Reinforcement Learning- Driven PCB PDN Design Automation

Haran Manoharan, Chulsoon Hwang

Hybrid IBIS Summit at IEEE EMC+SIPI 2025

Raleigh, North Carolina

August 22,2025

Outline

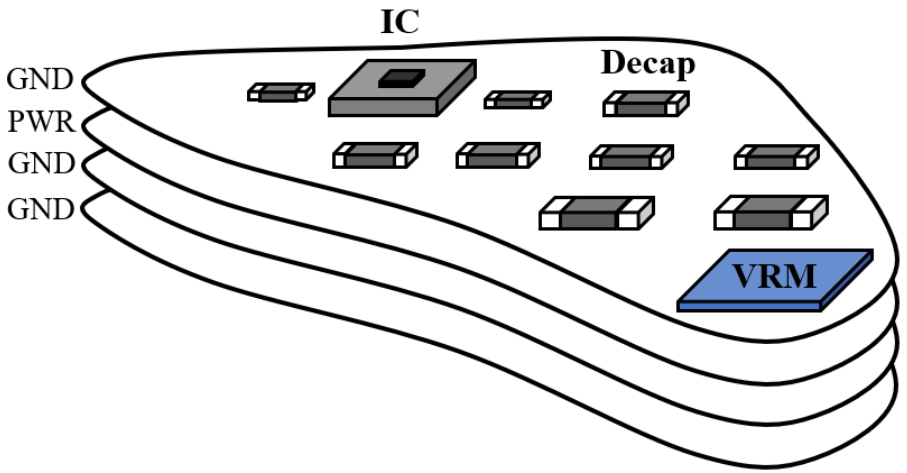
- Introduction
- Post-Layout Decap Optimization
- Pre- Layout Decap Layout Synthesis
- Pre-Layout Power Plane Shape and Stackup Synthesis
- Future Works

POST-LAYOUT DECAP OPTIMIZATION

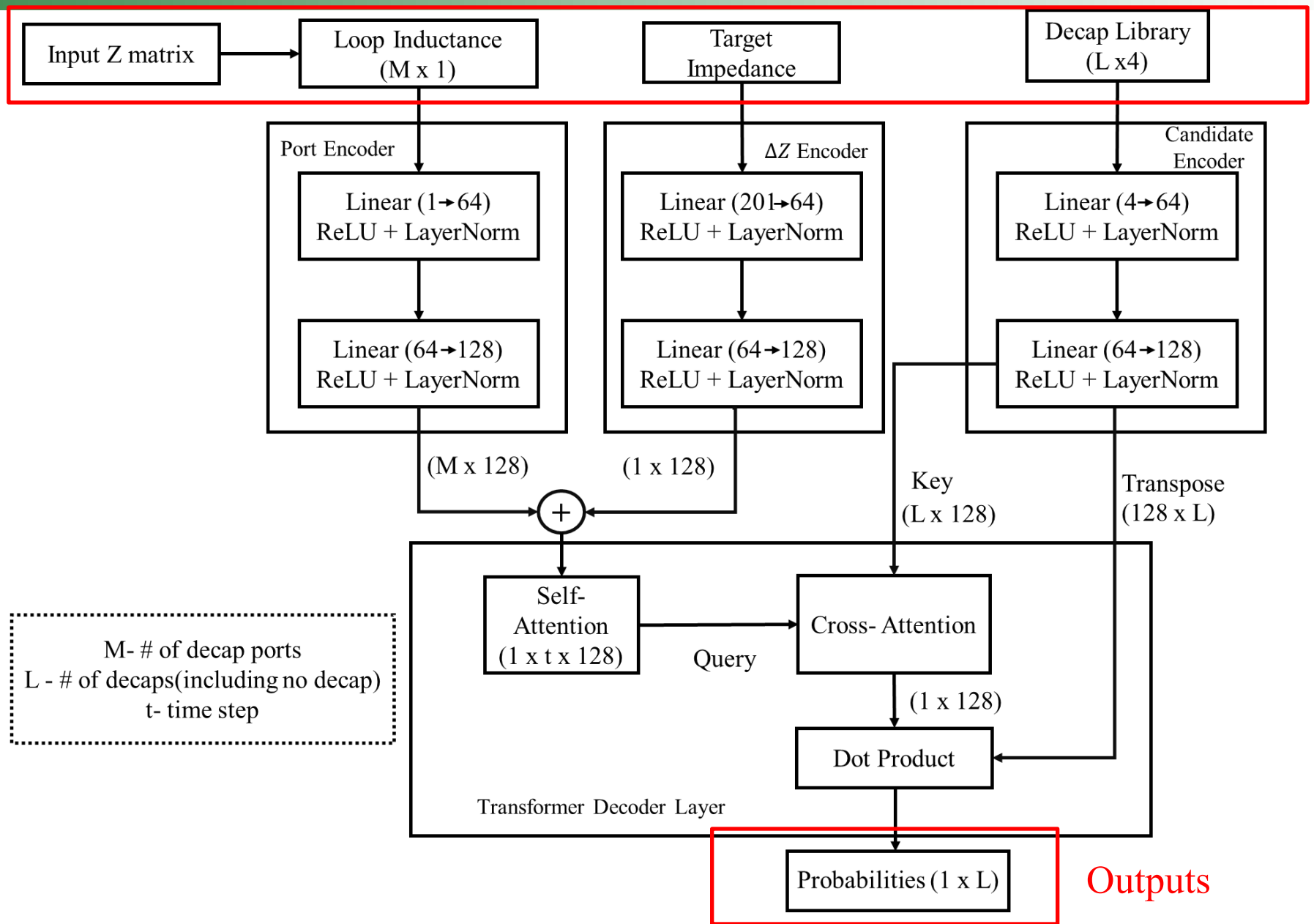
Introduction

- Decoupling capacitor optimization is a “hard combinatorial” problem.

- If we have 100 decap ports and 10 decaps in decap library the total # of combinations is 1×11^{100} combinations trying brute force is practically impossible.

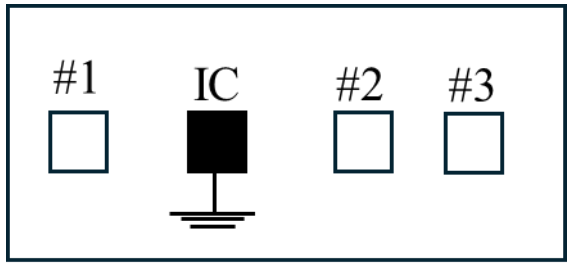


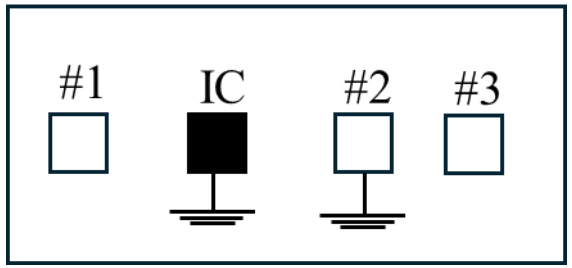
Model Architecture



Port Priority

- Ports are prioritized based on their loop inductance with respect to the IC port - lower inductance ports are ranked higher for decap placement.
- At each step, the port with the minimum loop inductance is shorted, and the impedance matrix is updated to reflect its connection.

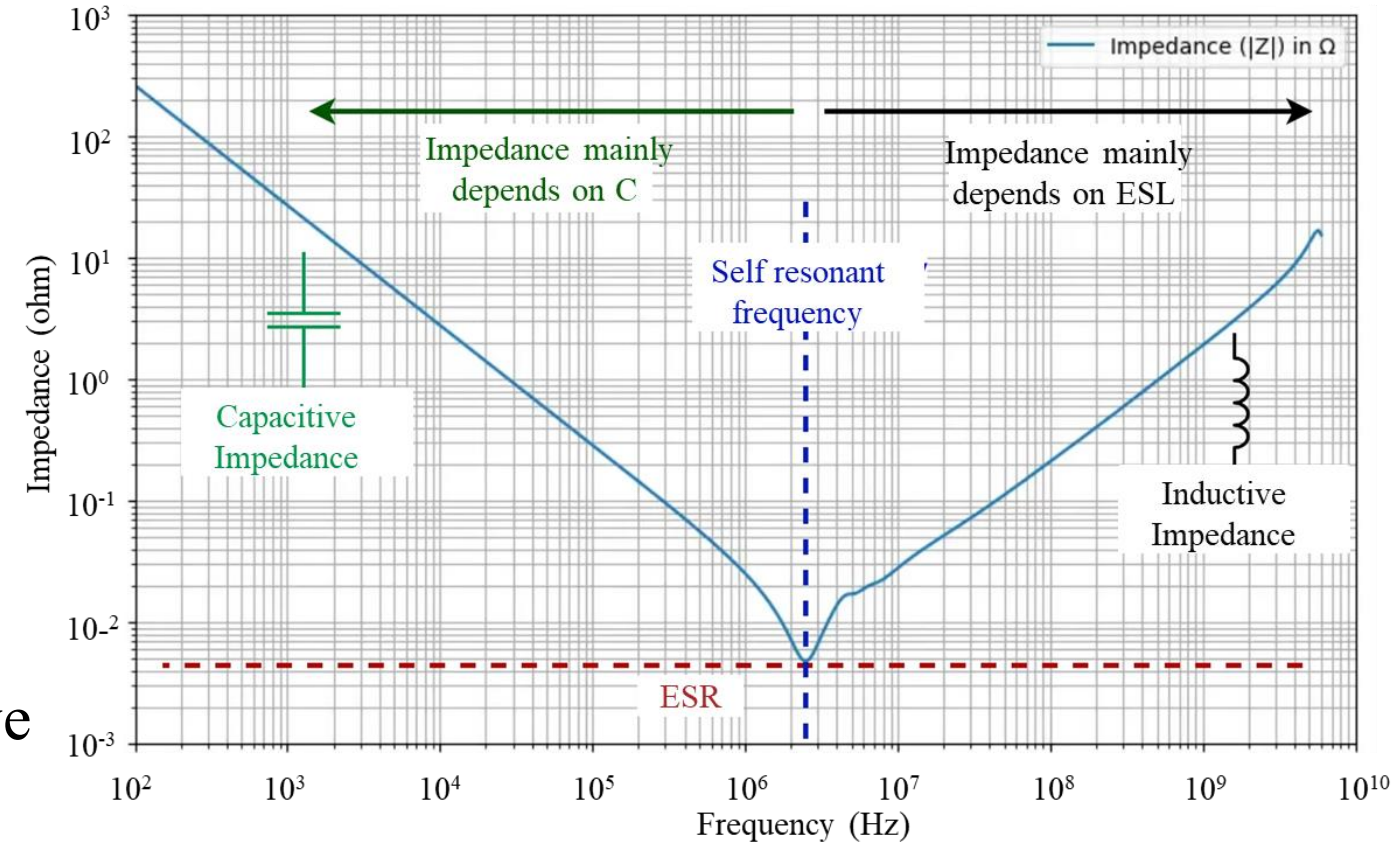




Decap Library Features

We characterize each decap by 4 distinct features

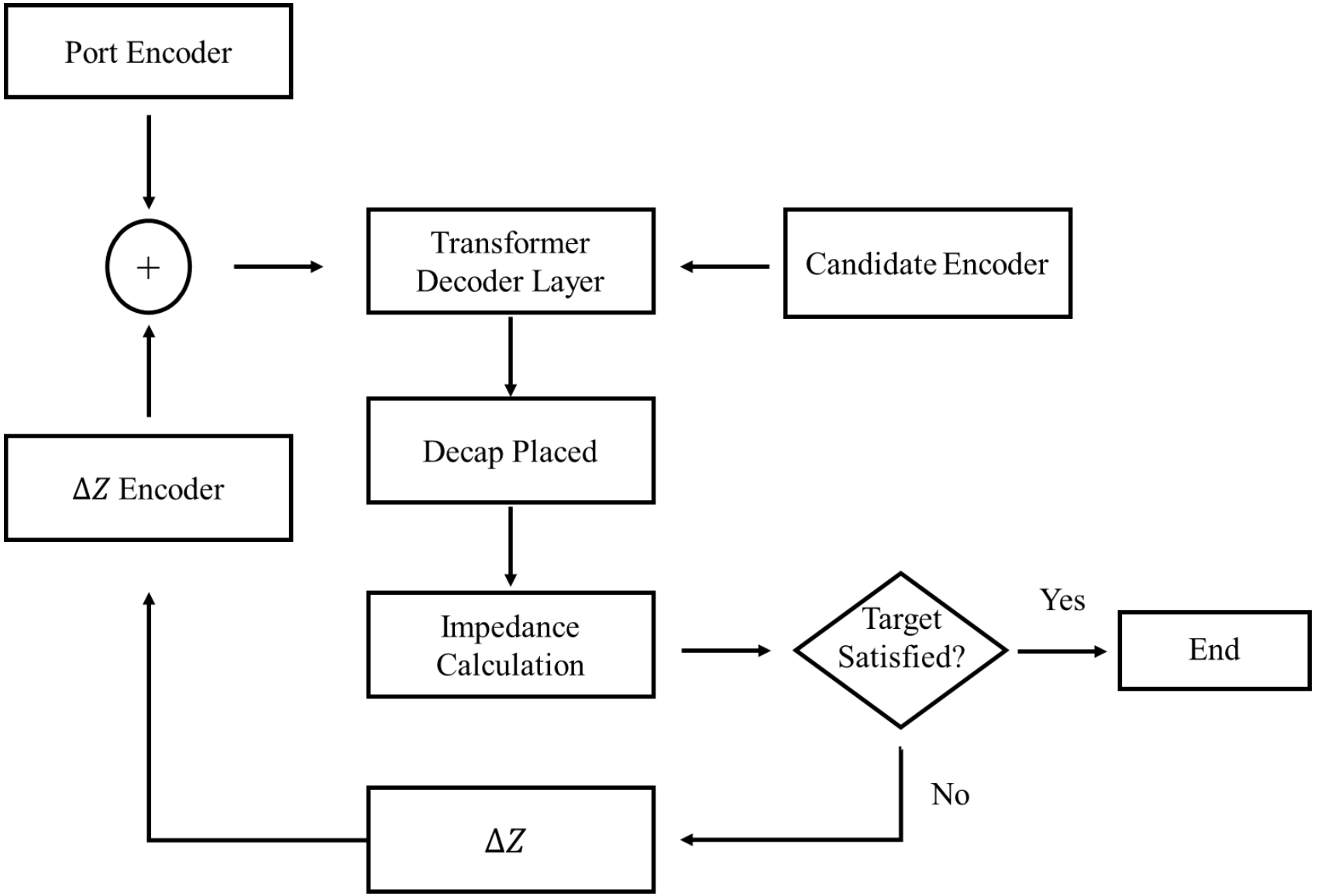
- ESL
- ESR
- Capacitance
- Package Size



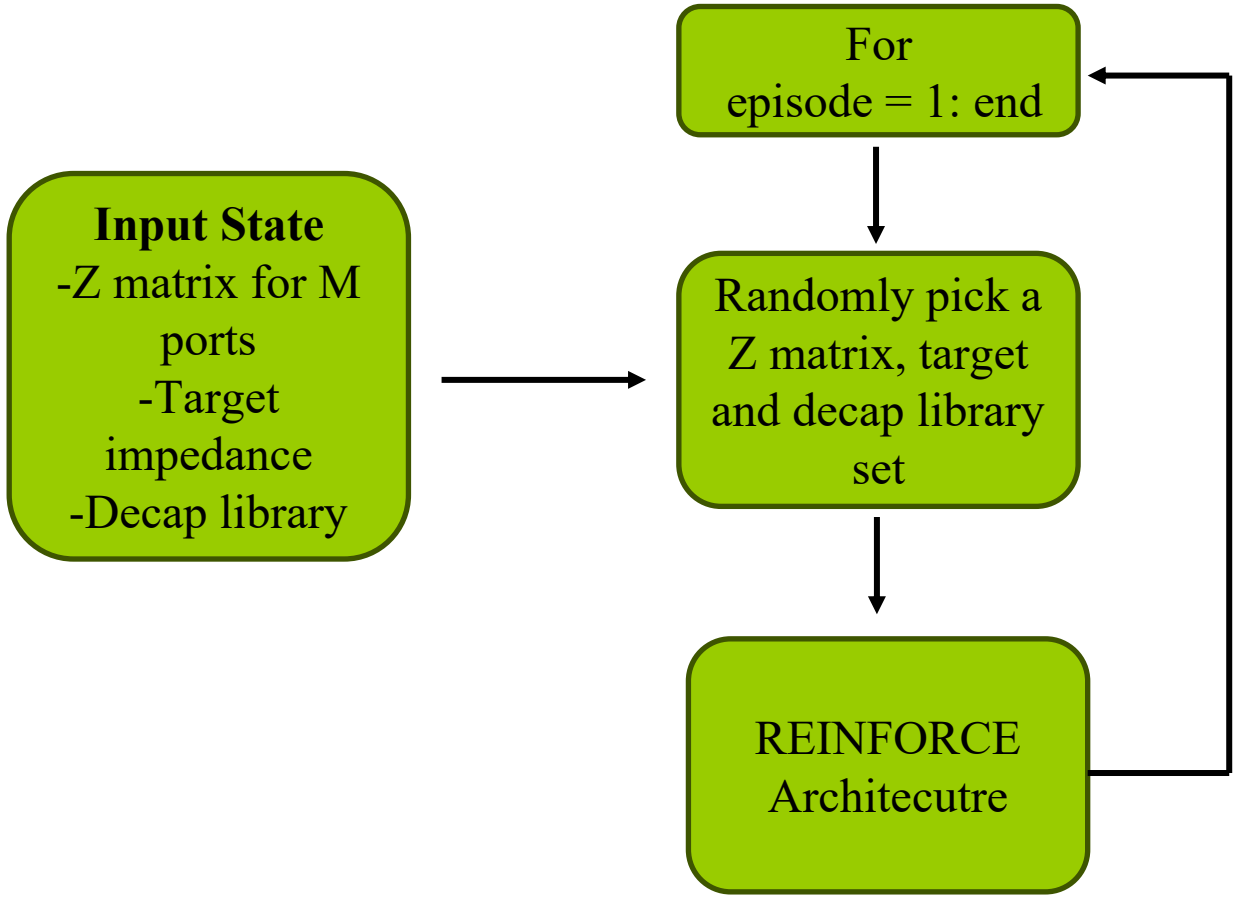
Data preprocessing

- From the impedance profile of decap we get the ESL, ESR and Capacitance value and Package size from the datasheet

Sequential Placement Loop



Training Architecture and Training Dataset

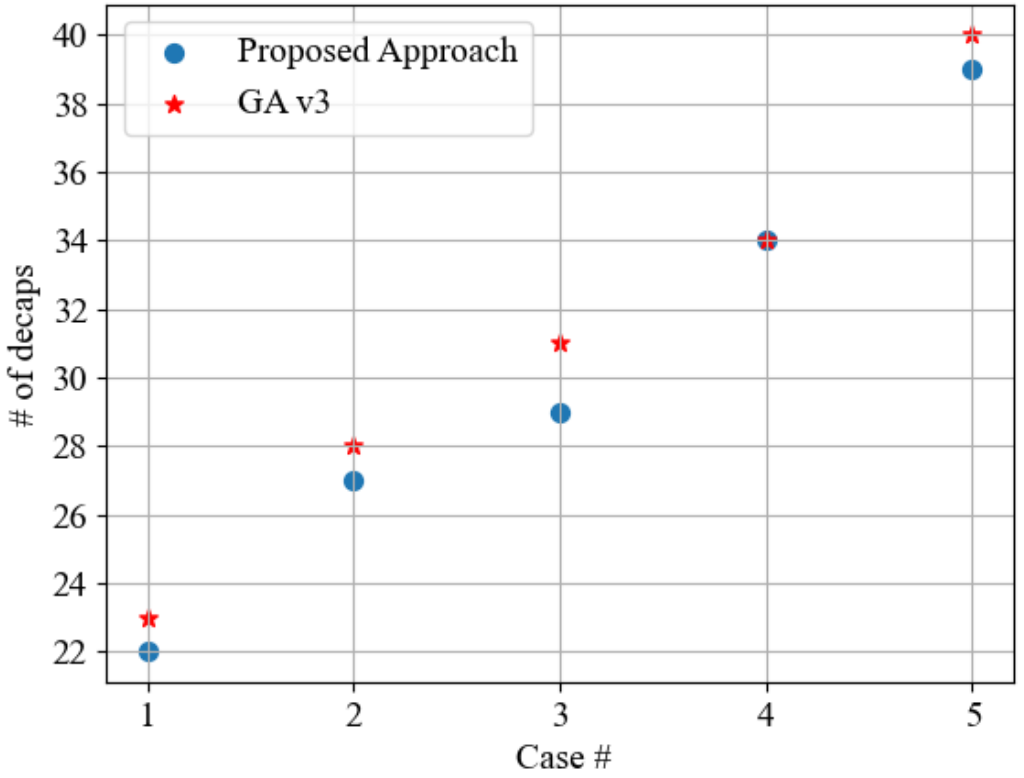


Training Architecture

- Created different configurations of decap board cases using PDN modelling method [1].
- Total number of cases – 10000 (total # of decap ports ranging from 25 to 75).
- Target impedance –sweeped from loose to strict target impedance. (1mohm to 1 ohm)
- Decap library –different # of decaps were used (total decaps from Murata ~1000)
- Total training time ~1 day

Testing Generalizability

- Decaps from a different vendor
- 50 port decap board case
- Sweeped across different target impedances

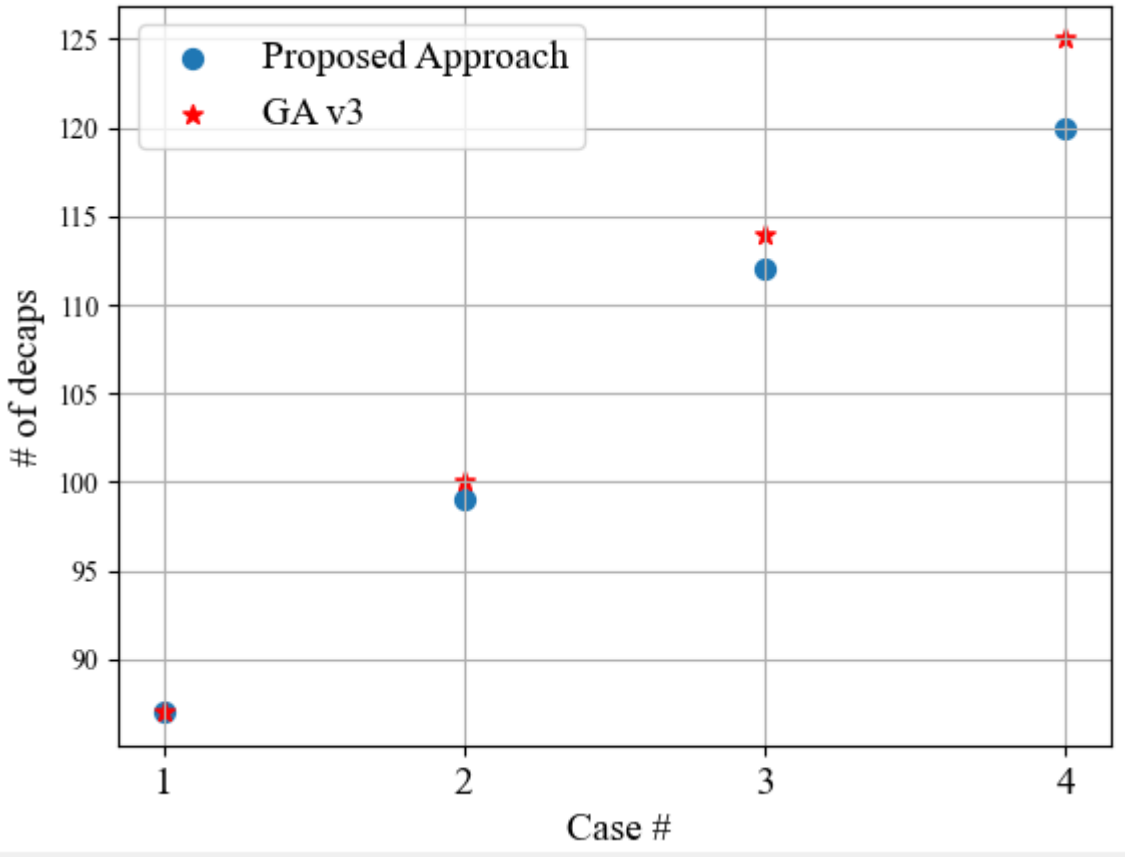


#	Proposed Approach		GA v3	
	# of decaps	Time taken	# of decaps	Time taken
1	22	300 ms	23	459 s
2	27	320 ms	28	469 s
3	29	400 ms	31	474 s
4	34	490 ms	34	504 s
5	39	610 ms	40	544 s

Testing Scalability

- Decaps from a different vendor
- 150 port decap board case
- Sweeped across different target impedances

#	Proposed Approach		GA v3	
	# of decaps	Time taken	# of decaps	Time taken
1	87	0.76 s	87	59 min
2	99	0.9 s	100	65 min
3	112	1.1 s	114	78 min
4	120	1.4 s	125	90 min

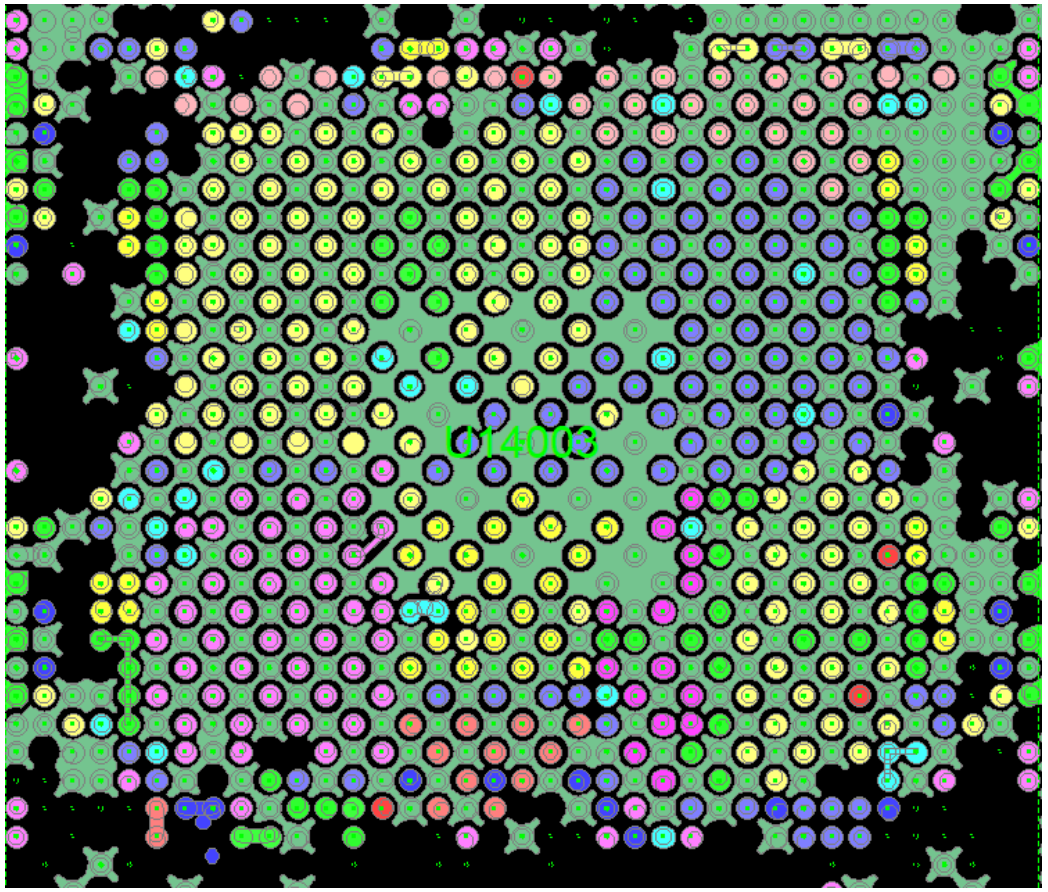


PRE-LAYOUT DECAP PLACEMENT SYNTHESIS

Introduction

- Modern SoCs have multiple power domains to optimize energy efficiency and performance by selectively controlling power to different components and functions within the chip.

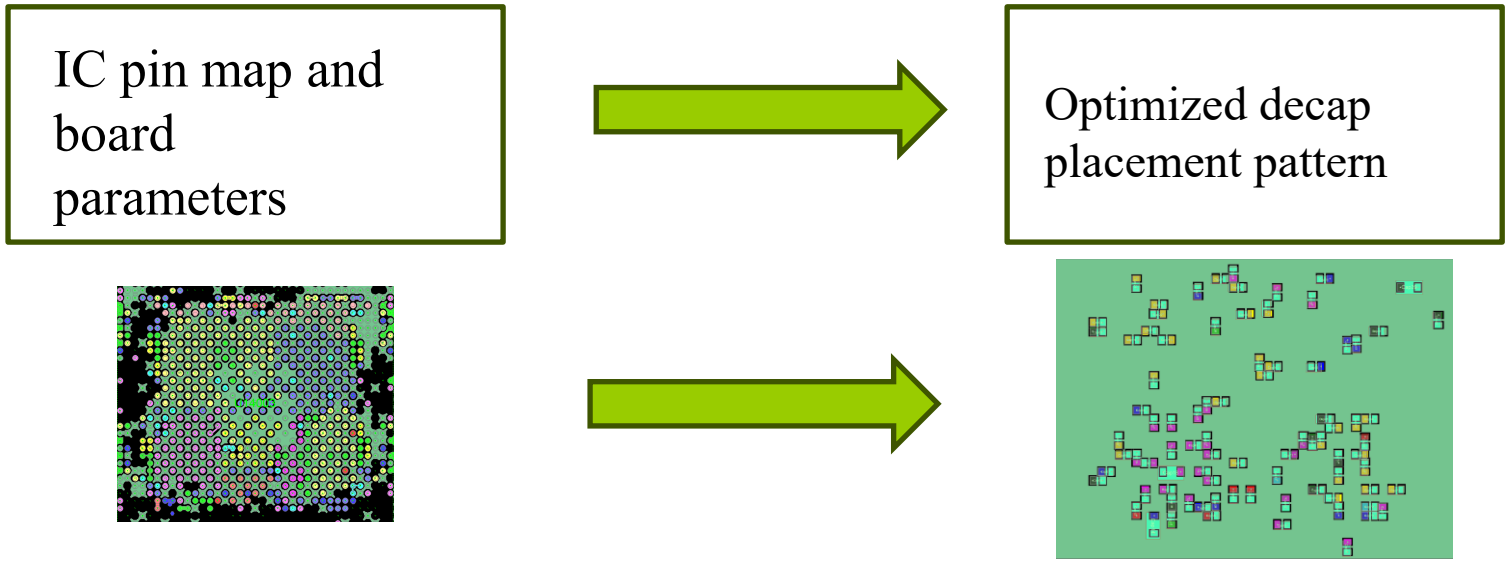
- Optimization of both placement and value for all the power domains simultaneously is a challenging task.



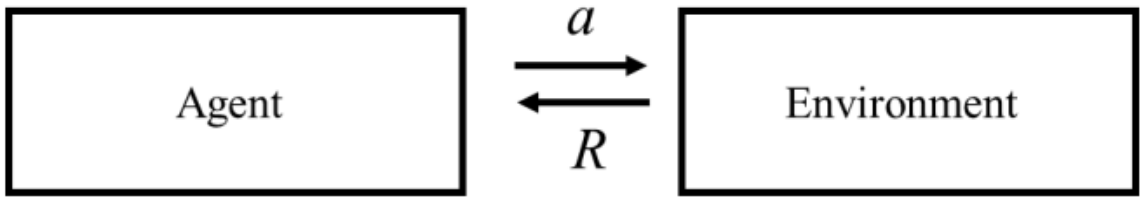
IC Pin Map

Goal

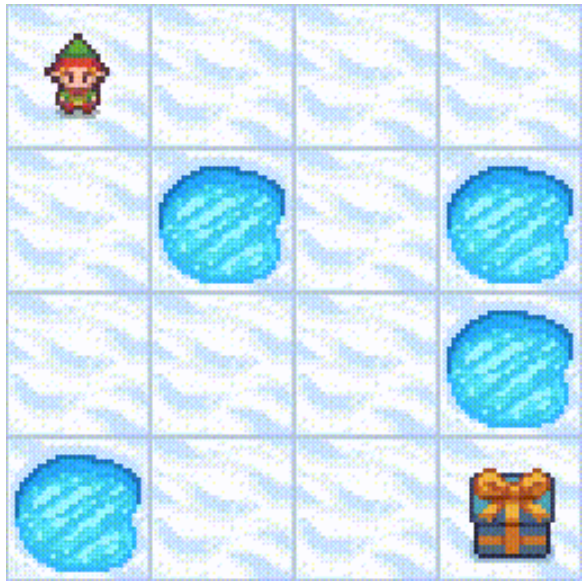
- An ML algorithm to automatically design and optimize the PDN layout for multiple power domains sharing the same layout area



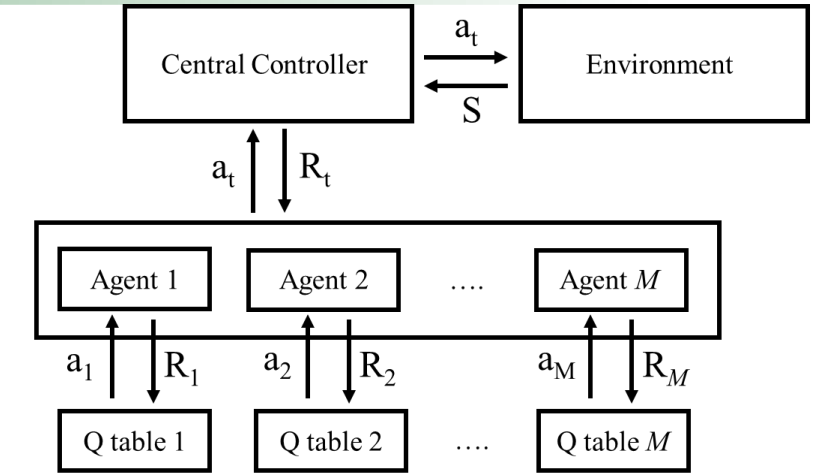
Single vs Multi-Agent Reinforcement Learning

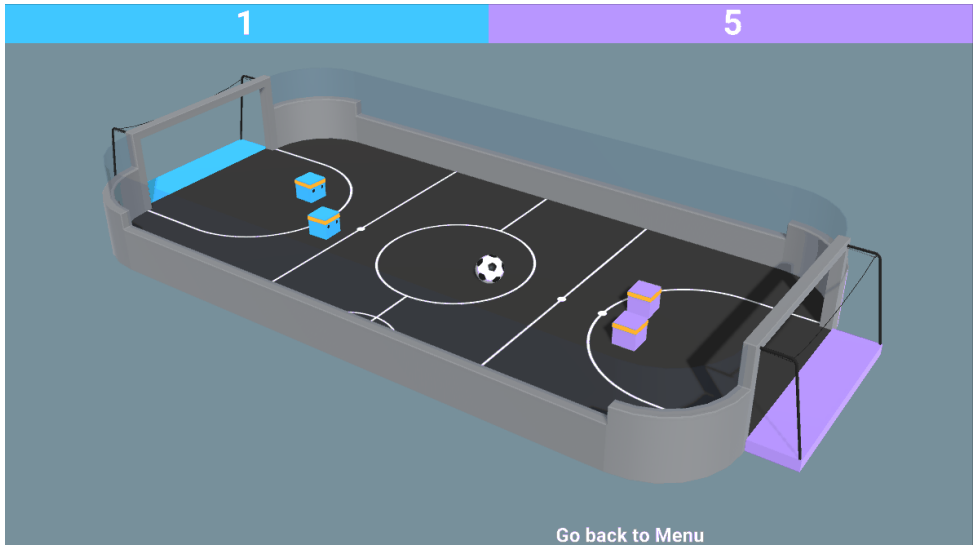


Single-Agent RL Framework



Single- Agent RL Example[1]

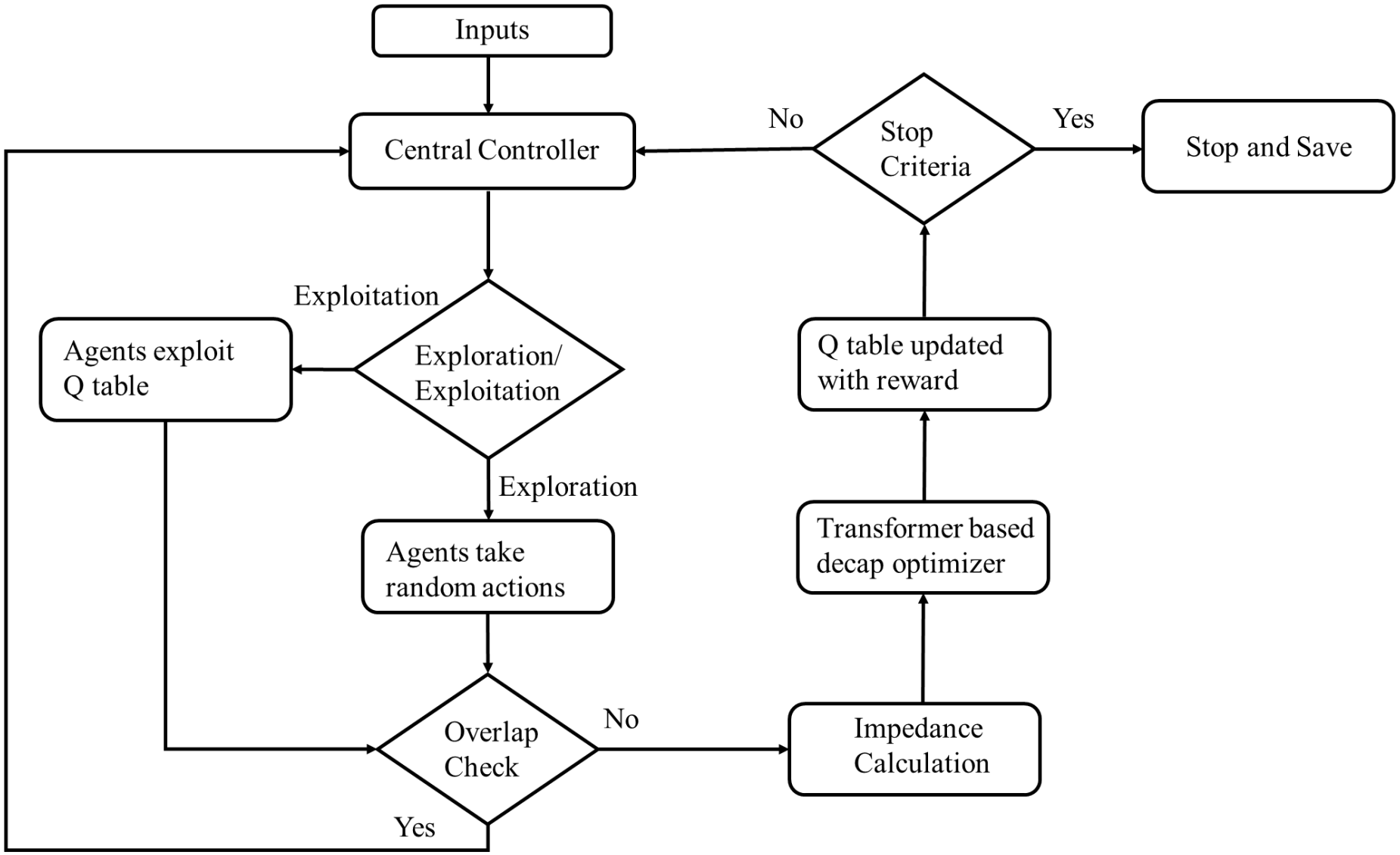




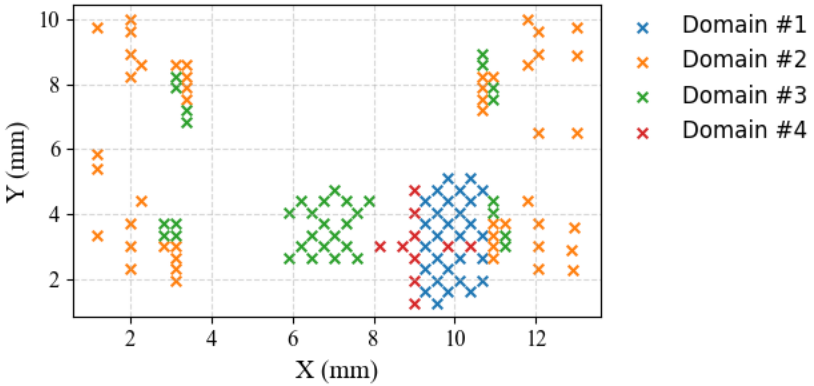
Multi-Agent RL Example [2]

[1] https://huggingface.co/blog/deep-rl-q-part2
[2] https://huggingface.co/learn/deep-rl-course/unit7/introduction-to-marl

Proposed Algorithm Flow

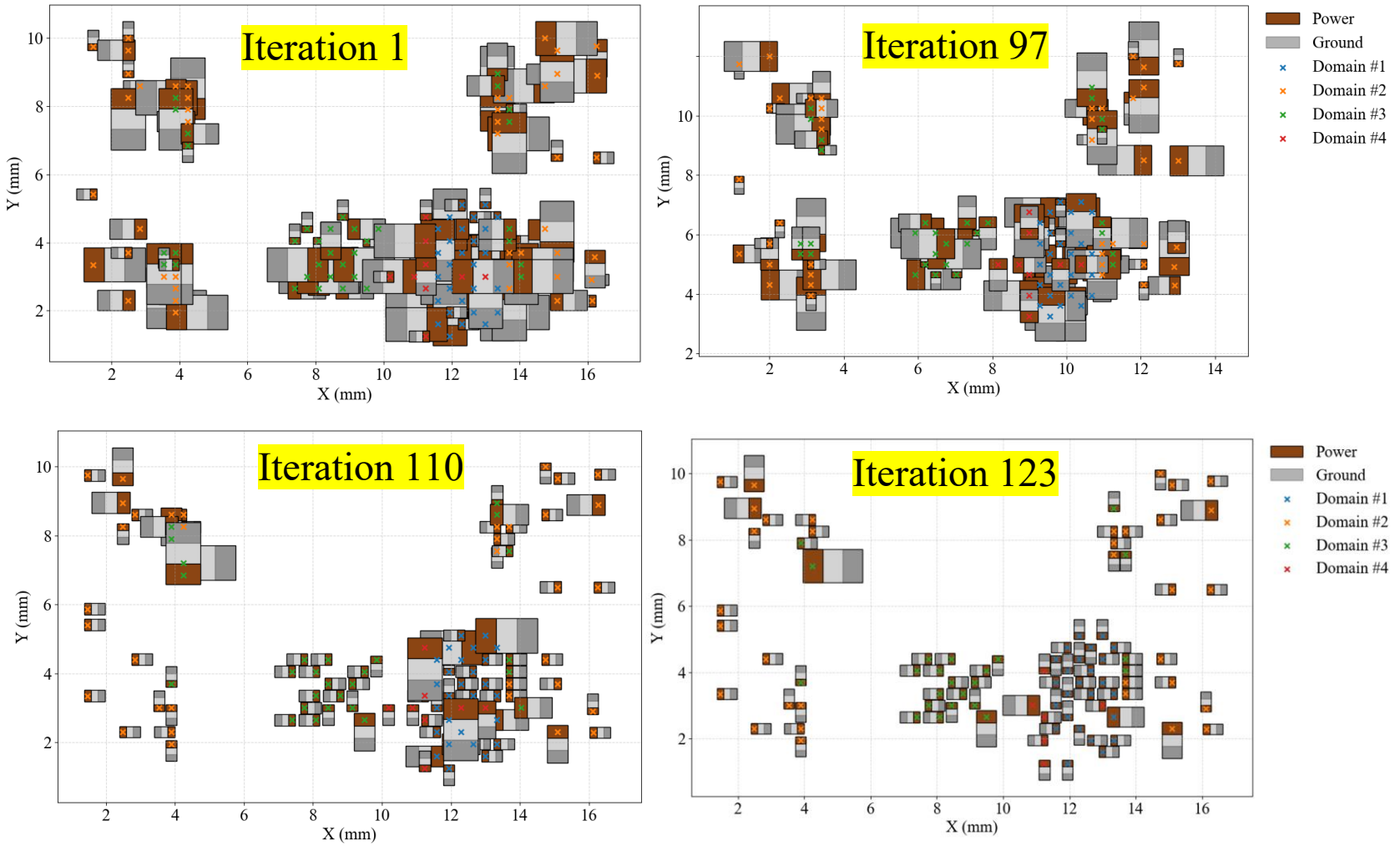


Test Case #1

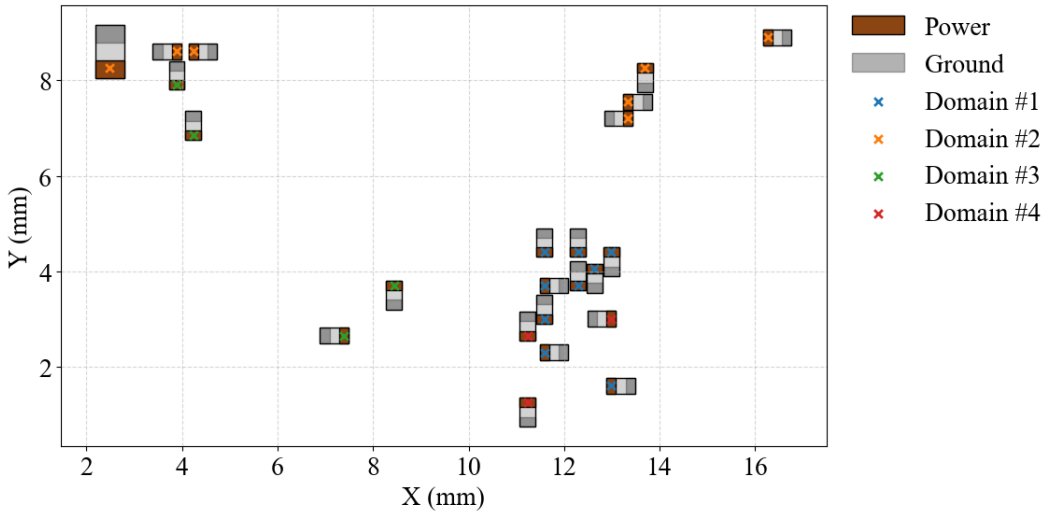


Top View

Domain #	Target Impedance (mohm)
1	8.4
2	17.5
3	43
4	80

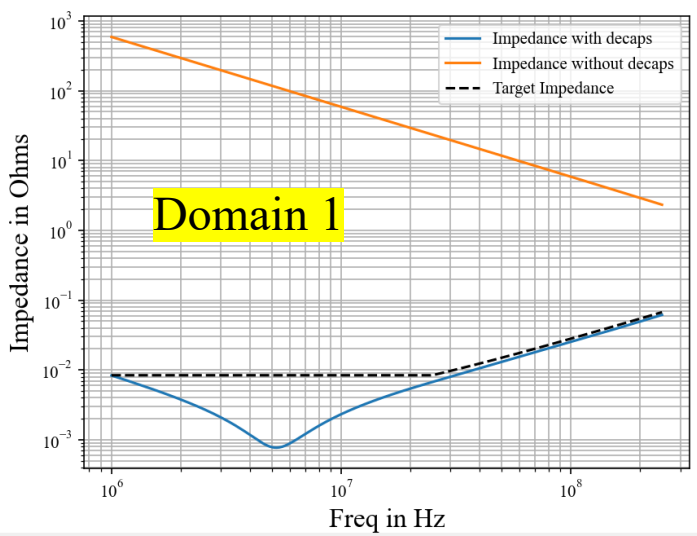


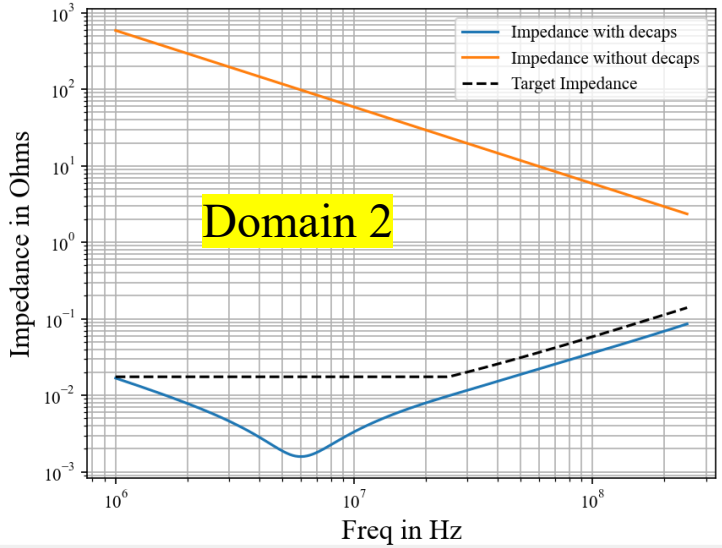
Test Case #1

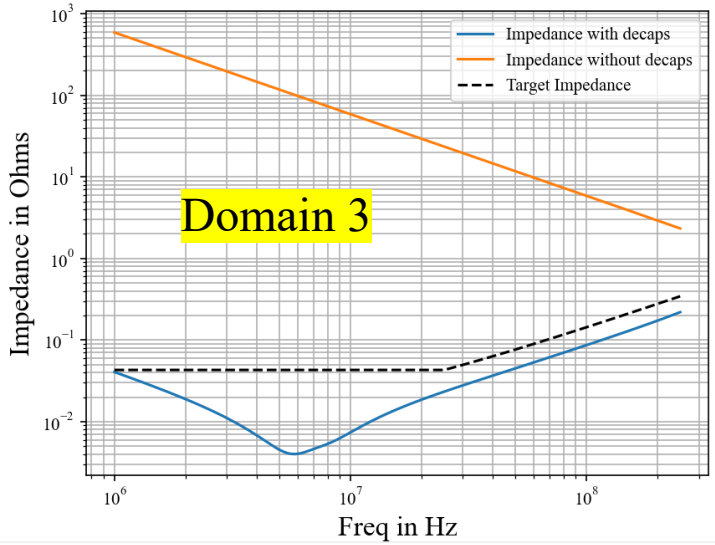


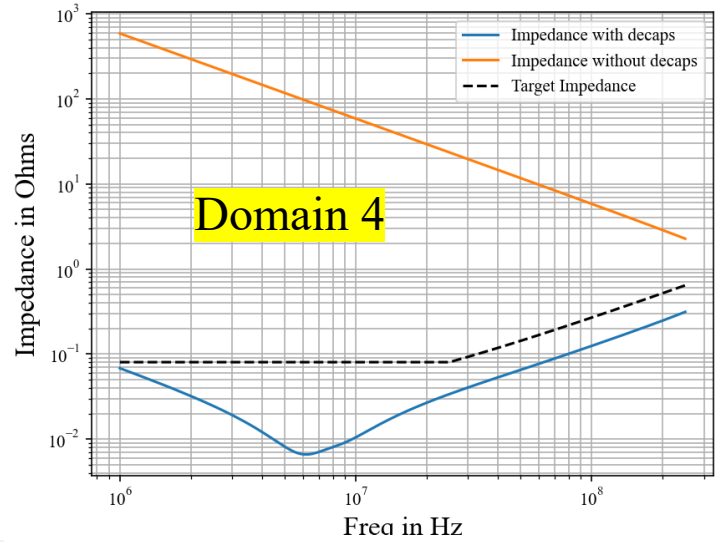
Optimized Decap placement

Domain #	# of decaps
1	9
2	7
3	4
4	3





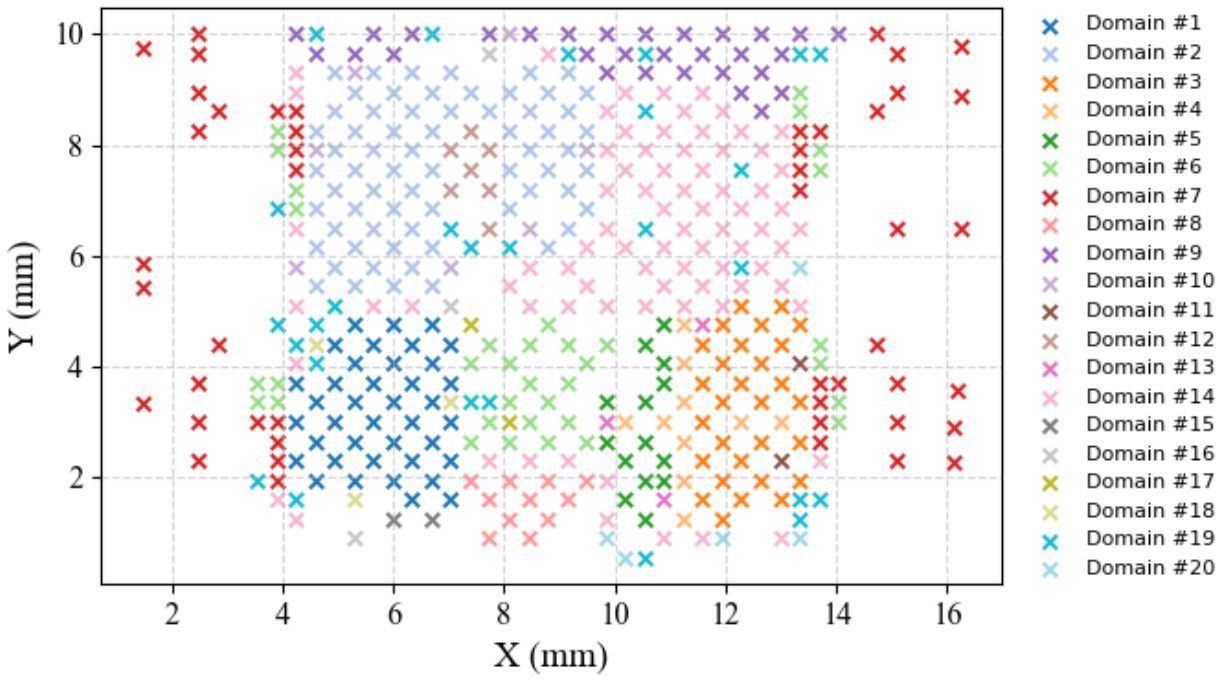




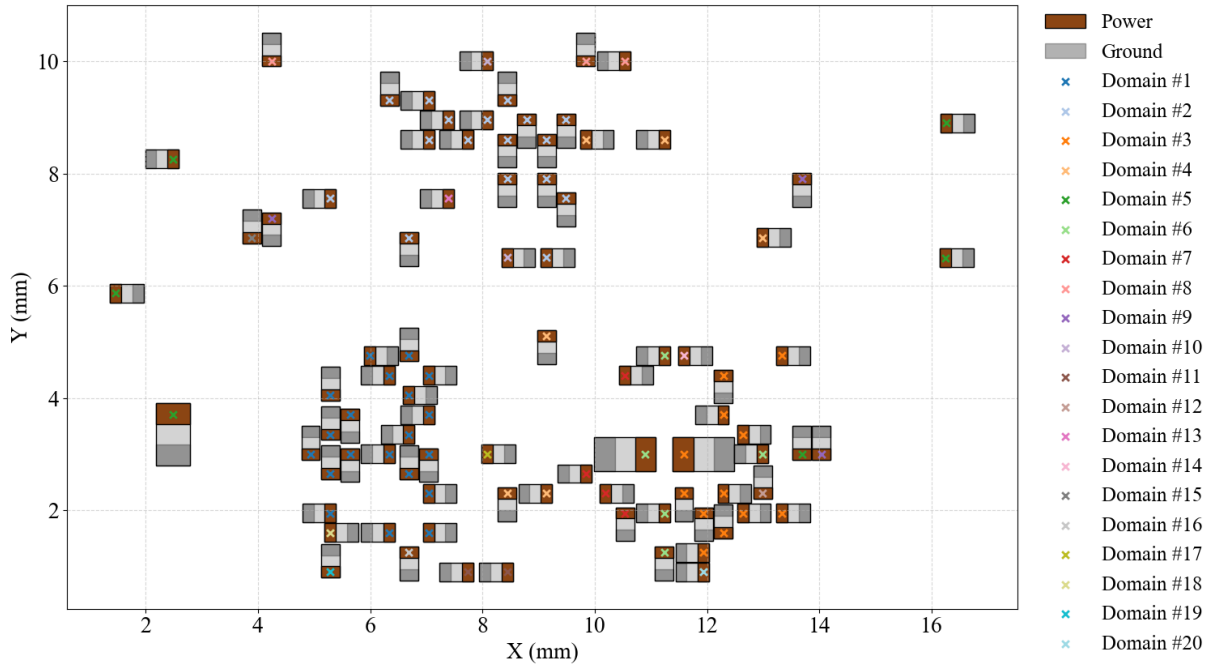
Complex Test Case

- Real World SOC with 20 power domains.

Top View



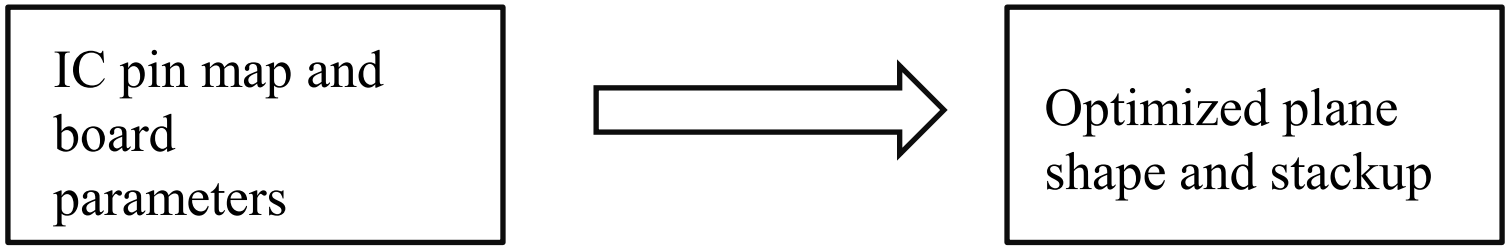
Optimized Decap placement



PRE-LAYOUT POWER PLANE AND STACKUP SYNTHESIS

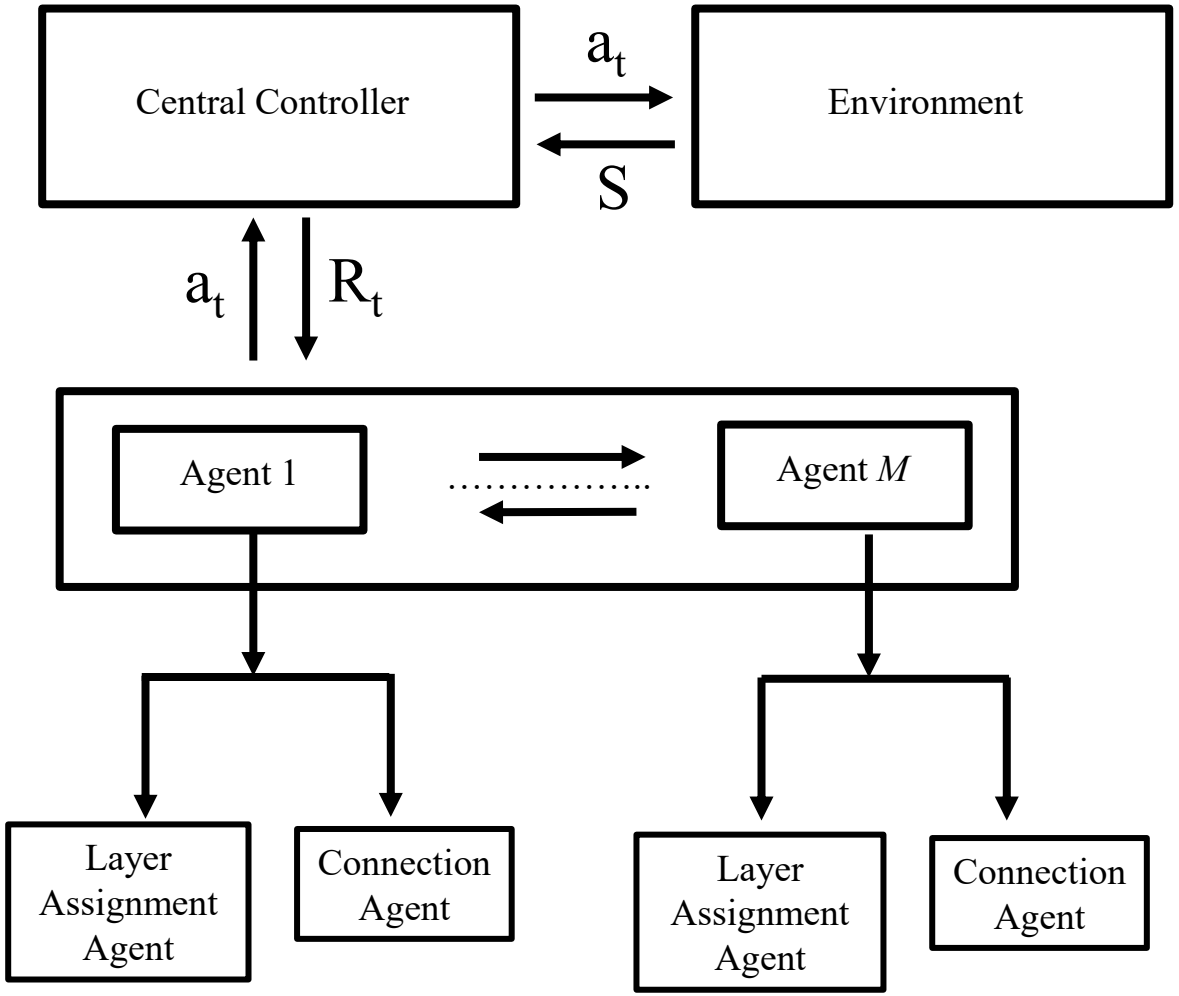
Introduction

- Modern high-performance systems have multiple power domains, each requiring isolated, low-noise power delivery.
- Traditionally, power plane shape and layer stackup are manually crafted by expert designers through trial-and-error and iterative simulations.
- Automatically synthesize power plane shapes and layer stackup for multi-power domain PCB PDN designs using Multi-Agent Reinforcement Learning.



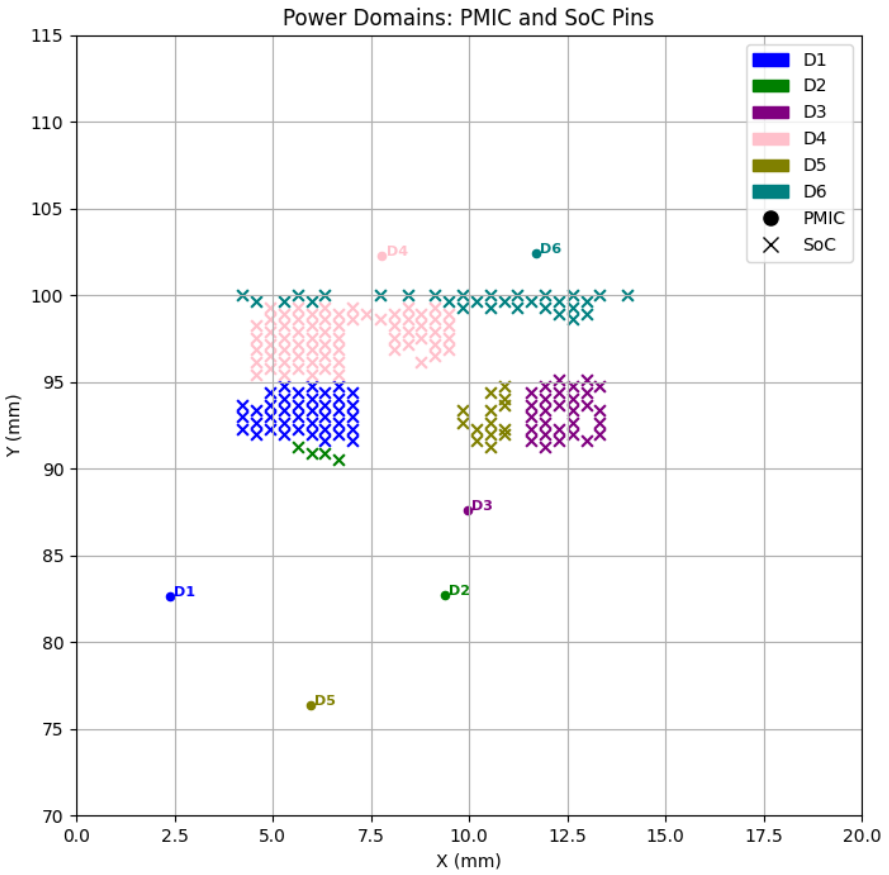
Proposed MARL Framework

- Each power domain is controlled by an individual agent that collaborates through a central controller.
- Specialized sub-agents handle tasks like **layer assignment** using stackup constraints and **connection routing** to ensure full power domain connectivity to PMIC pins.



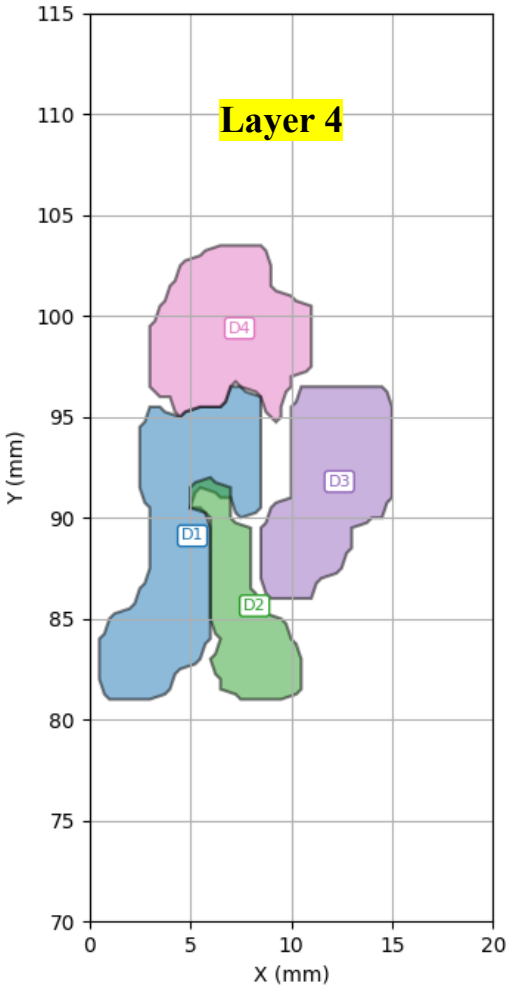
Test Case

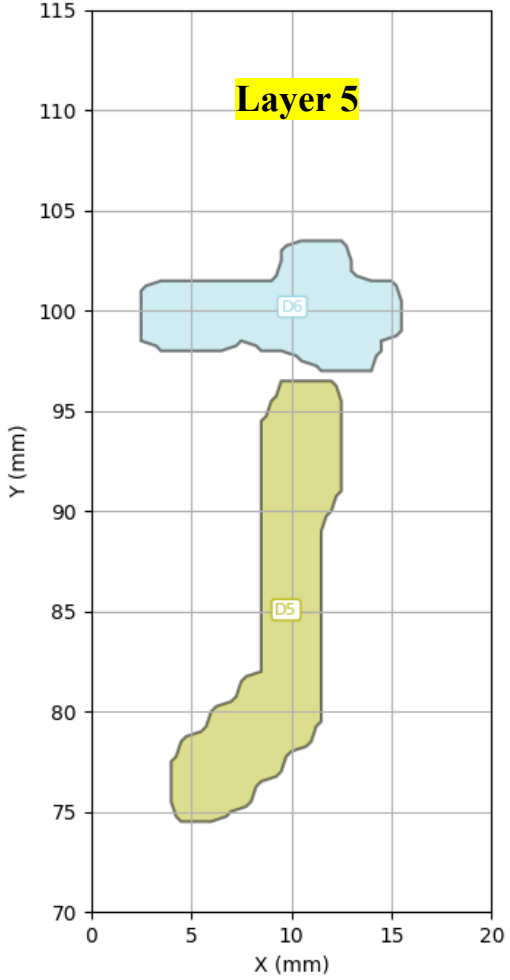
Input



Top View

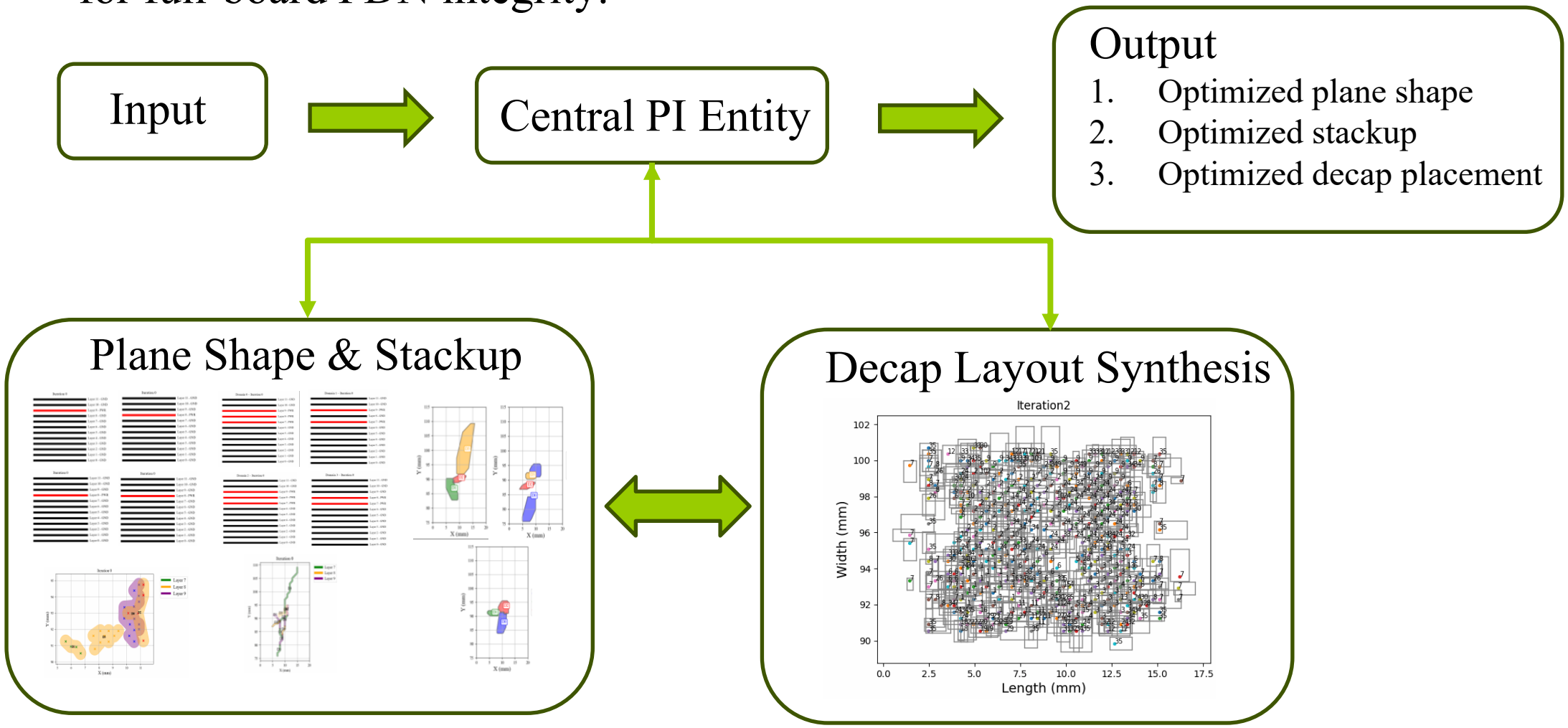
Output





Future Work

- Centralized PI entity that co-optimizes stackup, plane shapes, buried vias, and decaps for full-board PDN integrity.



Thank You