# ML-Based Fine-Grained Modeling of DC Current Crowding in Power Delivery TSVs for Face-to-Face 3D ICs

Zheng Yang[1], Zhen Zhuang[2], Bei Yu[2], Tsung-Yi Ho[2], Martin D.F. Wong[3], Sung Kyu Lim[1]

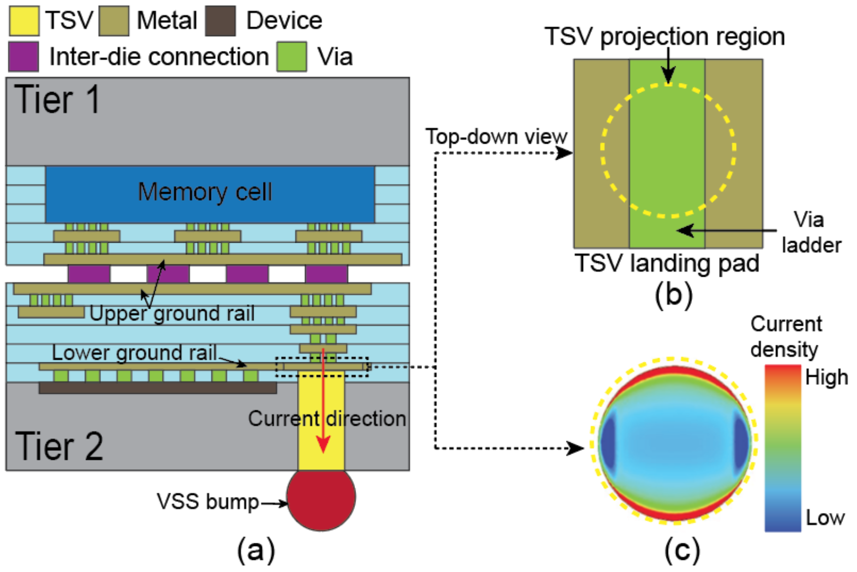
[1]Georgia Institute of Technology
[2]The Chinese University of Hong Kong
[3]Hong Kong Baptist University

# Outline

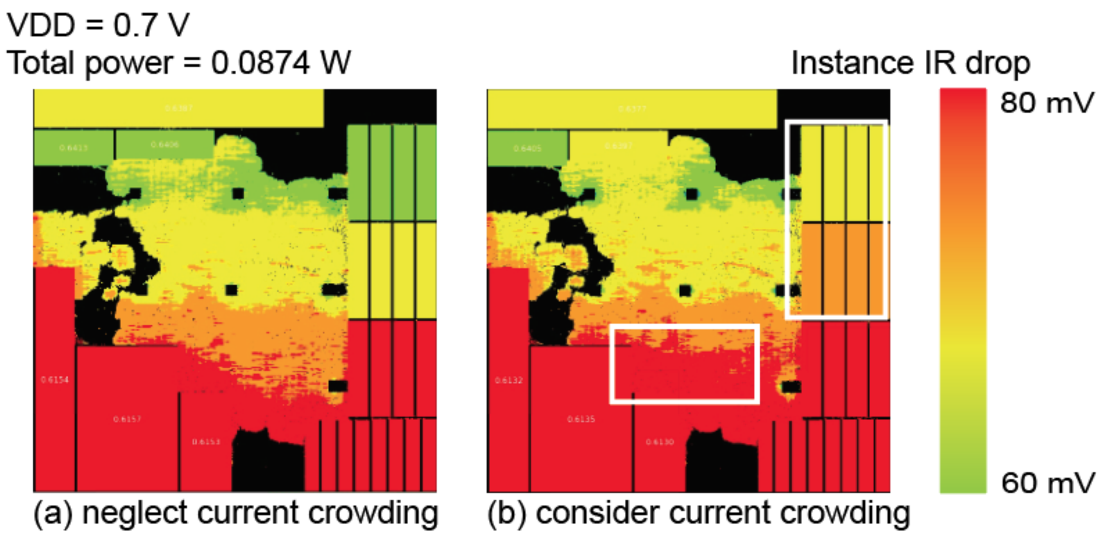
- Background
  - Introduction to current crowding
  - Current crowding in face-to-face 3D ICs
- Previous work
  - TSV resistor network model
- Motivation
  - Graph attention network (GAT)
- Methodology
  - Fine-grained graph construction
  - GAT-based machine learning model
- Experimental results
- Conclusion

# Introduction to Current Crowding

- Current non-uniformly distributes at the interface between TSV landing pad and via ladder
- Current crowding is caused by the different cross-sectional areas between TSV pillar, TSV landing pad and via ladder
- Current crowding leads to higher IR drop and reliability issue




▲ Fig 1: (a) face-to-face 3D IC structure (b) different cross-sectional areas (c) current density map

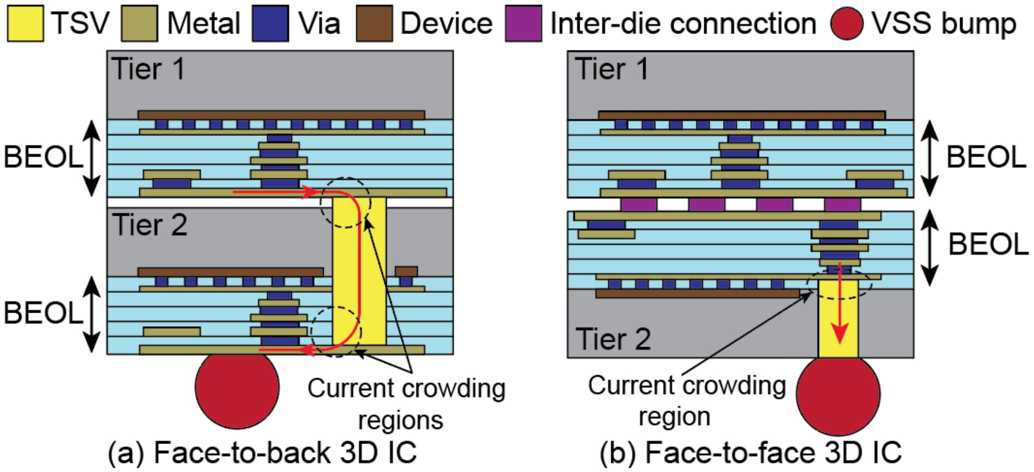



▲ Fig 2: Tier 2 IR drop maps

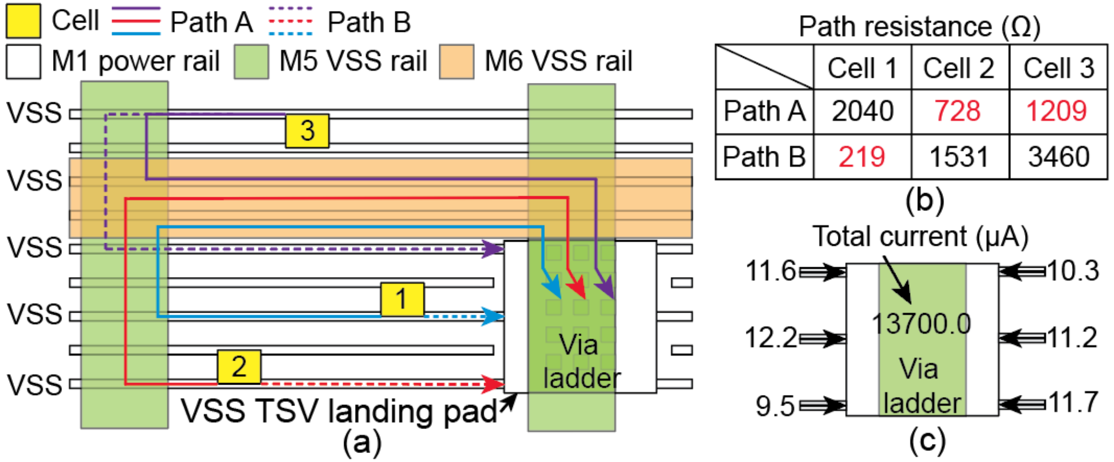
# Background

## Current Crowding in Face-to-Face 3D ICs

- In face-to-face 3D ICs, current crowding happens at the interface between TSV pillar, TSV landing pad and via ladder
- There are two kinds of power rail to TSV connections: power rail to the side of TSV and via ladder to the top of TSV landing pad
- Most of current flows to power TSV from the via ladder to the top of TSV landing pad




▲ Current crowding in face-to-back and face-to-face 3D ICs




▲ Fig 2: Power TSV connections in face-to-face 3D IC PDN

# Previous Work

## TSV Resistor Network Model[1]

**Motivation**

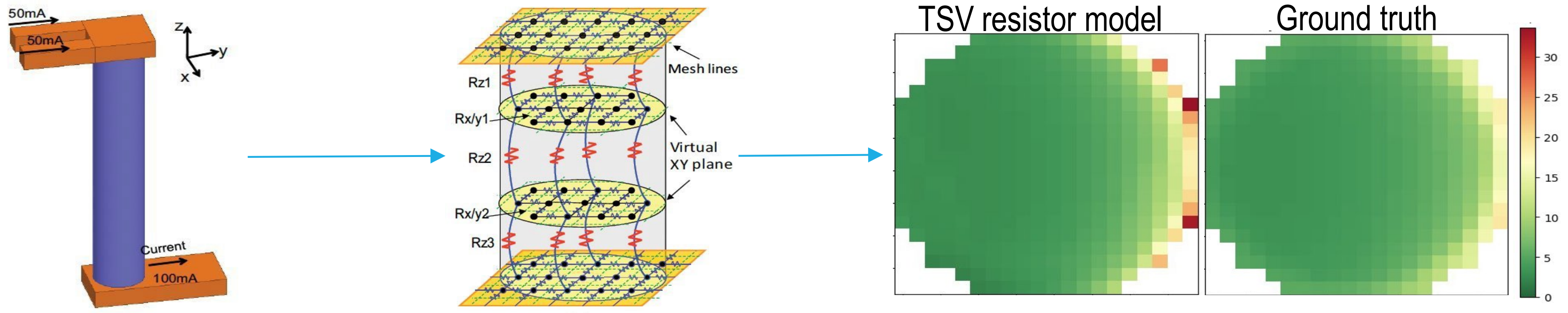
- Finite element analysis (FEA) is time-consuming, often taking over an hour to analyze.

**Methodology**

- This method segments TSV structure into a 3D mesh cube network where each mesh cube contains resistors
- Then construct a fine-grained resistor network representing the TSV structure

**Problem**

- In regions with severe current crowding, the method lacks accuracy
- For large resistor networks, this method still needs ~2 minutes




(a) (b) (c)

▲ (a) TSV structure (b) TSV resistor model (c) comparison of the TSV resistor model's current density map with ground truth

[1]Xin Zhao, Michael R. Scheuermann, and Sung Kyu Lim. Analysis and modeling of dc current crowding for tsv-based 3-d connections and power integrity. IEEE Transactions on Components, Packaging and Manufacturing Technology, 4(1):123–133, 2014.

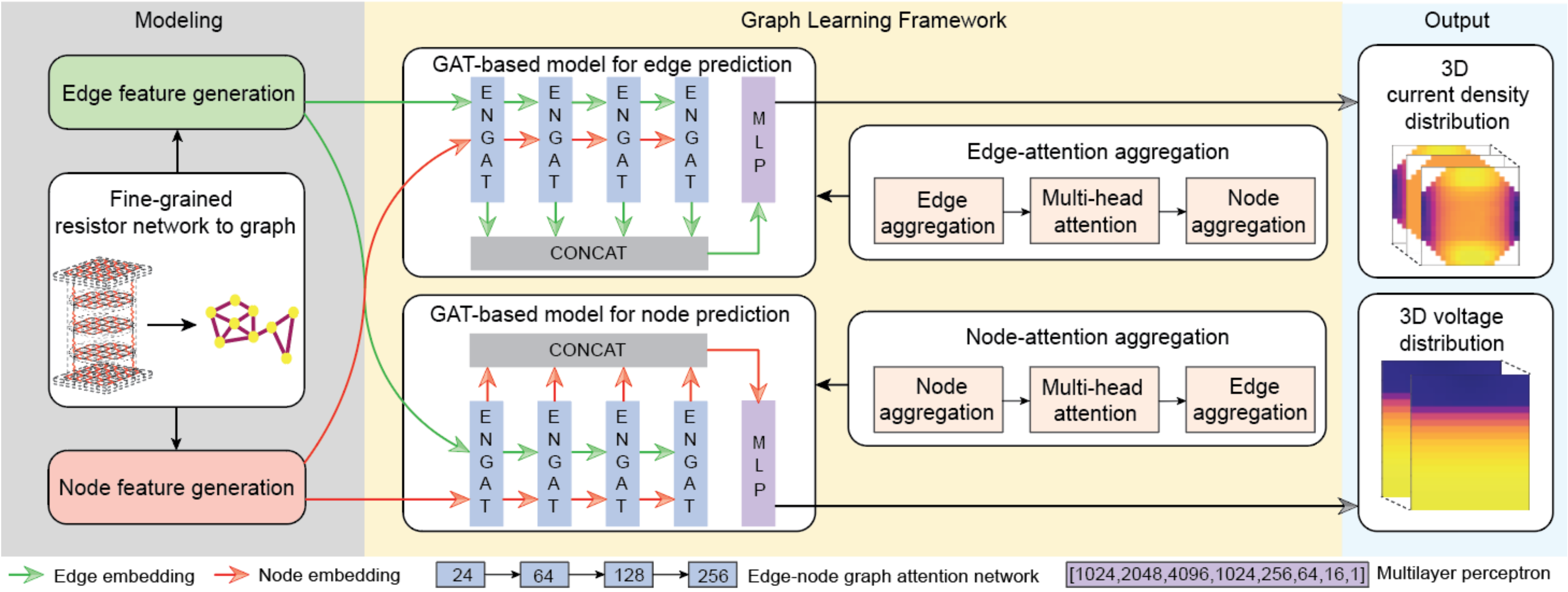
# Advantages of Graph Attention Network

- Self-attention mechanism allows GAT to extract useful information from graph structures
- Multi-head attention enhances GAT to process multiple types of features
- GAT effectively mitigates the accuracy degradation associated with the discretization of cubes compared to resistor model method
- Our GAT framework accurately predicts the current density distribution with an inference time of only 2–3 seconds

# Methodology

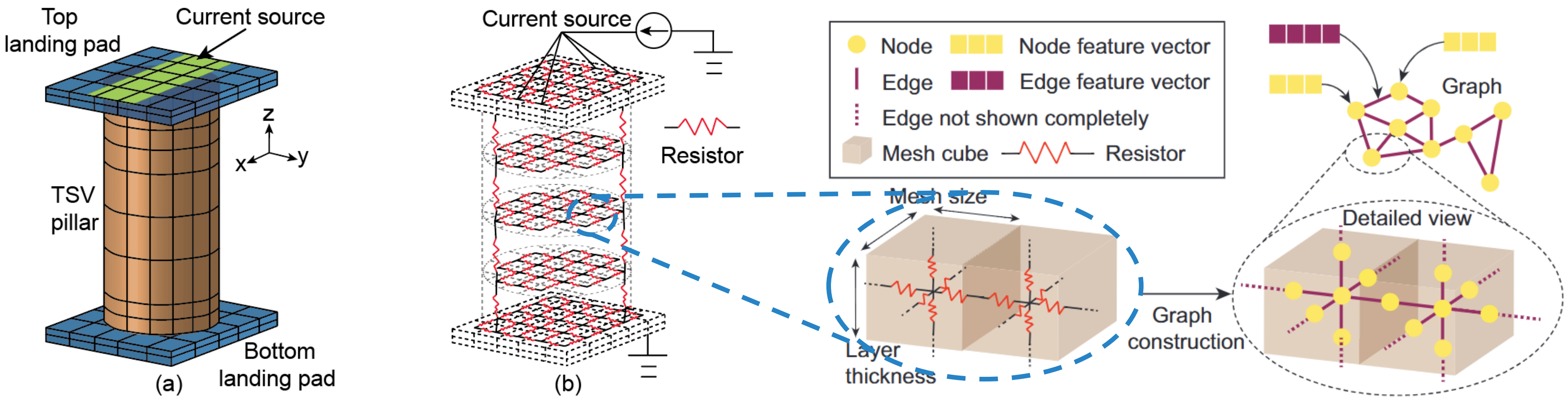
## Overview

- Fine-grained graph construction
- GAT-based framework



# Fine-Grained Graph Construction

- We assign feature vectors to each edge and node in the graph
- Edge
  - Each resistor in the resistor model is converted into an edge
  - Edge label is current density
- Node
  - Each terminal in the resistor model is converted into a node
  - Edge label is voltage




▲ From resistor network to fine-grained graph

# Methodology

## Fine-Grained Graph Construction

- We embed various information into node and edge feature vectors
- We introduce two structural similarity metrics to enrich the structural information – global structural similarity score (GSSS) and local structural similarity (LSSS)[2] – to edge feature vectors

| | Features | Type | Unit |
|---|---|---|---|
| input | coordinates | node | $\mu m$ |
| | voltage reference | node | bool |
| | coordinates | edge | $\mu m$ |
| | resistance | edge | $\Omega$ |
| | current | edge | A |
| | cross-sectional area | edge | $\mu m^2$ |
| output | voltage | node | V |
| | current density | edge | $A/m^2$ |

▲ Table 1: Fundamental input and output for GAT models

$$GSSS = sigmod(\frac{N_G(i) \cap N_G(j)}{N_G(i) \cup N_G(j)})$$

$N_G(i)$ and $N_G(j)$ represent the numbers of the neighbor nodes of node $i$ and node $j$

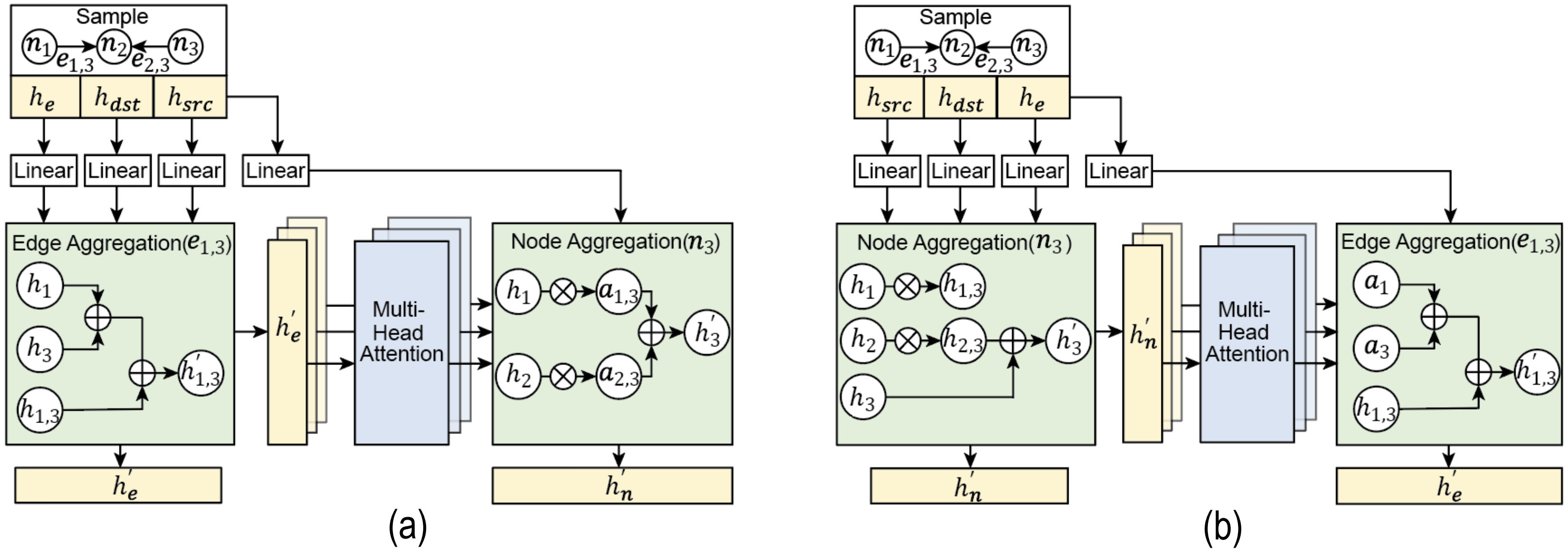
$$LSSS = sigmoid(\exp(-\frac{\|\boldsymbol{h}_i - \boldsymbol{h}_j\|^2}{2}))$$

$h_i$ and $h_j$ represent the feature vectors of node $i$ and node $j$, respectively

[2]Ankith Jain Rakesh Kumar and Bir Bhanu. Relational edge-node graph attention network for classification of micro-expressions. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 5819–5828,2023.

# GAT-Based Framework

- Due to the different nature of node prediction task and edge prediction task, we design two different GAT layers with different aggregation mechanisms – edge-node graph attention layer (ENGAT) with node-attention aggregation and edge-node graph attention layer (ENGAT) with edge-attention aggregation

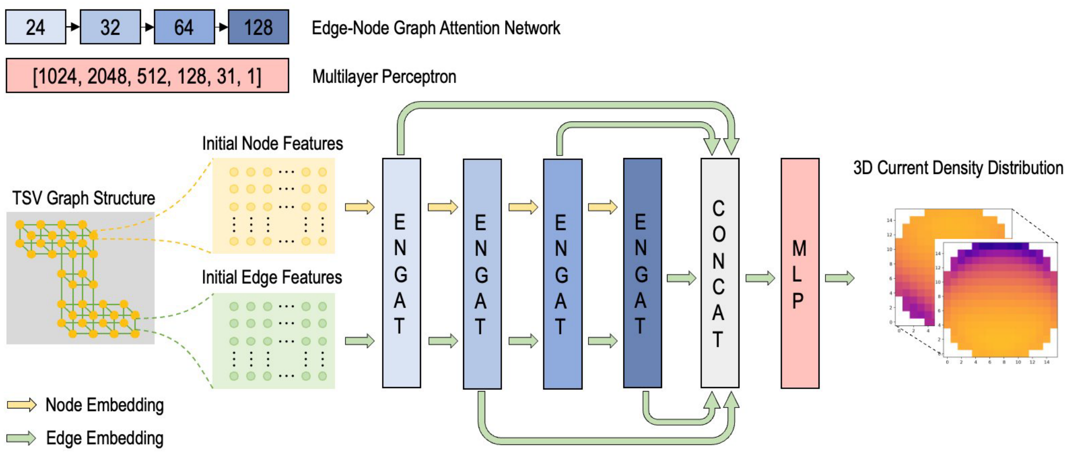



▲ (a) edge-attention aggregation for edge prediction (b) node-attention aggregation for node prediction

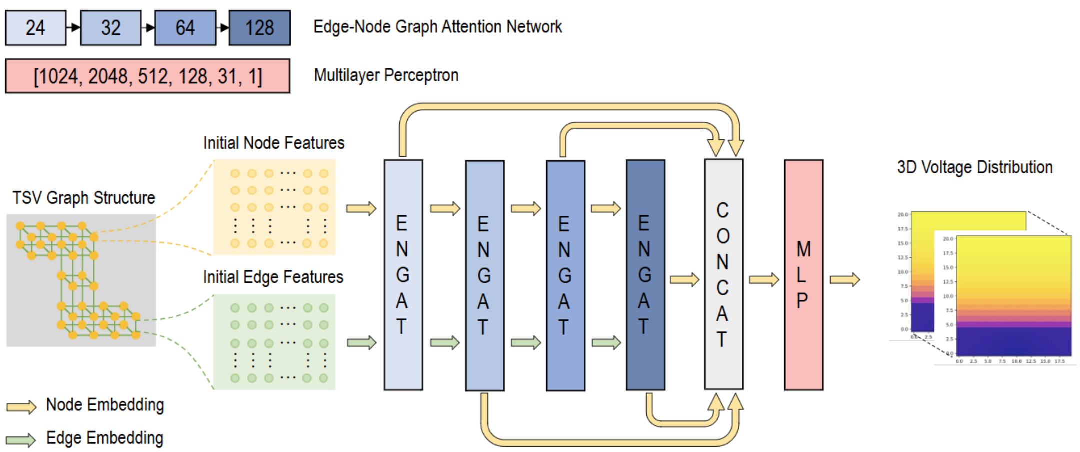
# Methodology

## GAT-Based Framework

- We use two different GAT models for edge prediction and node prediction
- Each GAT model has <u>four layers of edge-node graph attention layer</u> for encoding and <u>eight-layer multi-layer perceptron (MLP)</u> for decoding




▲ GAT model for current density prediction




▲ GAT model for voltage prediction

# Experimental Results

## Experimental Setup

- Considering four variables, we build a dataset consisting of 108 different TSV structures
- For each TSV structure, we build a fine-grained graph which contains thousands of edges and nodes
- We use ANSYS Q3D for simulation to get the current density distribution and voltage distribution of these 108 TSV structures as ground truth which takes 74 hours in total
- We use supervised learning to train two GAT models. The model of edge prediction is trained in 33.2 hours for 3000 epochs while the model of node prediction is trained in 33.8 hours for 8000 epochs

| Configurations | Values |
|---|---|
| TSV diameter ($\mu$m) | 2, 3, 4, 5 |
| TSV aspect ratio | 9, 10, 11 |
| landing pad thickness ($\mu$m) | 0.05, 0.1, 0.2 |
| via ladder width ($\mu$m) | 0.8, 1.0, 1.2 |
| # total data point | 108 |

▲ Table 1: Variables in our TSV dataset

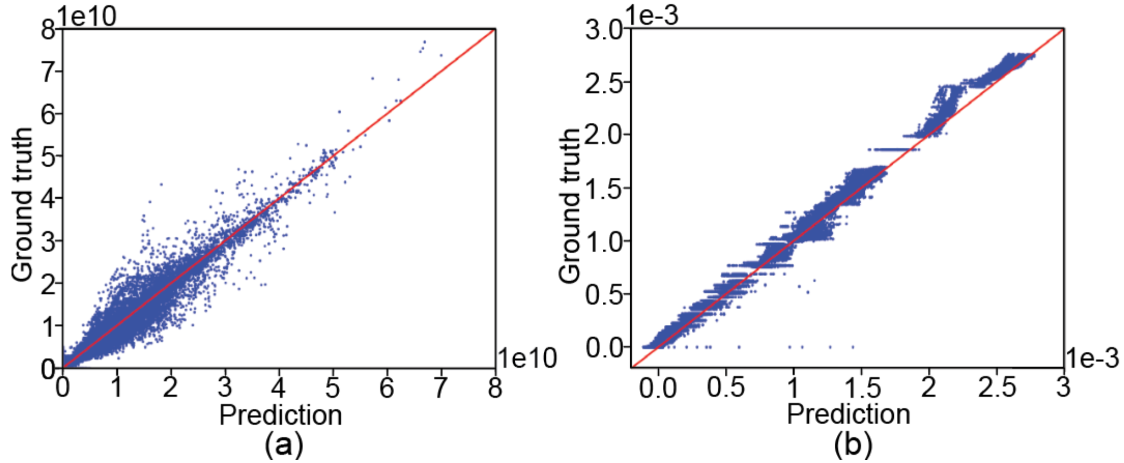
| GAT graph statistics | | | GAT dataset statistics | |
|---|---|---|---|---|
| # Edge (per graph) | Min | 31073 | # Graph (datapoint) | 108 |
| | Max | 49061 | # Graph for training | 87 |
| | Mean | 37729 | # Graph for testing | 21 |
| # Node (per graph) | Min | 20986 | # Edge in total | 4074732 |
| | Max | 33082 | # Node in total | 2759291 |
| | Mean | 25549 | Simulation time (hr) | 74 |

▲ Table 2: Our fine-grained graph dataset statistics

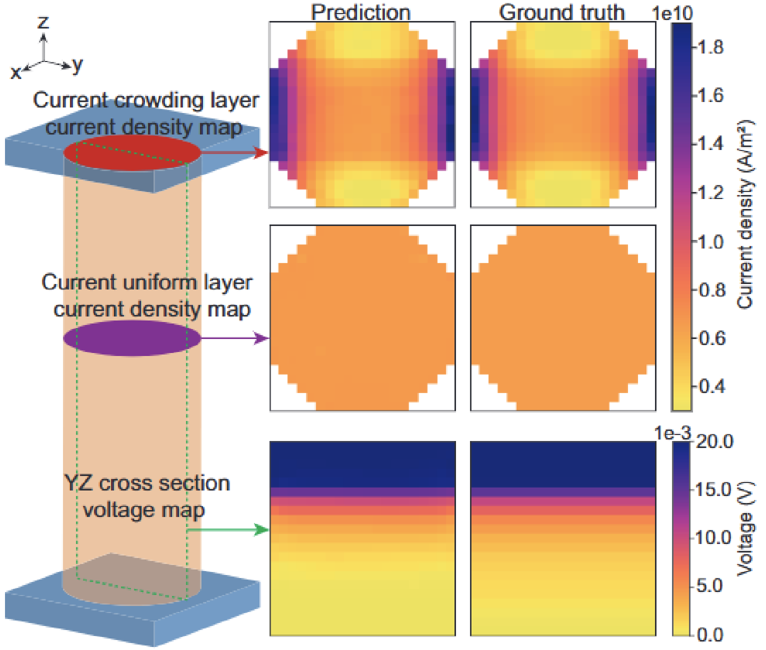
# Experimental Results

## Prediction Accuracy

- In the test set, our GAT models achieve good accuracy. The coefficient of determination ($R^2$) is 0.9776 for edge prediction. $R^2$ is 0.9952 for node prediction
- Our prediction results also show good symmetry




▲ Fig 1: (a) Current density prediction vs ground truth (b) voltage prediction vs ground truth




▲ Fig 2: GAT-based models prediction distribution

# Experimental Results

## Comparison with Different Methods

- Compared with convolution graph neural network (GCN)[3] and MLP, our GAT-based models demonstrate higher accuracy with 2-3 seconds inference time
- Compared with non-ML methods, our GAT-based models are more efficient for large-scale inference

| Method | Current density prediction | | | Voltage prediction | | |
|---|---|---|---|---|---|---|
| | $R^2$ | RMSE ($\times 10^8$ A/m$^2$) | Inference time ($s$) | $R^2$ | RMSE ($\times 10^{-5}$ V) | Inference time ($s$) |
| EMGraph [9] | 0.9182 | 7.7749 | **1.0271** | 0.9892 | 6.8503 | **1.4266** |
| MLP | 0.9076 | 8.2612 | 3.0333 | 0.9909 | 6.3013 | 2.6730 |
| Our GAT framework | **0.9776** | **4.0730** | 3.0626 | **0.9952** | **4.5526** | 2.7949 |

▲ Comparisons with other ML-based methods

| Method | Current density prediction | | | Voltage prediction | | |
|---|---|---|---|---|---|---|
| | $R^2$ | RMSE ($\times 10^8$ A/m$^2$) | Inference time ($s$) | $R^2$ | RMSE ($\times 10^{-5}$ V) | Inference time ($s$) |
| FEA (Ground truth) | N/A | N/A | 2564.6667 | N/A | N/A | 2564.6667 |
| Method in [3] | 0.2573 | 11.6486 | 136.0979 | **0.9983** | **2.3752** | 136.0979 |
| Our GAT framework | **0.9776** | **4.0730** | **3.0626** | 0.9952 | 4.5526 | **2.7949** |

▲ Comparisons with other non-ML methods

[3]Wentian Jin, Liang Chen, Sheriff Sadiqbatcha, Shaoyi Peng, and Sheldon X.-D.Tan. Emgraph: Fast learning-based electromigration analysis for multi-segment interconnect using graph convolution networks. In 2021 58th ACM/IEEE Design Automation Conference (DAC), pages 919–924, 2021.

# Conclusion

- We investigate the current crowding effect in face-to-face 3D ICs
- We propose a method for constructing fine-grained graphs based on TSV structures
- We design a supervised GAT-based framework for accurately predicting current density and voltage distribution
- We demonstrate that our GAT-based framework outperforms other ML-based methods and non-ML methods

# Thank You for Listening! Q&A