation	Type of Tool
LID	Location ID (address of a tool)

Mixed Integer Programming

MIP problems consists of constrained variables as integers as well as some acting as non-integers. MIP problems are solved using the **branch and bound** methods of enumeration. The Gurobi optimiser is used for this model.

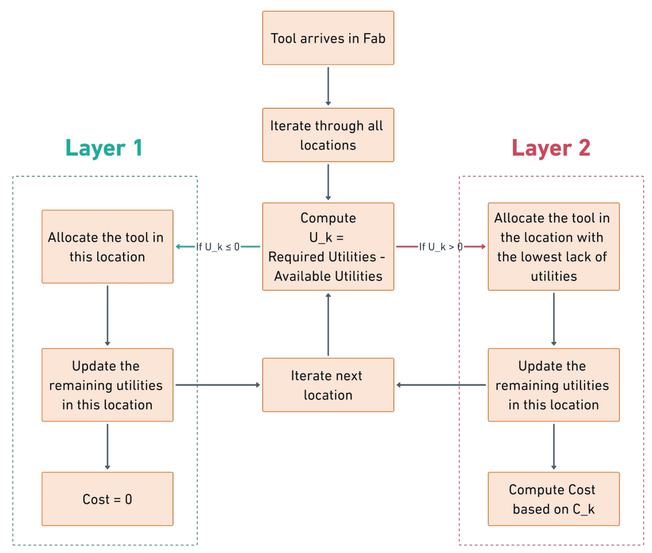


MIP Model Formulation

Sets	I = Total number of Tools J = Total number of Locations K = Total number of Utilities
Parameters	r_{ik} = Required units of *Utility k* for *Tool i* a_{jk} = Available units of *Utility k* at *Location j* c_k = Cost of 1 unit of *Utility k*
Decision Variables	$x_{ij} = \begin{cases} 1, & \text{if Tool } i \text{ is assigned to Location } j \\ 0, & \text{otherwise} \end{cases}$
Objective Function	$\min Z = \sum_i \sum_j \left(x_{ij} \times c_k \sum_k \max\{(r_{ik} - a_{jk}), 0\} \right)$
Constraints	$\sum_j x_{ij} = 1 \quad \forall i = 1, \ldots$ (1) $\sum_i x_{ij} \leq 1 \quad \forall j = 1, \ldots$ (2)

Greedy Heuristic Programming

This model acts as an **incumbent** to the MIP model. In a scenario, where the MIP model is not able to solve the problem for an **optimal solution**, **greedy heuristic algorithm helps** to find the **most feasible solution**.



Comparison of Design

Mixed Integer Programming

✔ Able to allocate all tools in fab simultaneously

✔ Able to find the most optimal solution

✘ Not able to update preset available utilities after each allocation of tool

✘ Might run into memory error if problem is being scaled up

Greedy Heuristic Programming

✔ Able to replicate real-life decision making process through sequential allocation of tools

✔ Able to update preset available utilities after each allocation of tool

✘ Not able to find the most optimal solution

✘ Might result in longer run time when scaled up

Results

Mixed Integer Programming

Tools Allocation

Tool Name	Allocated Location in the Fab
Tool A1	F24E
Tool A2	F20F
Tool B1	F04H
Tool B2	F32E
Tool C1	F20D
Tool C2	F24F
……	……
Total Cost	**2195**

Greedy Heuristic Programming

Tools Allocation

Tool Name	Allocated Location in the Fab
Tool A1	**F04D**
Tool A2	F02N
Tool B1	F18D
Tool B2	**F04D**
Tool C1	F02S
Tool C2	F06J
……	……
Total Cost	**2245**

Individual Cost of Allocation

Tool Name	Location	Cost
Tool A1	**F04D**	**65**
Tool A2	F02N	70
Tool B1	F18D	70
Tool B2	**F04D**	**70**
Tool C1	F02S	50
Tool C2	F06J	50

✱ Tools A1 and B2 are in Location F04D

Recommendations

- If the information on **all tools** (including future tools) entering the fab are available, then the **MIP model** should be chosen.
- If the number of tools incoming **exceed** the number of available locations, and sequential allocation is needed, the **greedy heuristics model** should be chosen.
- Depending on Micron's needs, they can choose to minimise either cost or lead time.

Future Steps

- Create an **updated cost function** that can account for the actual utility cost and lateral building cost. These costs are subject to global supplies of chemicals for utilities and materials for laterals. This allows the model to **mimic the real-life** context more accurately.
- **Sharing of utilities** across different locations. This can occur when a particular location has insufficient utilities for the tool placed there. This allows for more combinations and **introduces a trade-off** between the cost of utilities and cost of building laterals.

Conclusion

✔ Created **two separate models** (minimise cost or lead time) for Micron to automate their tool allocation as part of COSMOS

✔ **Reduced time spent** in layout planning from as high as 72 hours to **less than an hour**, depending on the size of the batch of incoming tools

✔ **Removed obstacles** of human error and inconsistent allocations

✔ Proposed models can be **scaled up** to be used in all of Micron's fabrications