# Methodology of Resolving Design Rule Checking Violations Coupled with Fully Compatible Prediction Model

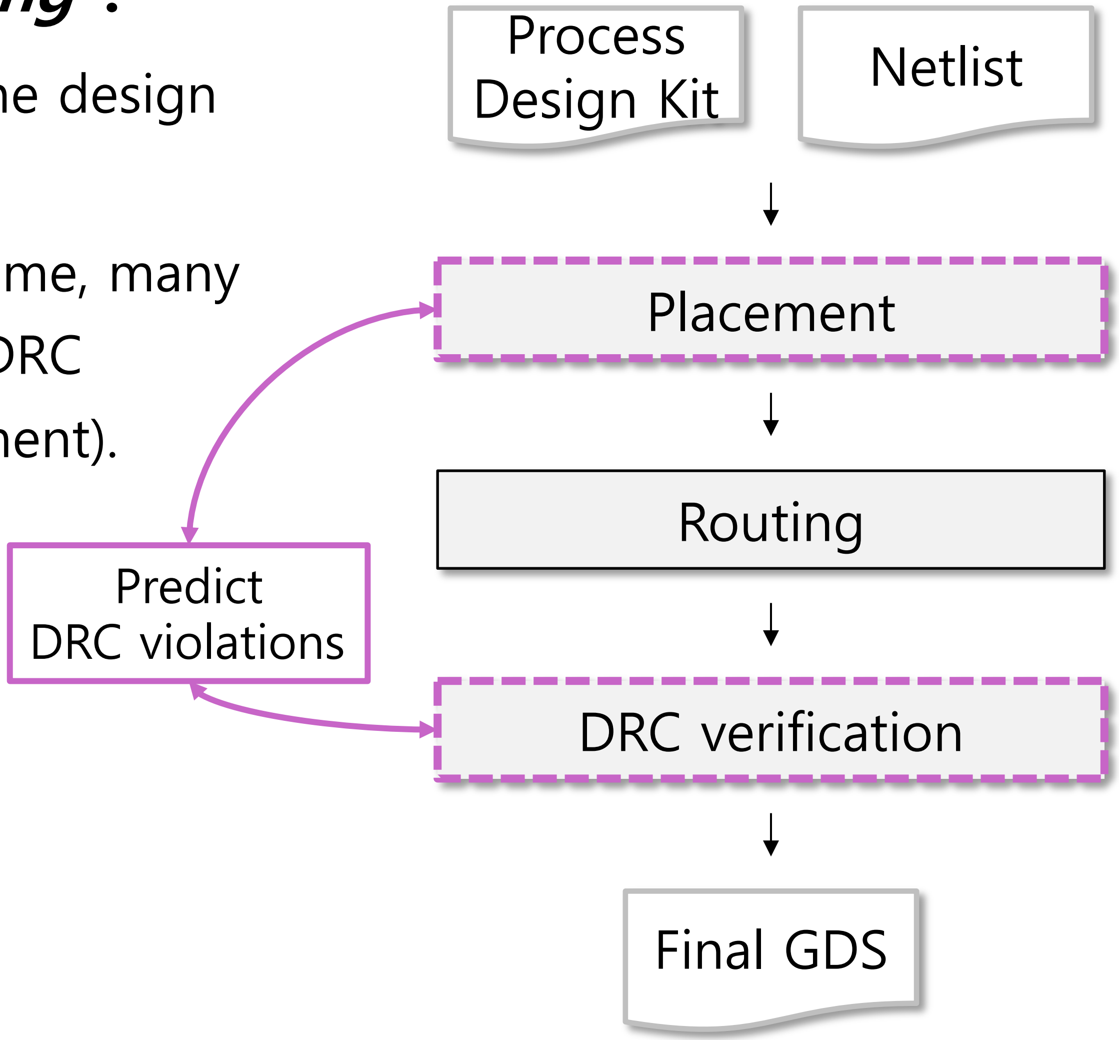
**Suwan Kim, Hyunbum Park, Kyeonghyeon Baek, Kyu-Myung Choi, and Taewhan Kim**

International Symposium on Physical Design

SNUCAD

# Introduction: Design Rule Check (DRC) Violations

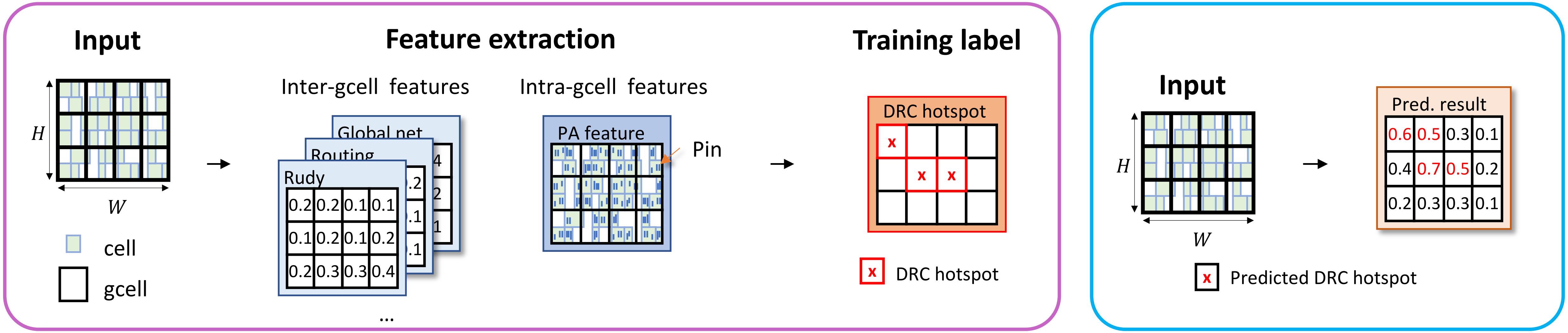
- **DRC violations are verified after "*Routing*".**
  - If the number of DRC violations is too large, the design cannot be taped out!
  - Since "routing" takes a significant amount of time, many researchers have addressed the prediction of DRC violations in the early stage (i.e., during placement).



# Introduction: DRC Violations Prediction (Machine Learning)

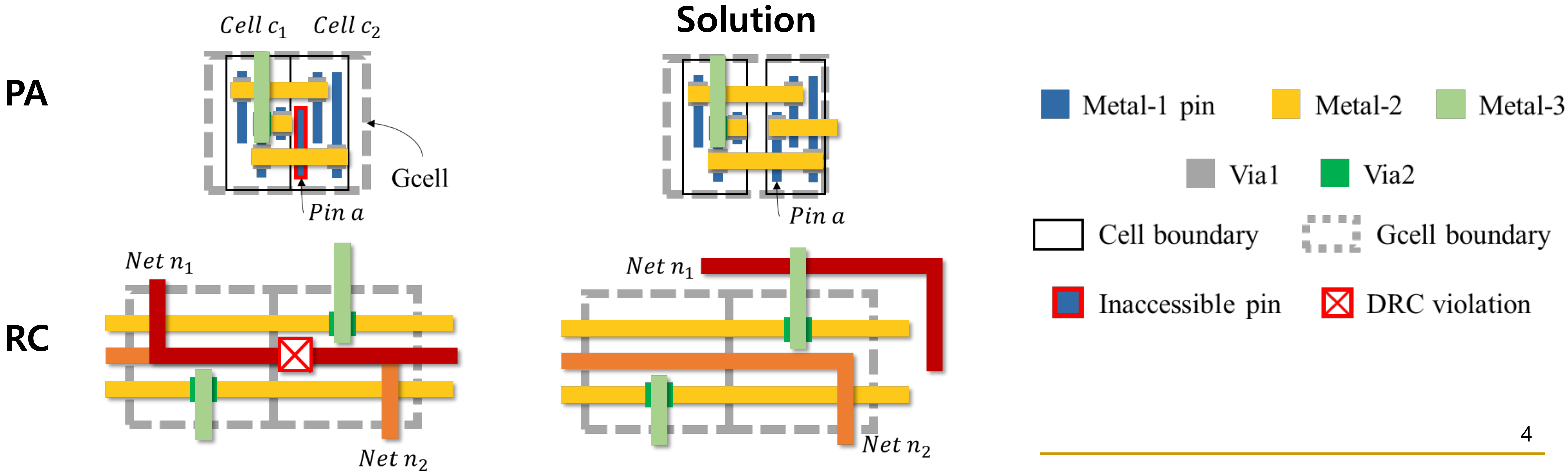
- **Process for predicting DRC hotspot**
  - (DRC hotspot refers to the location where DRC violation occurred.)
  - 1. Divide the placement into a global cell (gcell) grid.
  - 2. Extract features from inter/intra-gcells.
  - 3. Perform supervised learning, assigning a label of 1 to DRC hotspots and a label of 0 elsewhere.

**Training** **Inference**



# Motivation: Two Main Causes of DRC Violation

- **1. Pin accessibility (PA)**
  - (Cause) Blocked pins, (Resolution) Insert whitespace between cells.
- **2. Routing congestion (RC)**
  - (Cause) Excessive routing demand, (Resolution) Detour congested regions.
- **Solutions are not fully interchangeable!**



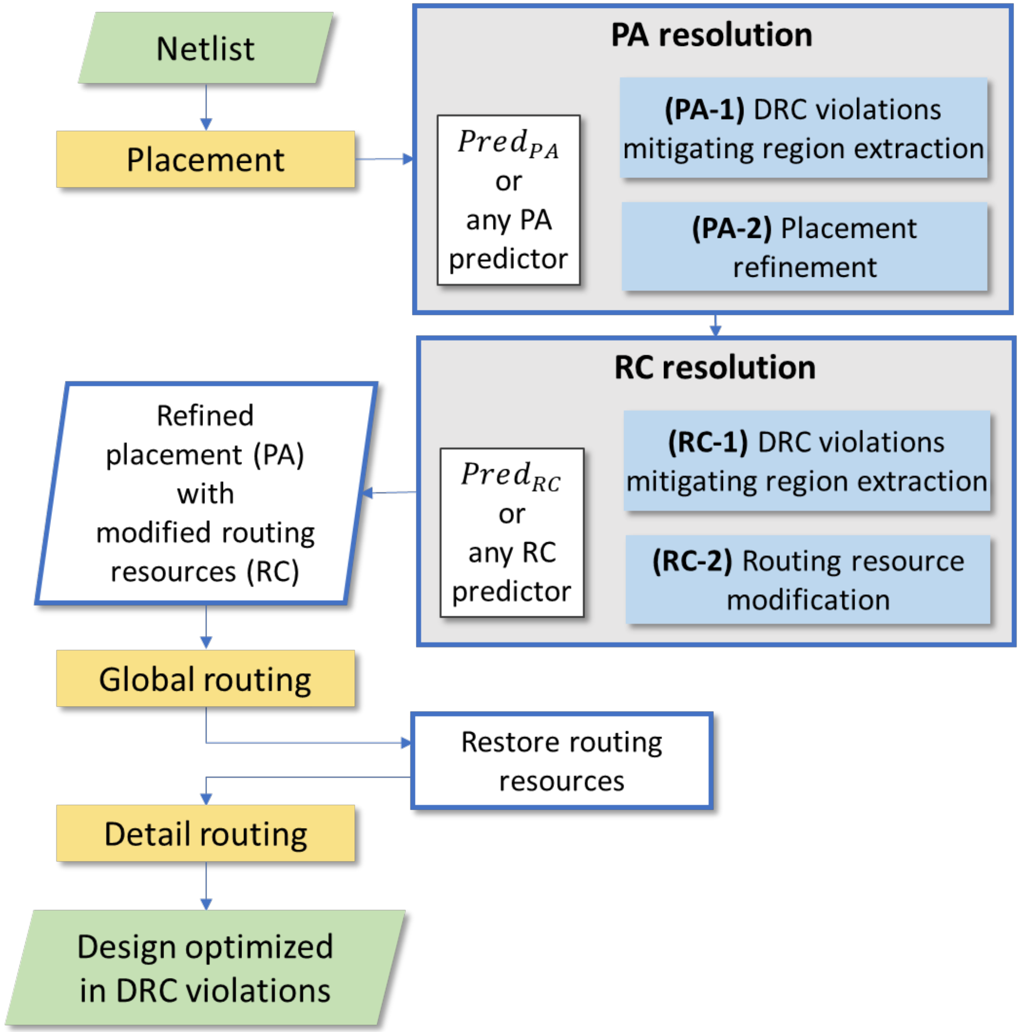
# Motivation: Previous Works

- **Related works**
  - Predict and resolve all types of DRC violations.
    - [1]: Redistribution of whitespace is irrelevant to the prediction value.
  - Predict and resolve only a single type of DRC violations
    - [2]: PA, [3]: RC
- **Proposed framework**
  - Separate model and resolution scheme for PA and RC DRC violations.

[1] Chan, Wei-Ting J., et al. (2017) "Routability optimization for industrial designs at sub-14nm process nodes using machine learning." ISPD.
[2] Yu, Tao-Chun, et al. (2019) "Pin accessibility prediction and optimization with deep learning-based pin pattern recognition." DAC.
[3] Chen, Jingsong, et al. (2020) "PROS: A plug-in for routability optimization applied in the state-of-the-art commercial EDA tool using deep learning." ICCAD.

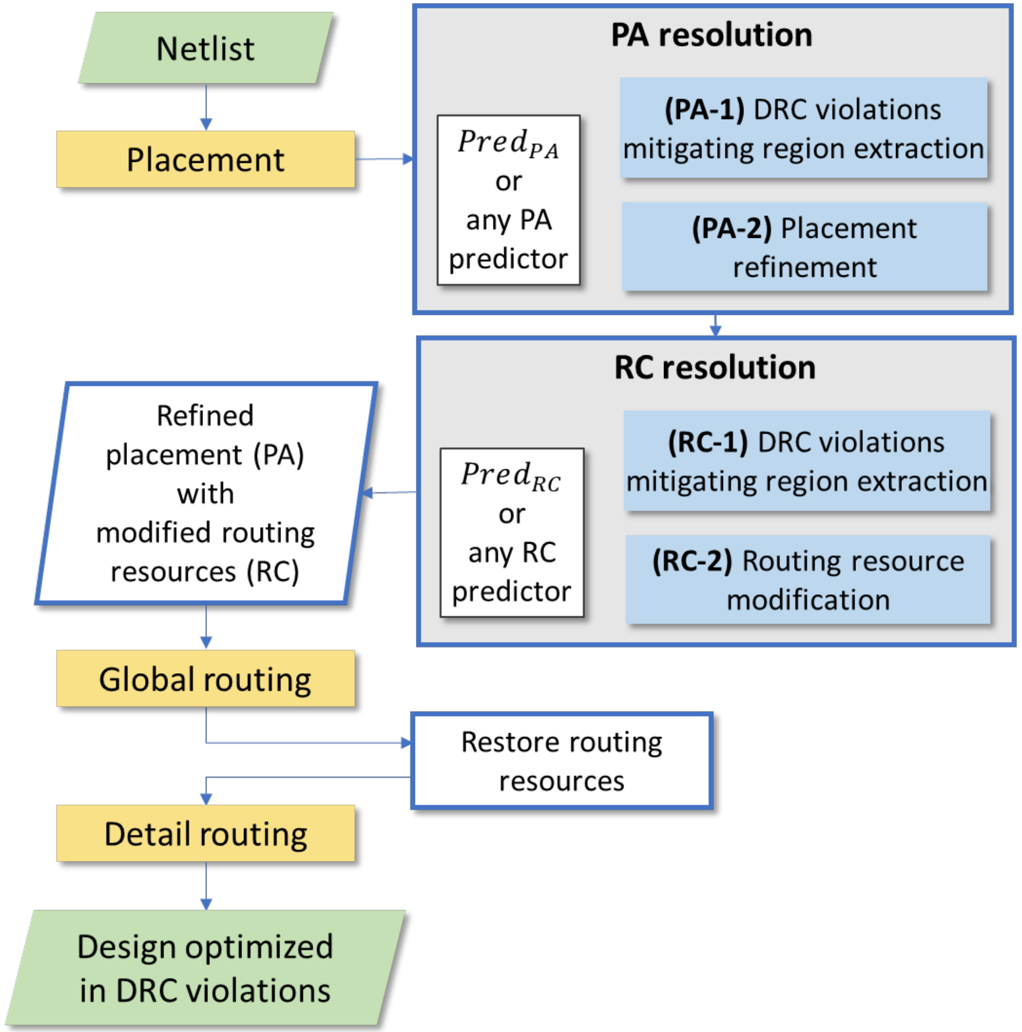
# Methodology

- **Proposed flow**
  - Take placement as inputs,
  - Propose a methodology to separate PA/RC prediction from a given machine learning-based prediction model.
  - Propose a separate PA/RC resolution scheme.



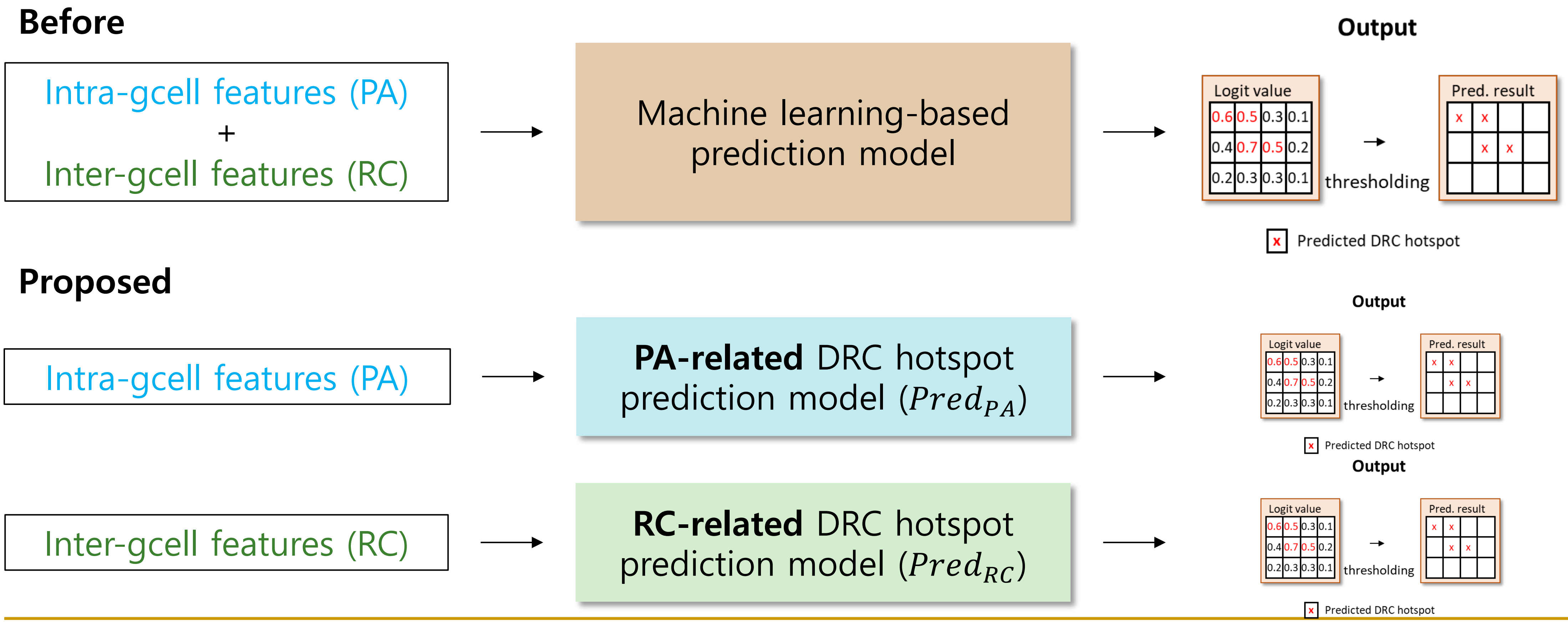
# Methodology

- **Proposed flow**
  - Take placement as inputs,
  - Propose a methodology to **separate PA/RC prediction** from a given machine learning-based prediction model.
  - Propose a separate PA/RC resolution scheme.



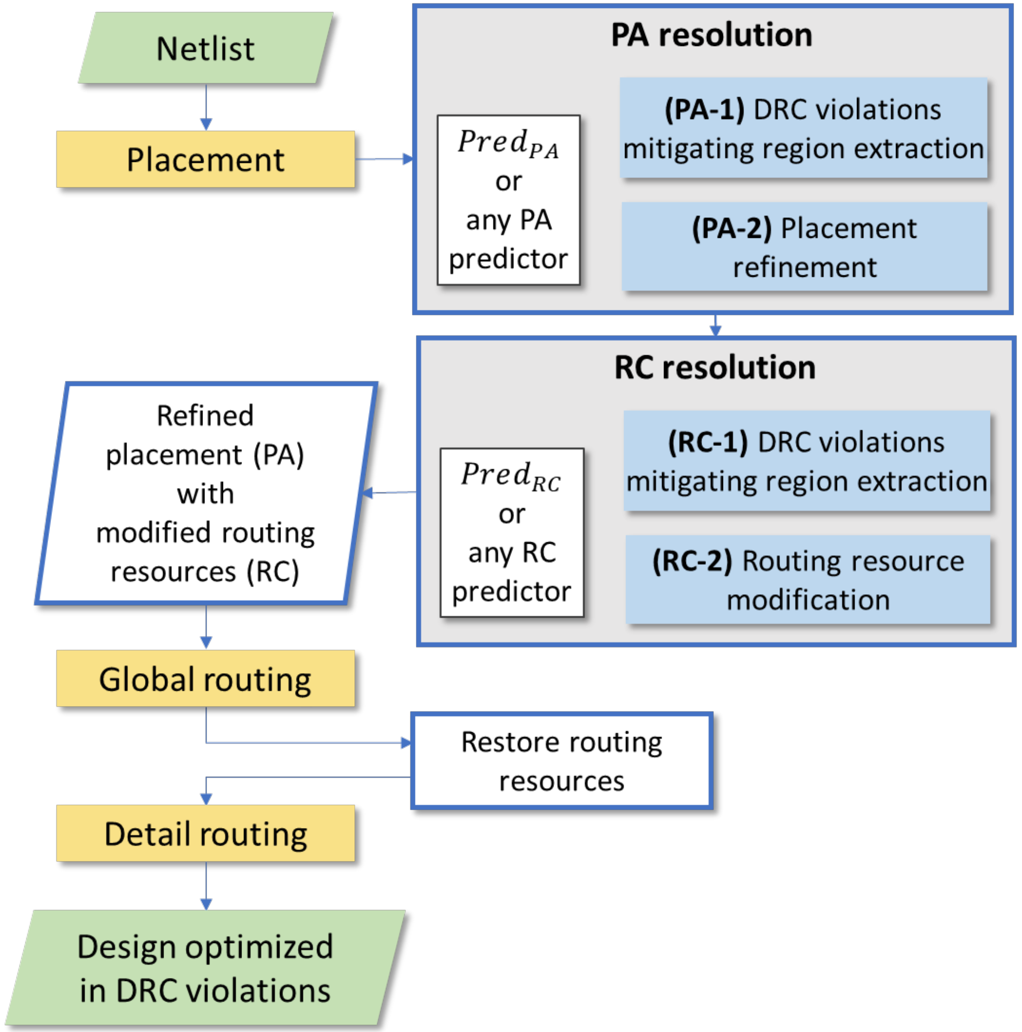
# Develop PA/RC-specified DRC Violation Prediction Model

- **Process**
  - 1. Separate input features (PA: intra-gcell features, RC: inter-gcell features)
  - 2. Retraining models while treating dataset having *noisy label*.



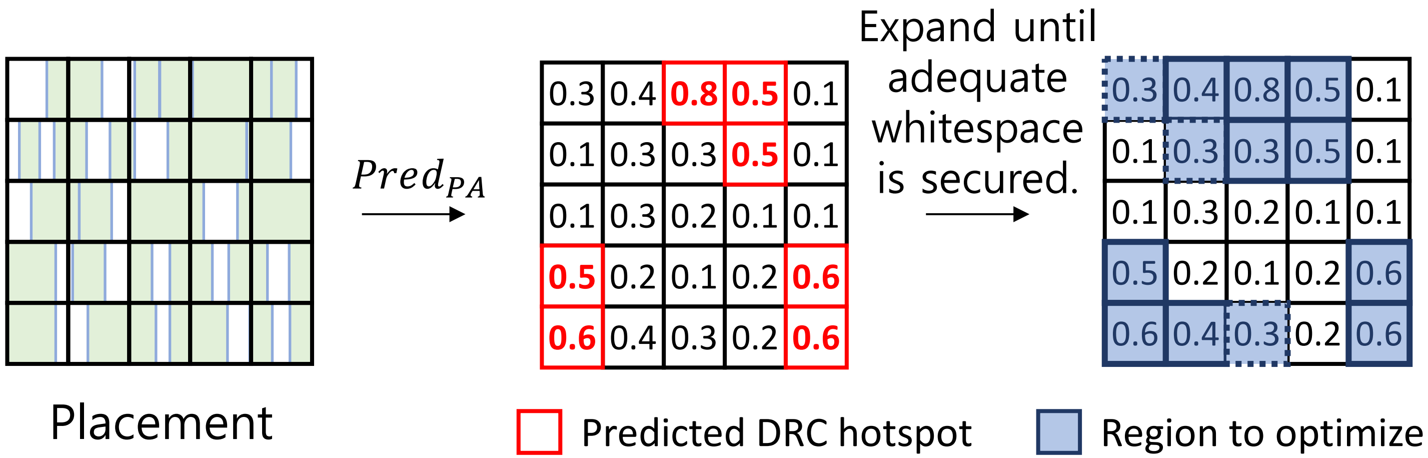
# Methodology

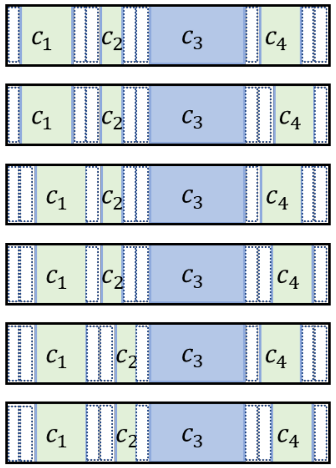
- **Proposed flow**
  - Take placement as inputs,
  - Propose a methodology to separate PA/RC prediction from a given machine learning-based prediction model.
  - Propose a separate **PA**/RC resolution scheme.



# PA-related DRC Violations Resolution Scheme

- **Objective**
  - Refine placement to be *optimal* in terms of the prediction value from $Pred_{PA}$
- **Process**
  - 1. Define a region to optimize.
  - 2. Encode possible whitespace distribution of a row in the region.



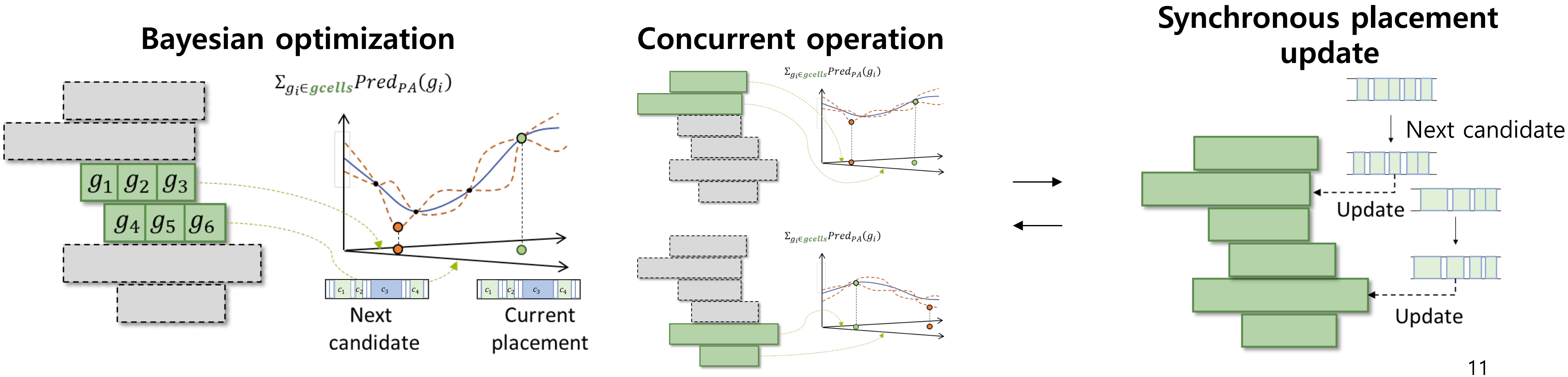




| Whitespace distribution | Placement | Encoded value |
|---|---|---|
| [1, 2, 2, 1, 2] | | 1 |
| [1, 2, 2, 2, 1] | | 2 |
| [2, 1, 2, 1, 2] | | 3 |
| [2, 1, 2, 2, 1] | | 4 |
| **[2, 2, 1, 1, 2] (Original)** | | 5 |
| [2, 2, 1, 2, 1] | | 6 |

# PA-related DRC Violations Resolution Scheme

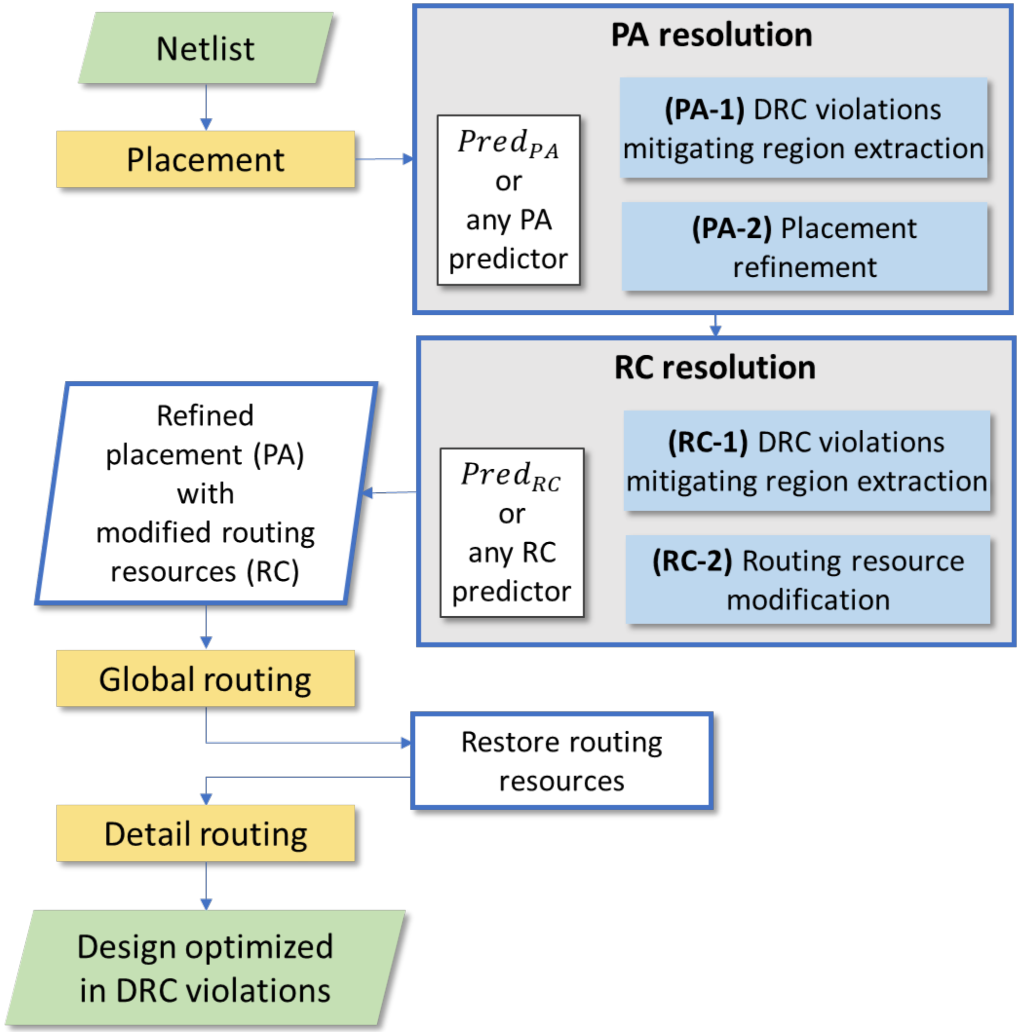
## Process

- 1. Define a region to optimize.
- 2. Encode placement for each row.
- 3. Perform concurrent Bayesian Optimization in units of 5 rows.
  - Update placements synchronously.
- 4. Repeat 1-3 until (1) predicted value < threshold or (2) maximum # iterations is reached.



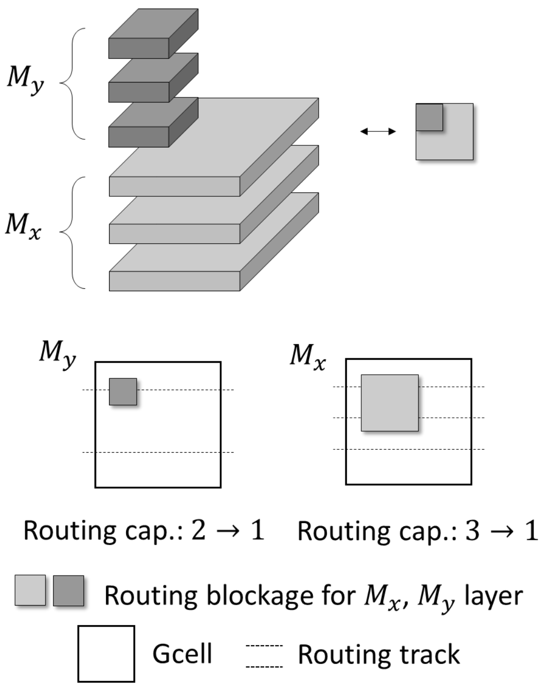
# Methodology

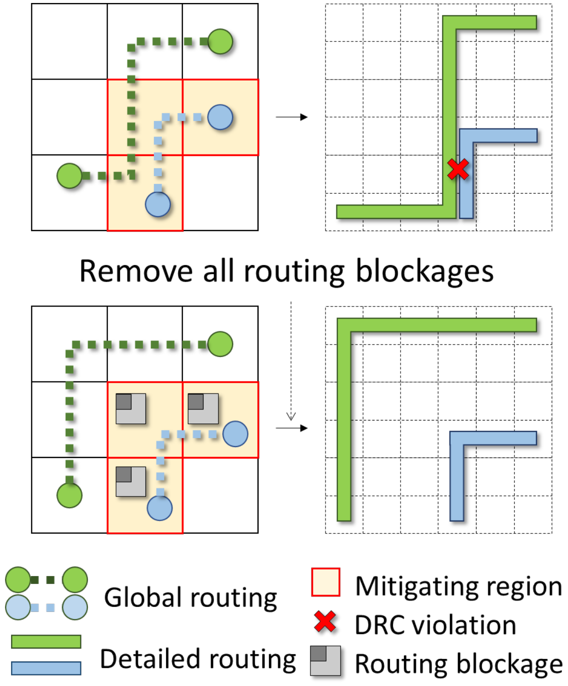
- **Proposed flow**
  - Take placement as inputs,
  - Propose a methodology to separate PA/RC prediction from a given machine learning-based prediction model.
  - Propose a separate PA/**RC** resolution scheme.



# RC-related DRC Violations Resolution Scheme

- **Suggested method in previous studies [3]**
  - Manipulating net/via weight to detour congested region
    - ⇒ Not applicable without internal tool access.
- **Proposed method**
  - Manipulate gcell capacity in DRC hotspot predicted by $Pred_{RC}$ using routing blockage.




[3] Chen, Jingsong, et al. (2020) "PROS: A plug-in for routability optimization applied in the state-of-the-art commercial EDA tool using deep learning." ICCAD.

# Experimental Results

## # DRC violations

- ECO router: 1.26% / [1]: 6.20% / [1], PA+RC: 23.62% / [4] (PA+RC), PA+RC: 26.54%
- Integrated resolution schemes can synergize with each other.

| # DRC | Baselines | | | Ours | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Predictor** | - | - | [1] | [1] | | | [4] (PA) | [4] (RC) | [4] (PA+RC) |
| **Resolution** | - | ECO router | [1] | PA | RC | PA + RC | PA | RC | PA + RC |
| B18 | 331 | 297 | 290 | 323 | **199** | 200 | 224 | 267 | 297 |
| LDPC | 277 | 191 | 277 | **167** | 280 | **167** | 189 | 277 | 189 |
| ETH | 1583 | 1590 | 1477 | 1276 | 1275 | 1276 | 1487 | **1063** | 1155 |
| B19 | 863 | 857 | 852 | 752 | 822 | 752 | 860 | **704** | 747 |
| ECG | 803 | 760 | 775 | 569 | 486 | 569 | 752 | **474** | **474** |
| AES | 1101 | 1072 | 1071 | 980 | 1018 | 980 | 1011 | 1013 | **953** |
| TATE | 3627 | 3244 | 3386 | 3123 | **2370** | **2370** | 3041 | 2894 | 2958 |
| JPEG | 5308 | 5707 | 4903 | 4802 | 4542 | 4298 | 4765 | 4142 | **3433** |
| **Ratio** | 1 | 0.9874 | 0.9380 | 0.8632 | 0.7912 | 0.7638 | 0.8874 | 0.7798 | **0.7346** |
| Impr. | | 1.26% | 6.20% | 13.68% | 20.88% | 23.62% | 11.26% | 22.02% | 26.54% |

[1] Chan, Wei-Ting J., et al. (2017) "Routability optimization for industrial designs at sub-14nm process nodes using machine learning." ISPD.
[4] Baek, Kyeonghyeon, et al. (2022) "Pin accessibility and routing congestion aware drc hotspot prediction using graph neural network and u-net." ICCAD

# Experimental Results

- **Other design metrics (Wirelength, worst slack, and total power)**
  - Wirelength: Our RC increases wirelength by 1% in average.
  - Worst negative slack: Not degraded by our methodology.
  - Total power: increased by 0.5% in average.

| Average | ICC2 | [1] | [4] + Our PA | [4] + Our RC | [4] + Our PA+RC |
|---|---|---|---|---|---|
| **Wirelength** increment (↓ , the better) | 0.00% | -0.06% | 0.03% | 1.02% | 0.98% |
| **Worst negative slack** increment (↑ , the better) | 0.00% | -15.81% | 9.65% | 3.15% | 4.37% |
| **Total power** increment (↓ , the better) | 0.10% | 0.43% | 0.41% | 0.62% | 0.50% |

# Conclusion

- **This work proposed DRC violations resolution model that can be tightly coupled with the prediction model.**
- **We developed two different methodologies that were able to be selectively applied to PA/RC related DRC violations.**
- **We reduced # DRC violations by 26.54% compared to the conventional flow with a cost of 0.5% degradation in total power.**

# THANKS FOR WATCHING!
# Q&A?