Invited: Artificial Netlist Generation for Enhanced Circuit Data Augmentation

2025.03.18

Seonghyeon Park

Pohang University of Science and Technology
Department of Electrical Engineering
CAD and SoC Design Lab

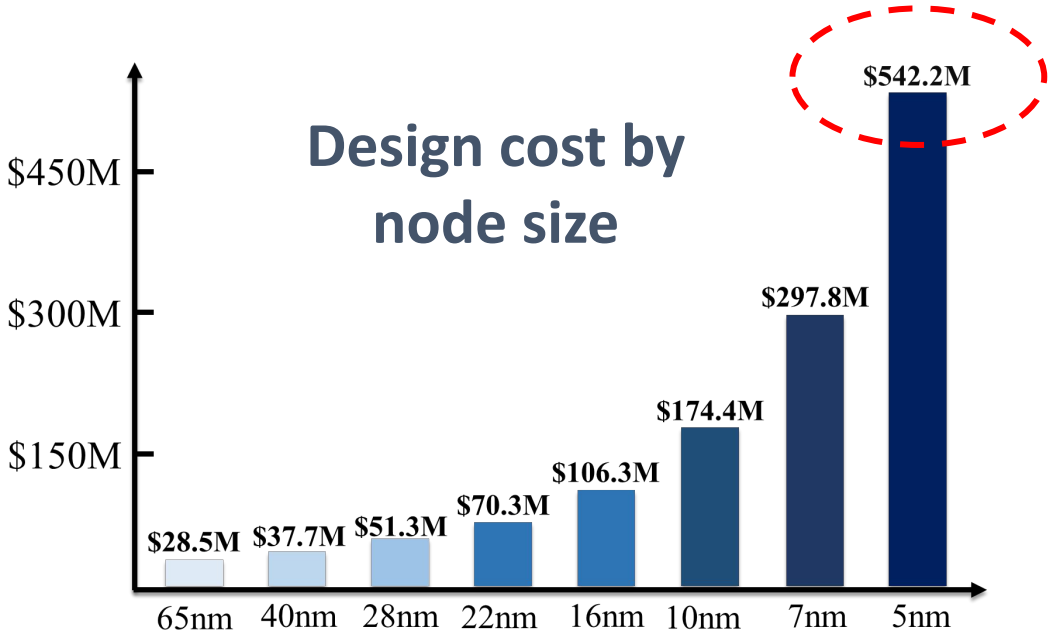
Outline

- Introduction
- Netlist Generator for Suboptimality
- Artificial Netlist Generator for Routability Prediction
- RTL Code Augmentation for LLMs
- Discussion

Introduction

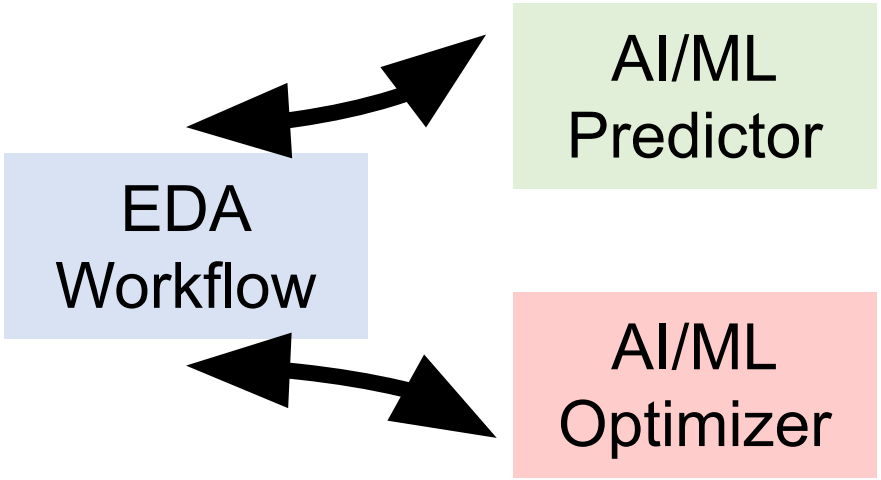
AI/ML-EDA: Dawn of New Era?

- Too many design rules and transistors **(design crisis due to huge TAT)**
- AI/ML can be a breakthrough to enhance traditional EDA workflow w/ **prediction** and **optimization**



*International Business Strategy Corporation (2014)

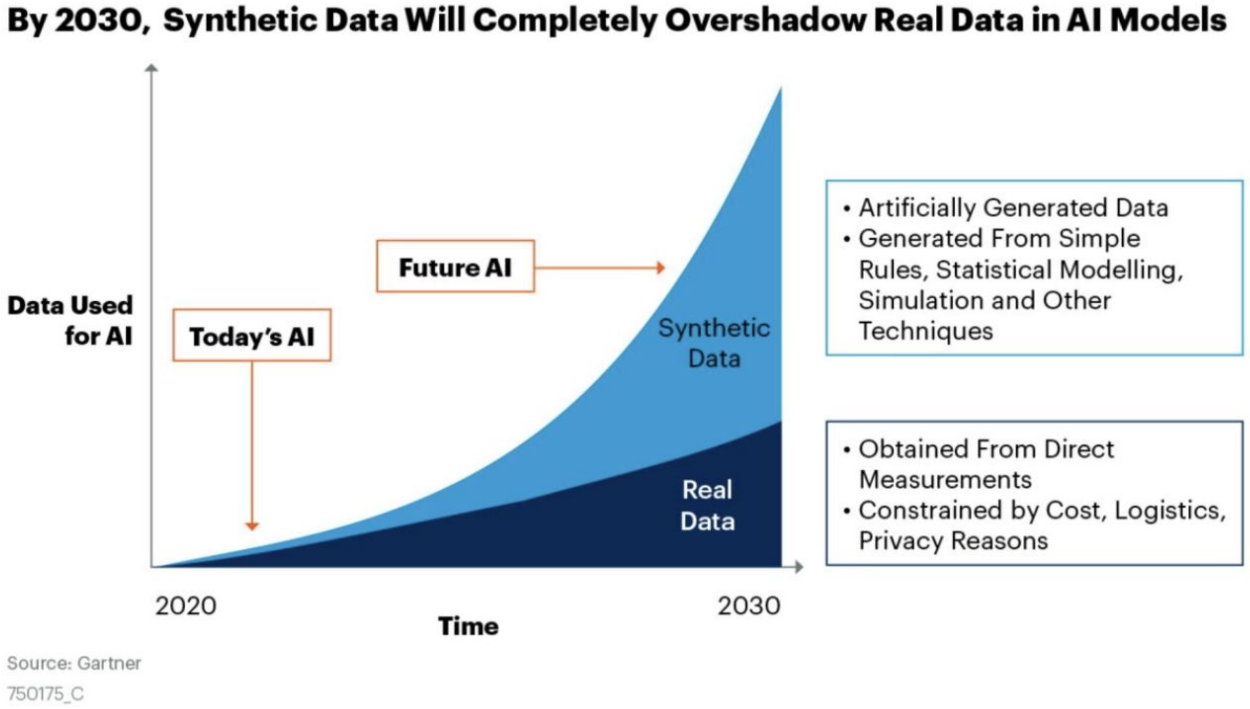
Predict design quality & prune design space

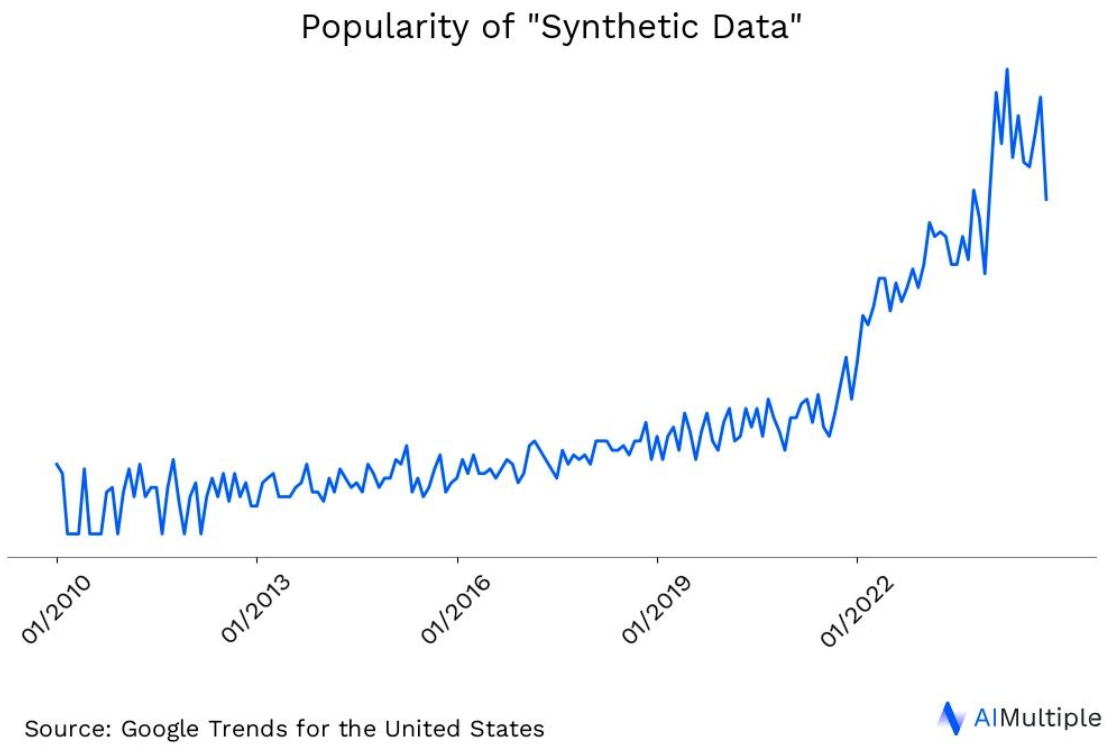


Improve QoR w/ learning-based opt.

Key Idea: Synthetic Data

- Real chip design data is sparse and expensive!
- Synthetic data can save cost and improve ML model quality

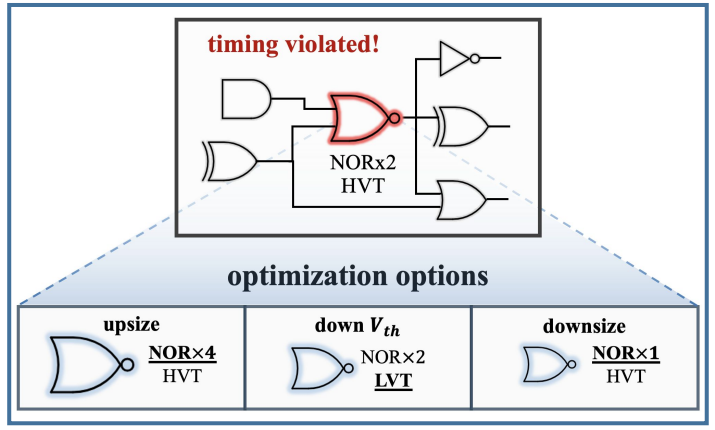




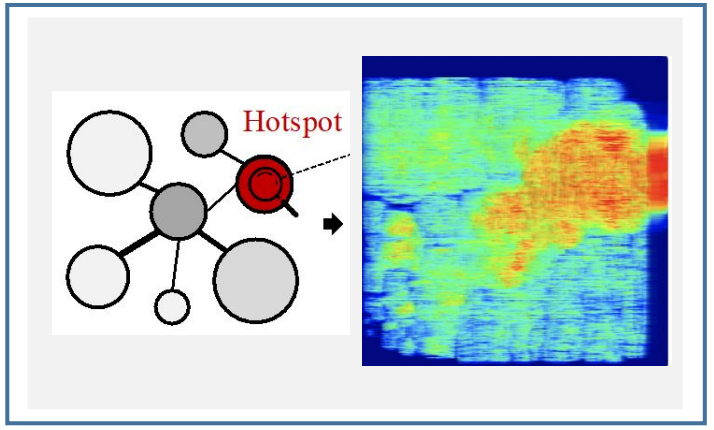
https://research.aimultiple.com/synthetic-data/
https://www.forbes.com/sites/robtoews/2022/06/12/synthetic-data-is-about-to-transform-artificial-intelligence/?sh=758208b97523

Usage of Artificial Design Generators

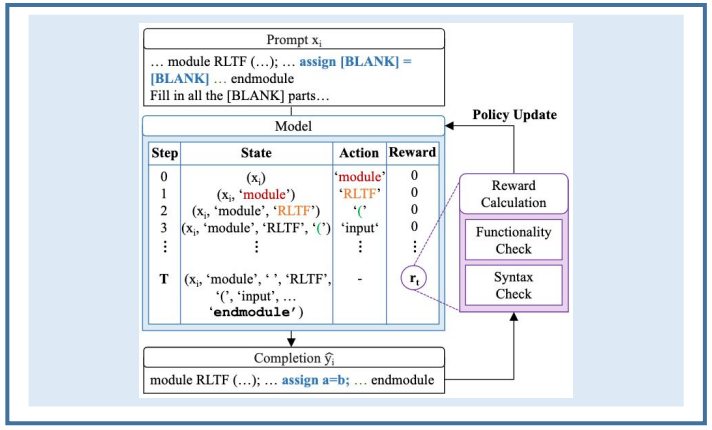
1. Timing Opt



2. Routability Opt

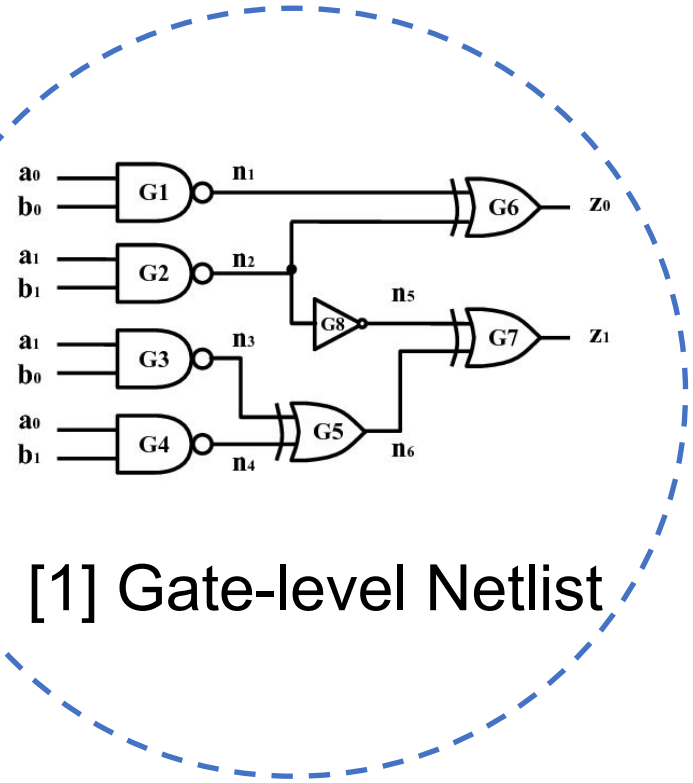


3. RTL Generation

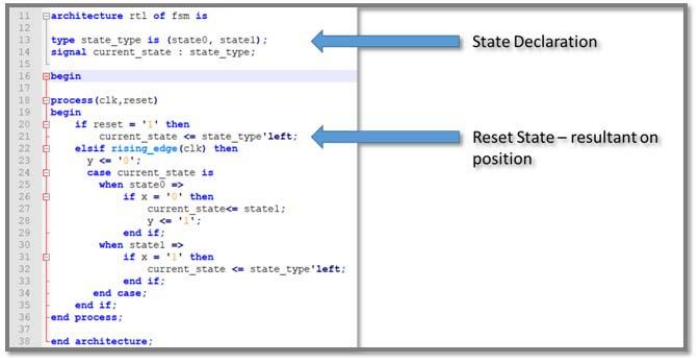


Synthetic Data in EDA

- Data types that can be used in EDA



[1] Gate-level Netlist



[2] RTL Codes

https://bluepearlsoftware.com/issue-8-rtl-coding-space-high-reliability-applications/
https://www.researchgate.net/figure/The-gate-level-netlist-of-post-synthesized-and-mapped-2-bit-multiplier-over-GF-2-2_fig3_310441143

Netlist Generation for suboptimality

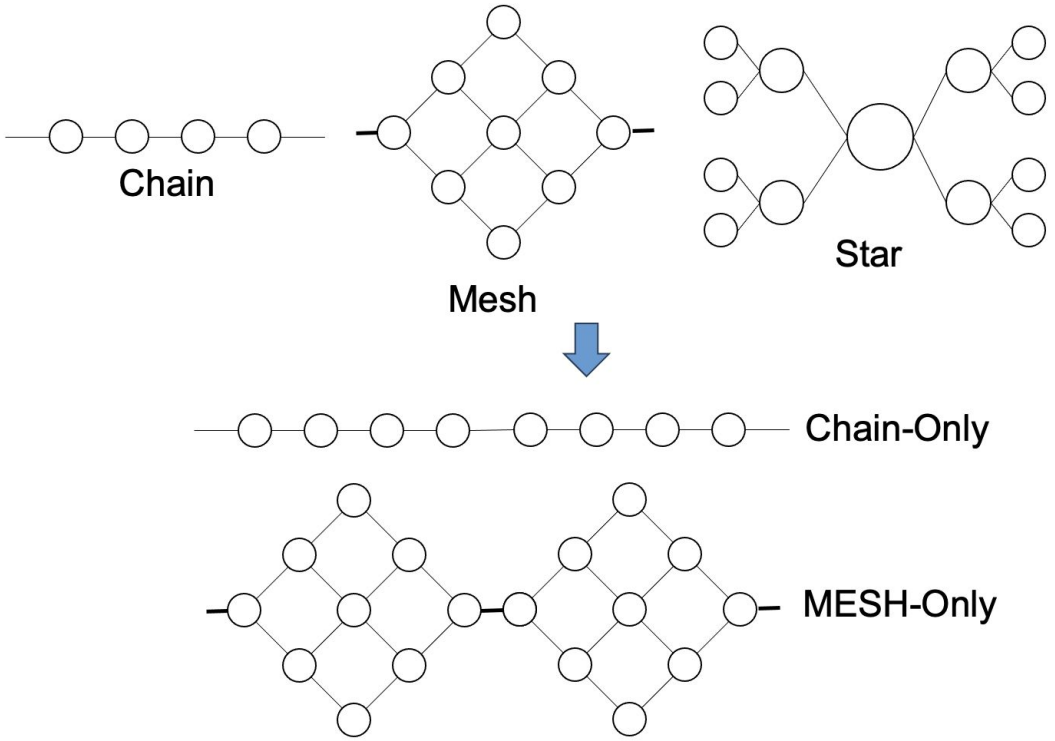
[DAC 2010] *Eyecharts: Constructive Benchmarking of Gate Sizing Heuristics*

[ISPD 2012] *Construction of Realistic Gate Sizing Benchmarks With Known Optimal Solutions*

[DAC 2010]

Suboptimality Evaluation of Gate Sizing

- **Eyechart**
- Built from three basic topologies, optimally sized with DP – allow suboptimalities to be evaluated
- A variety of eyecharts are generated by varying
  - Circuit topology
  - Power-size, delay-size characteristics of library
- Evaluate suboptimalities of existing heuristics studied under these variations

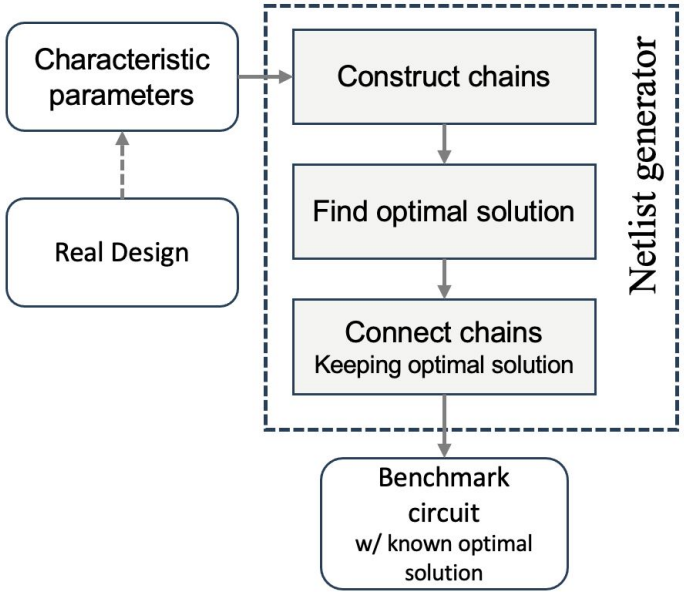


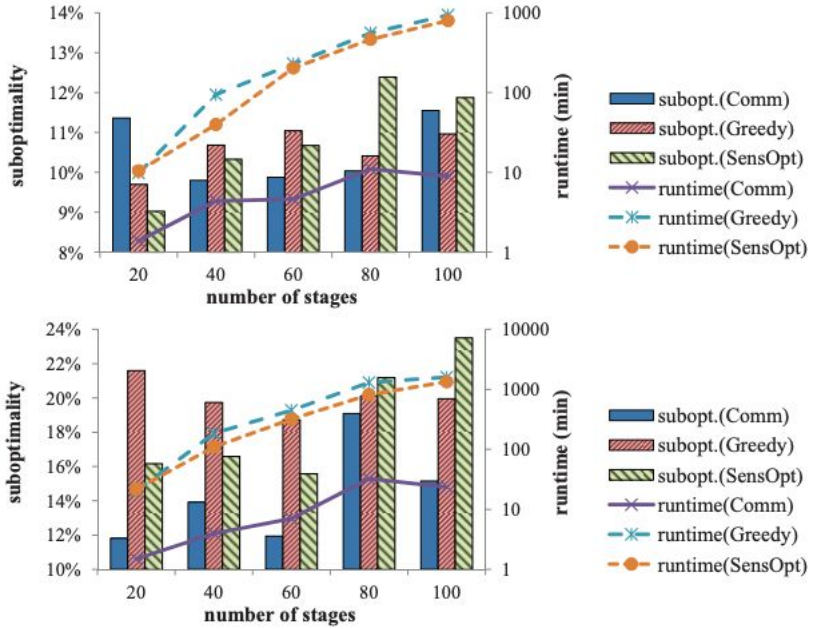
Gupta et al., “Eyecharts: Constructive Benchmarking of Gate Sizing Heuristics”, DAC 2010.

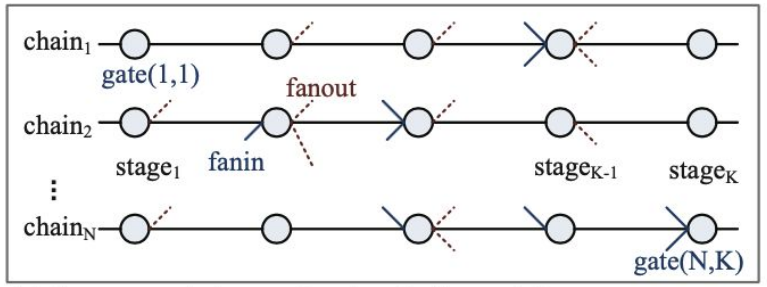
[ISPD 2012]

Suboptimality Evaluation of Gate Sizing

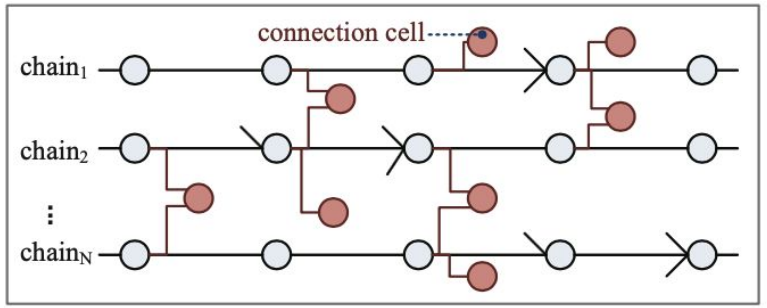
- **NetGen**
- Improve Eyechart circuits' different topology from real design
- The benchmarks resemble real designs w/ **known optimal solution**



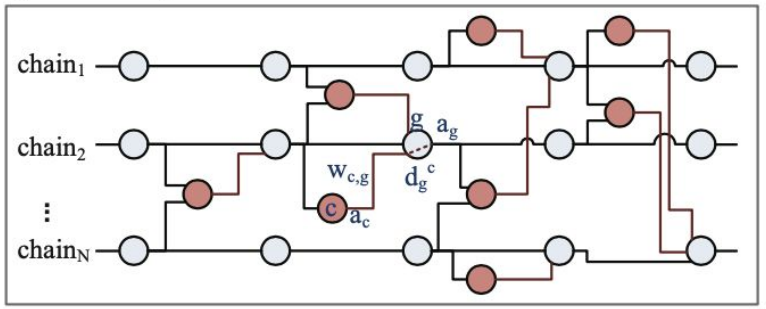




(a) Construct chains, and assign fanins and fanouts.



(b) Attach connection cells, and execute DP to determine optimal sizing.

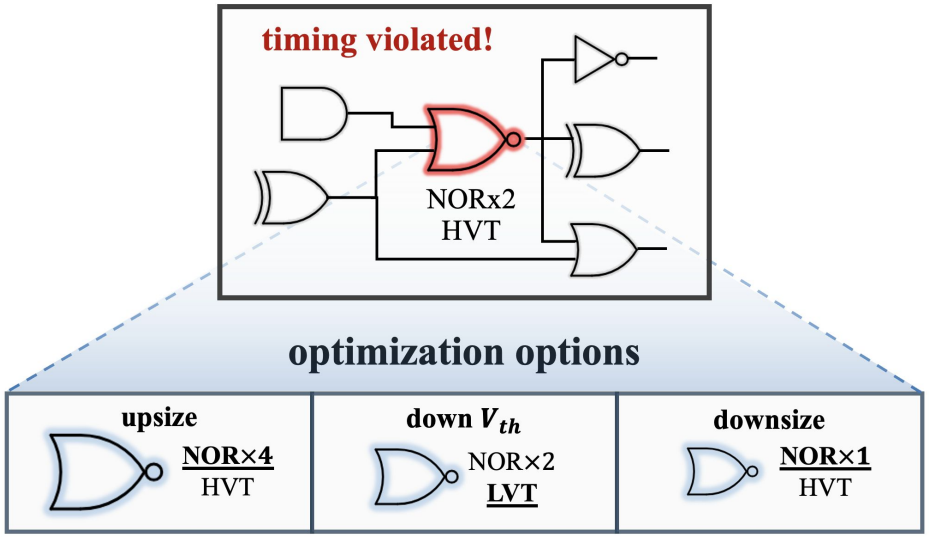


(c) Connect chains subject to the (setup) timing constraint.

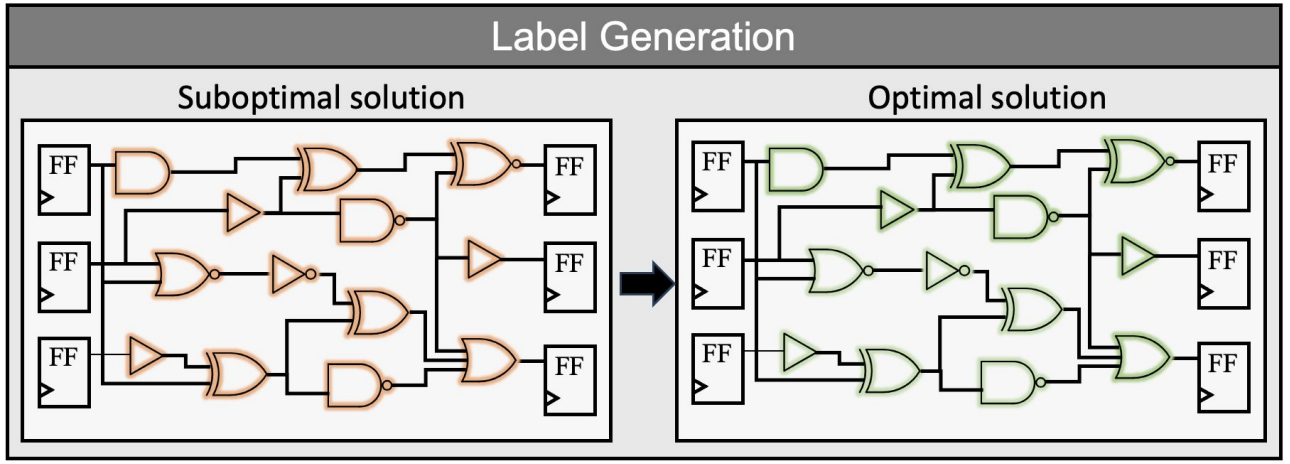
A. B. Kang and S. Kang "Construction of Realistic Gate Sizing Benchmarks with Known Optimal Solutions", ISPD 2012.

Timing Optimization w/ Known Optimal Solution

- Various opt. methods (e.g., gate sizing, Vth swap) are applied sequentially and causes suboptimality
- If we know the optimal solution, we can improve supervised-learning based gate sizer.



Available optimization options to relieve timing violation



Gate-Level Netlist Generation for ML

[DATE 2021] *Machine Learning Framework for Early Routability Prediction with Artificial Netlist Generator*

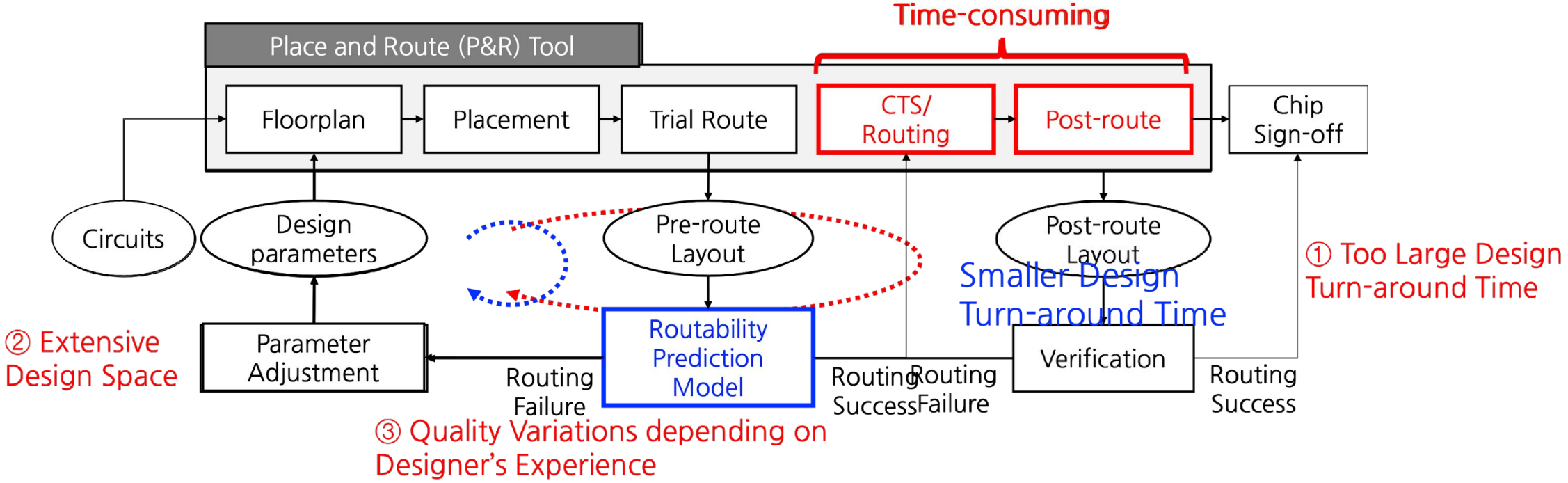
[TCAD 2023] *Construction of Realistic Place-and-route Benchmarks for Machine Learning Applications*

[ICCAD 2023] *Routability Prediction and Optimization Using Explainable AI*

Motivation: ML for Routability Prediction

- ML models are trained from data
  - No need for human involvement in the modeling process
- Neural networks can identify the connections between high-dimensional input data and output data
  - Higher predictive accuracy performance

How to Mitigate These Problems?



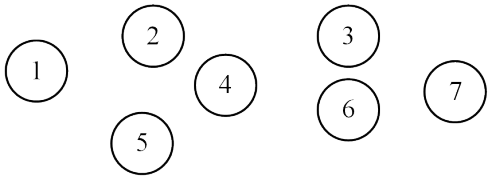
Artificial Netlist Generator (ANG)

[DATE 2021], [TCAD2023]

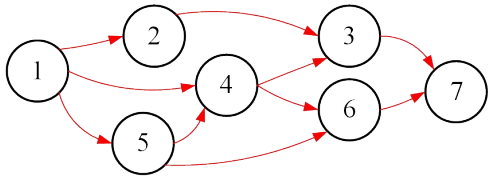
- **Exploration of topological shape of circuit**

Size of circuit: Number of instances (N_{inst}), Number of primary I/Os (N_{prim})
Interconnect complexity: Average net degree (D_{avg}), Average net bounding box (B_{avg})
Timing delay: Average depth of timing paths (T_{avg})

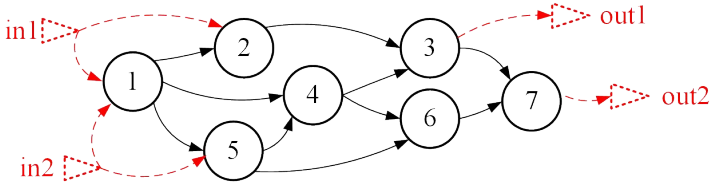
topological parameters



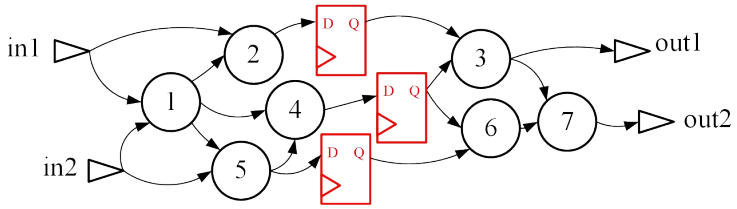
(1) initialized graph



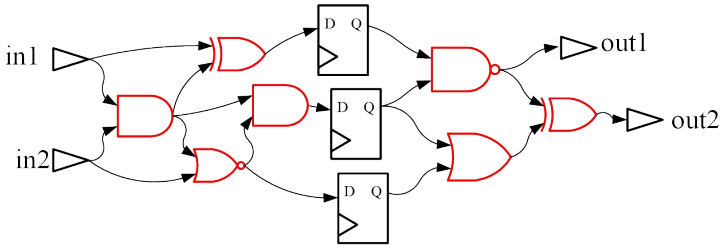
(2) distribution matching



(3) generate primary I/Os



(4) construct timing paths

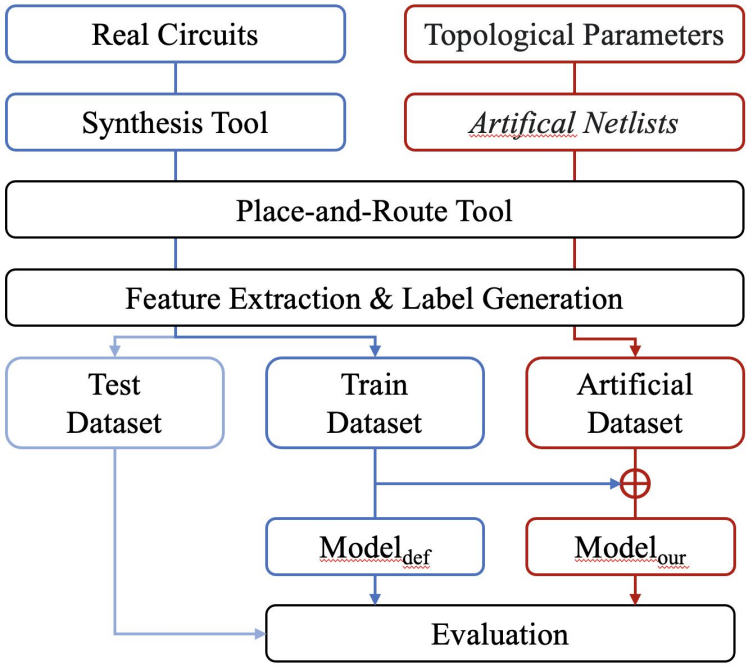


(5) map standard cells

Gate-Level Netlist Generation

[DATE 2021], [TCAD2023]

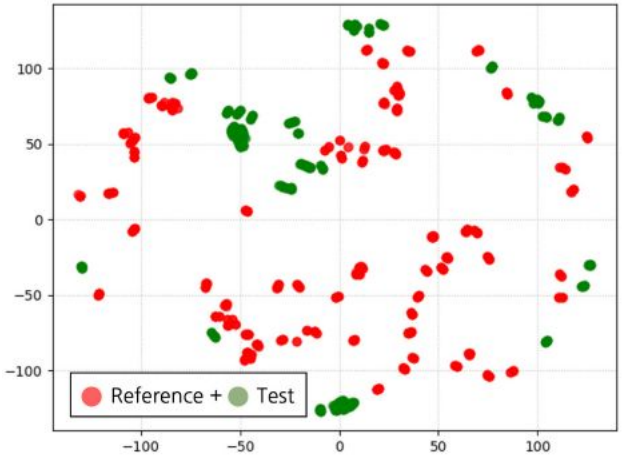
Design of Experiments



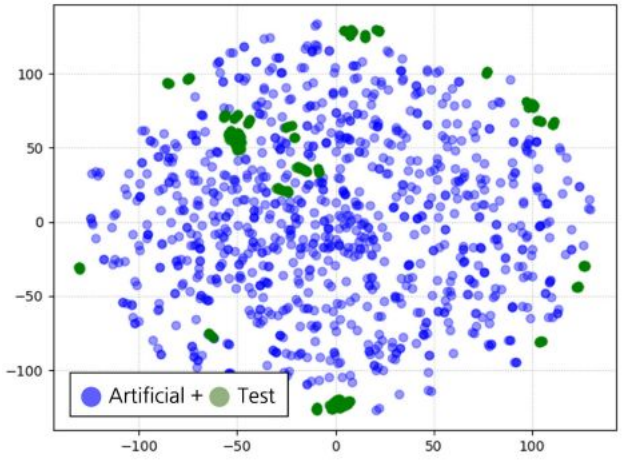
	Bounded circuit space	Statistics of artificial circuits			
		Min	Max	Avg.	Std.
N_{inst}	$[200, 80000]$	276	75649	22278	20292
N_{prim}	$[20, 5000]$	22	4636	504	954
D_{avg}	$[2.5, 4.5]$	2.76	4.26	3.42	0.307
B_{avg}	$[0.1, 1.5]$	0.11	1.41	0.381	0.275
T_{avg}	$[2.0, 20.0]$	2.02	19.41	7.28	3.488
S_{ratio}	$[0.05, 0.30]$	0.05	0.30	0.182	0.067

320 artificial netlists

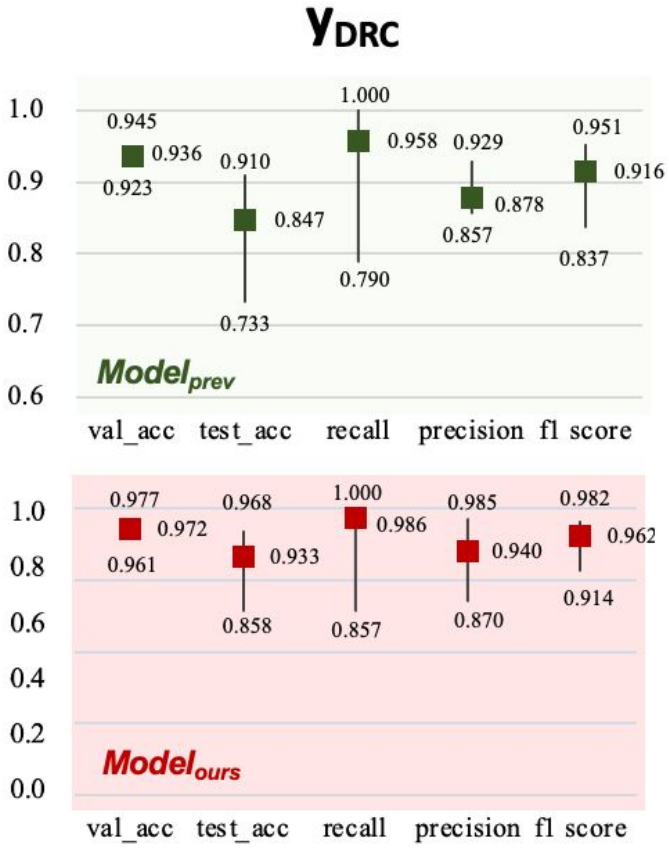
Data distributions



(a)



Model performance

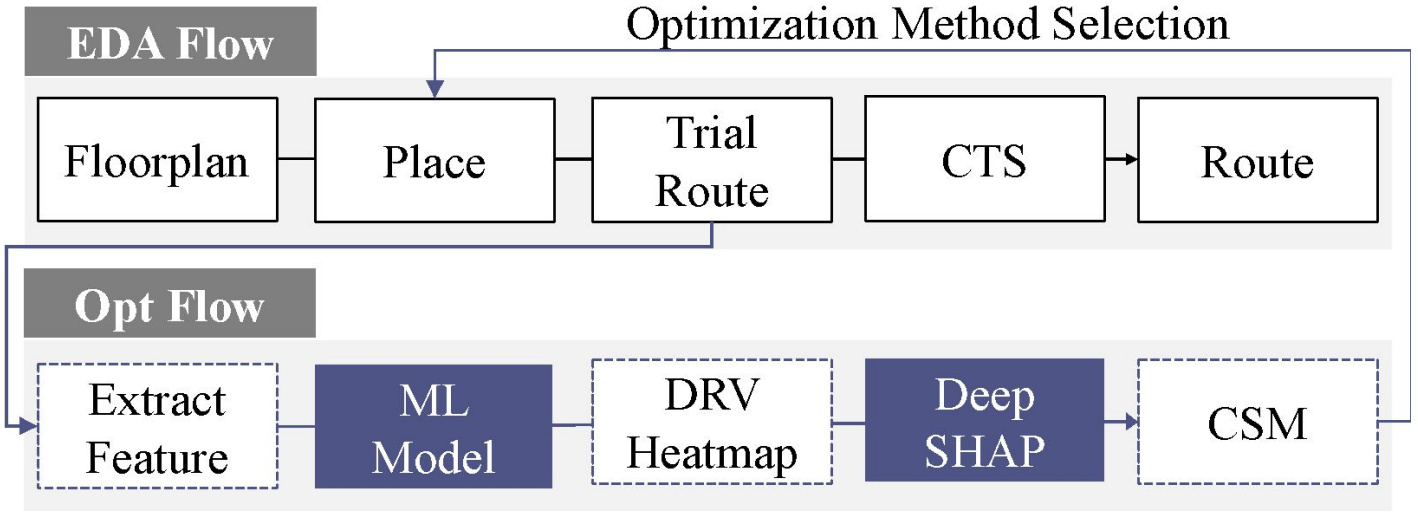


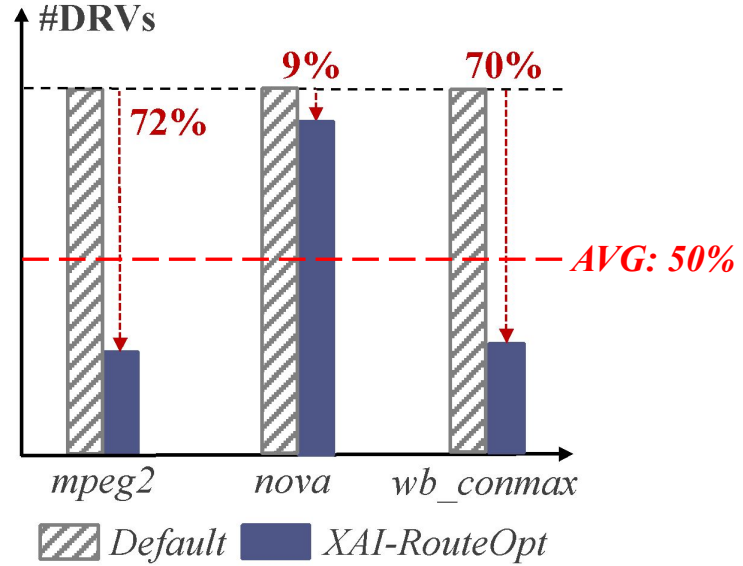
test accuracy **~9%** higher

F1 score **~5%** higher

DRV Prediction and Optimization: Flow and Results

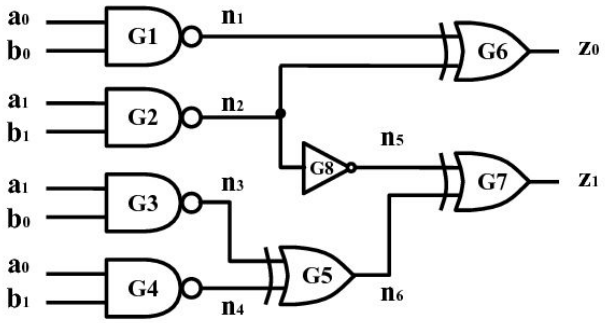
- (1) Train ML model with **ANG-augmented dataset**
- (2) Predict DRV heatmap based on ML model
- (2) Generate contribution score map with XAI to identify the key features causing DRVs
- (3) Perform optimization processes based on the identified key features



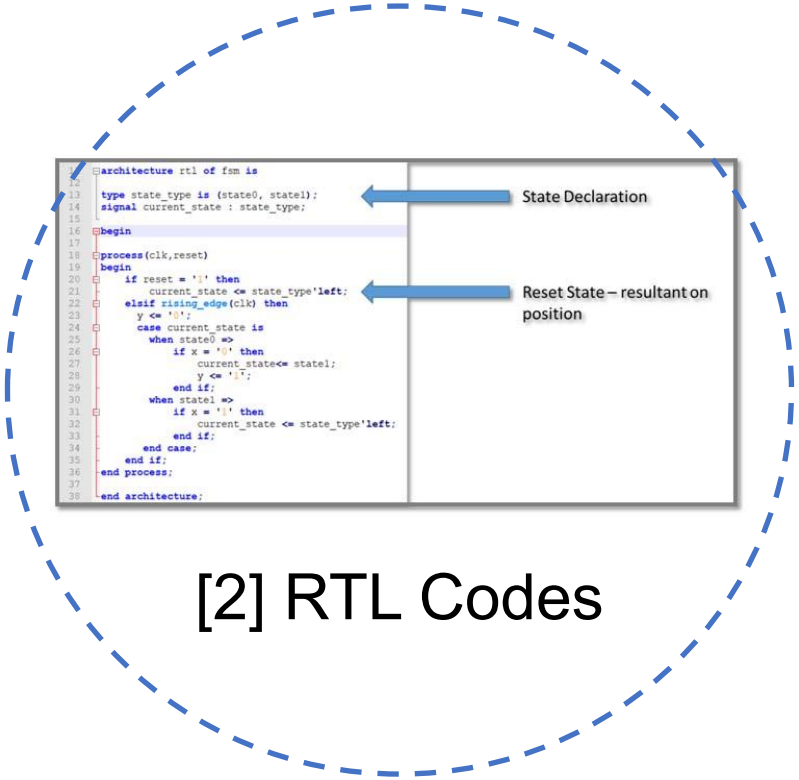


Synthetic Data in EDA

- Data types that can be used in EDA



[1] Gate-level Netlist



[2] RTL Codes

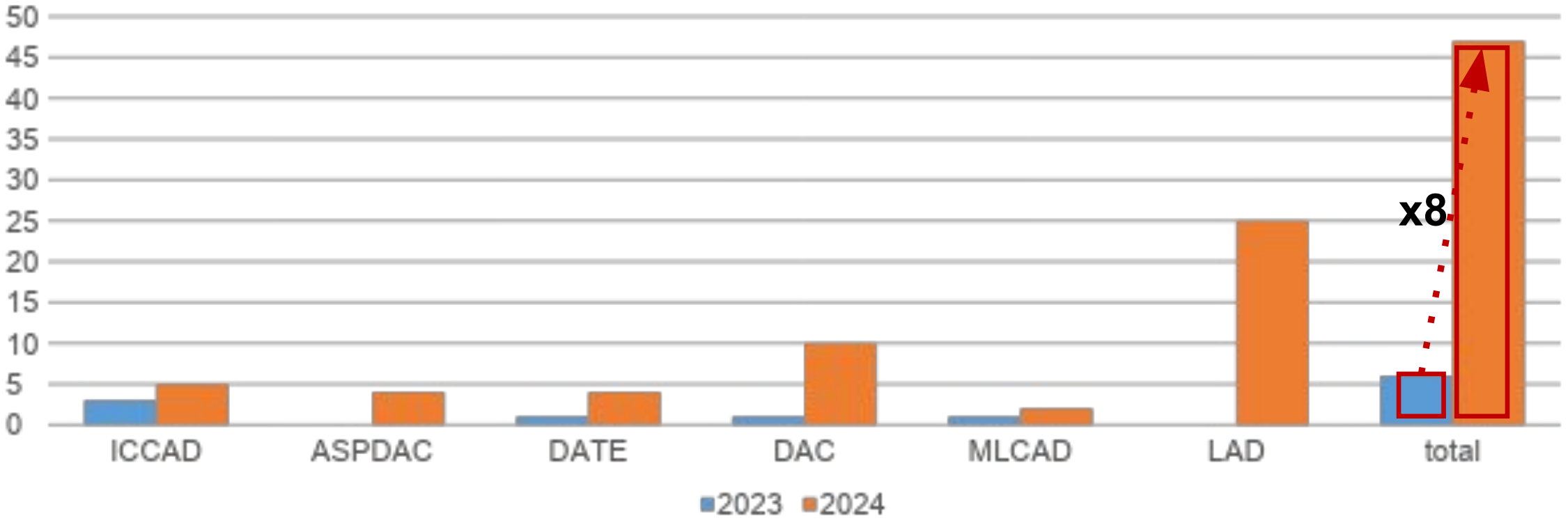
https://bluepearlsoftware.com/issue-8-rtl-coding-space-high-reliability-applications/
https://www.researchgate.net/figure/The-gate-level-netlist-of-post-synthesized-and-mapped-2-bit-multiplier-over-GF-2-2_fig3_310441143

RTL code Generation for LLMs

[DATE 2025] *Improving LLM-based Verilog Code Generation with Data Augmentation and RL*

Motivation: LLMs for EDA

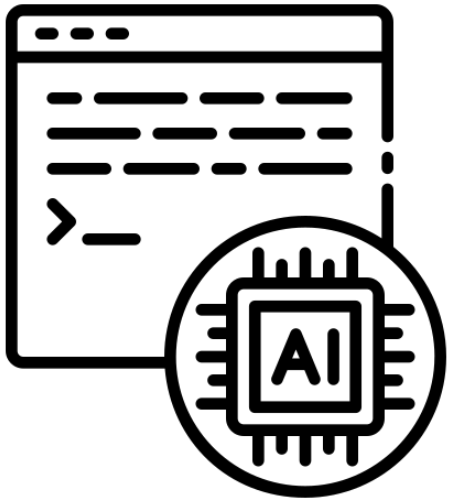
- **LLMs-related papers in EDA**



*The LAD conference has been around since '24.

Motivation: Challenges in LLM-based RTL Generation

Data Scarcity

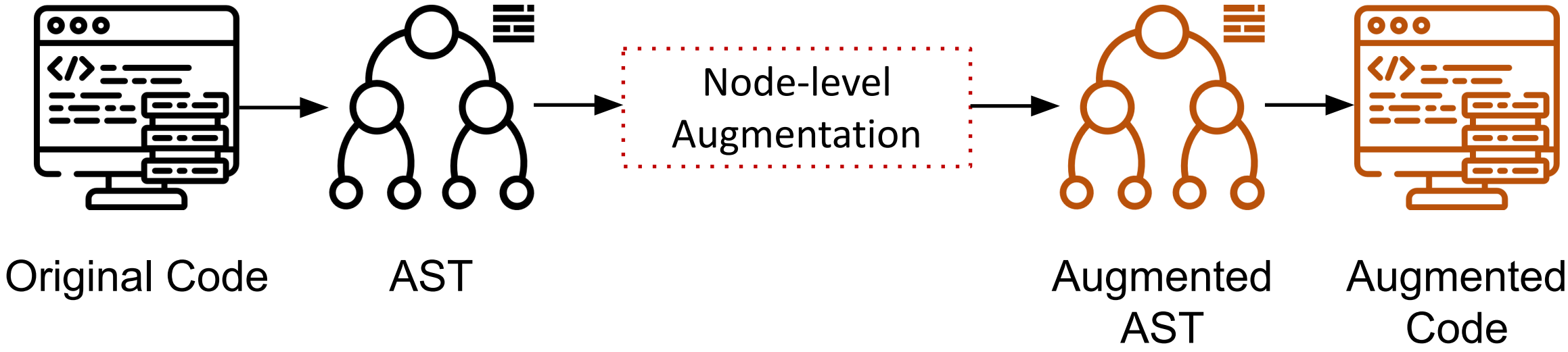


Prompt Generation

Fine-tuning Methodology

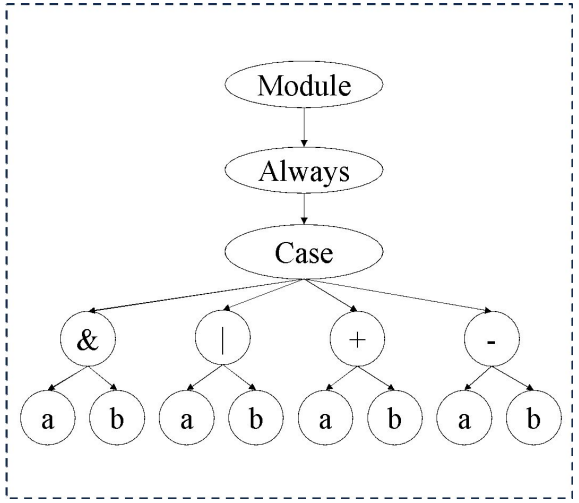
AST-based Data Augmentation

- **Augmentation of the code's syntactic structure**
  - How to represent the syntactic structure of RTL code?
    - Abstract Syntax Tree (AST)

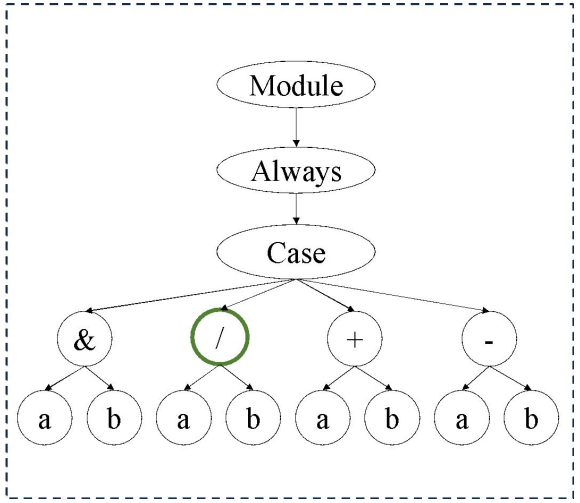


AST-based Data Augmentation

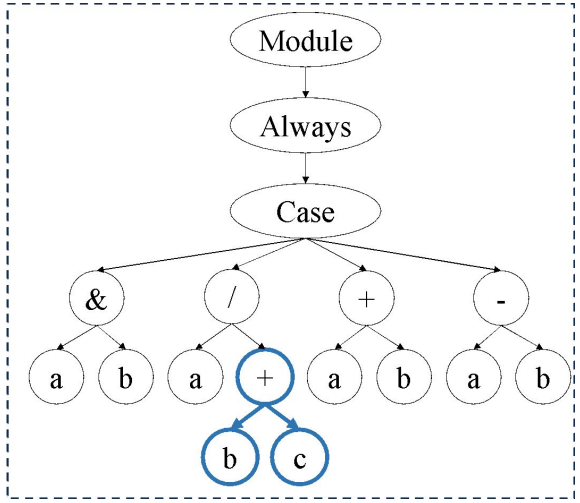
- **Augmentation of the code's syntactic structure**
  - Node-level Augmentation
    - Probabilistically perform augmentation while traversing nodes



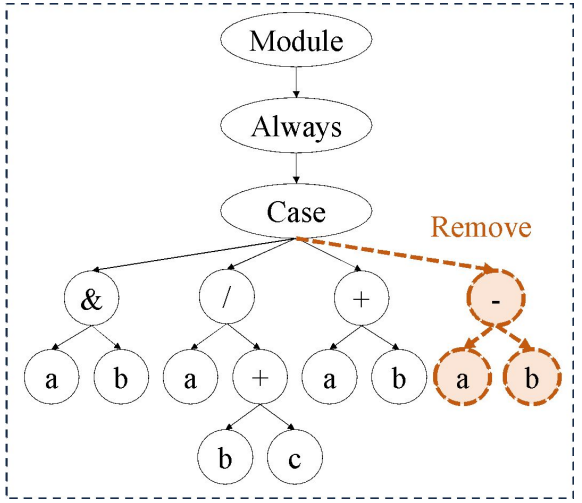
[Original AST]



[Modification]



[Insertion]



[Deletion]

AST-based Data Augmentation: Example

- **Augmentation of the code's syntactic structure**
  - Example of AST-based Code Augmentation

```
module top (
  input d, input clk, input pre, input clr,
  input en, output reg q);
  initial q <= 1'b0;
  always @(negedge clk or posedge pre or
negedge clr) if(pre) q <= 1'b1;
  else if(!clr) q <= 1'b0;
  else if(en) q <= d;
endmodule
```

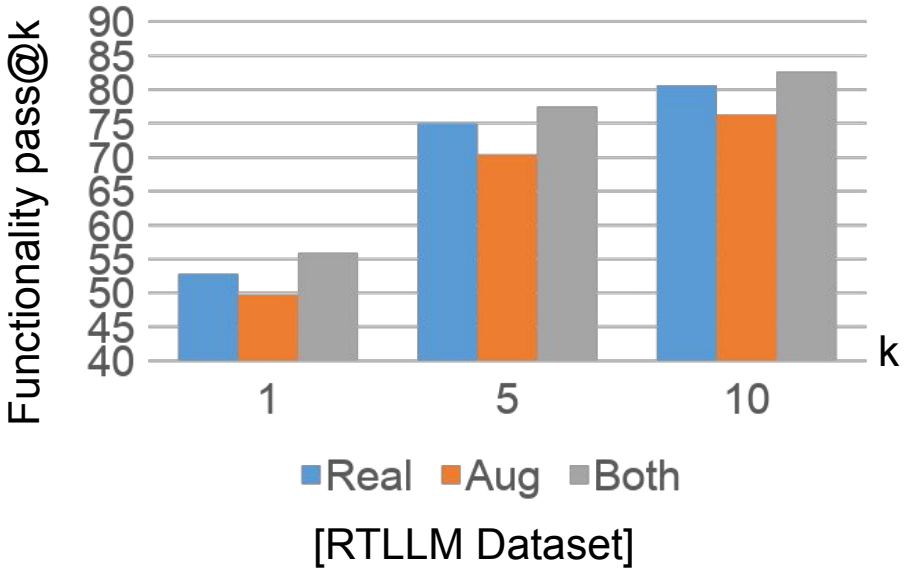
[Original Code]

```
module aug_top (
  input d, input clk, input pre, input clr, input en,
  output reg q, input wire x , input wire y ,
  input wire z , input wire w );
  initial q = ~x;
  always @(negedge clk or posedge pre or
negedge clr) if(pre) q <= y * z;
  else if(!clr) q <= -w;
  else if(en) q = !d;
endmodule
```

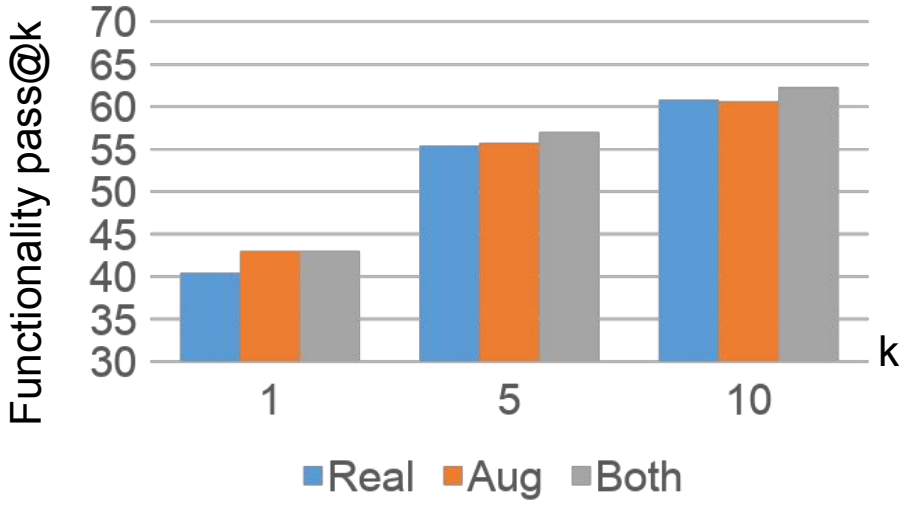
[Augmented Code]

RTL Code Generation with Augmented Code Data

- **Augmentation of the code's syntactic structure**
  - Evaluation of Augmented Code Data
    - Comparison of using only the real dataset (Real), only the augmented dataset (Aug), and both combined (Both)
    - In all cases, using both datasets yields the best performance



[RTLLM Dataset]

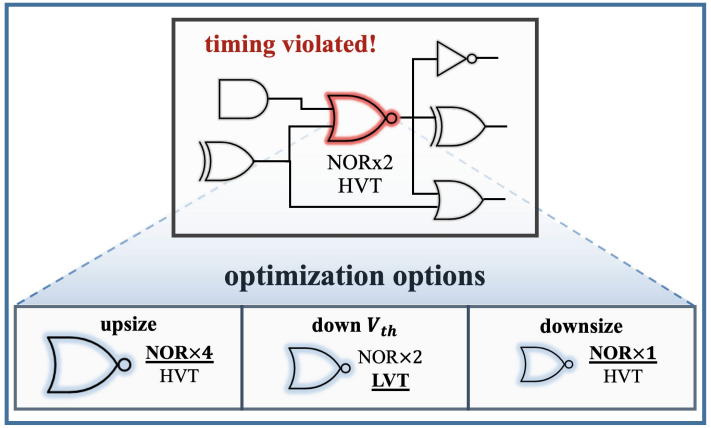


[VerilogEval Dataset]

Discussion

Challenges and Future Direction

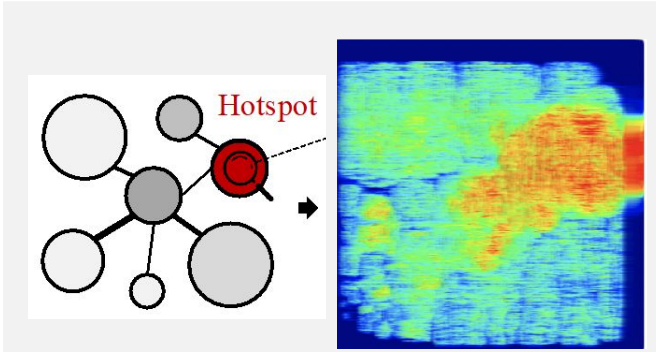
1. Timing Opt



Limitations: Specifically fitted for contest benchmark

Direction: Reduce gap between real and artificial design, Handle realistic delay model

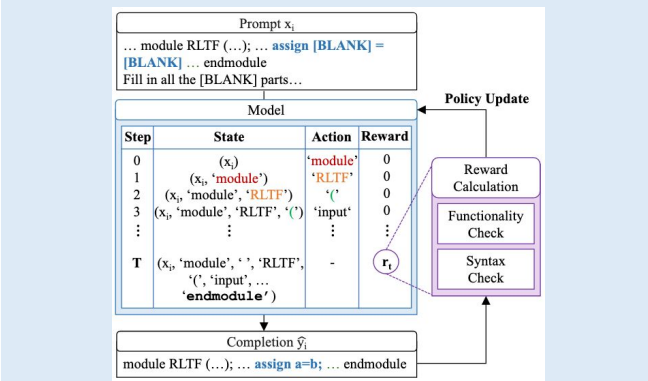
2. Routability Opt



Limitations: Lack of macro consideration, Hard to generate large designs

Direction: Support macros, Hierarchical design, Enable large design gen

3. RTL Generation

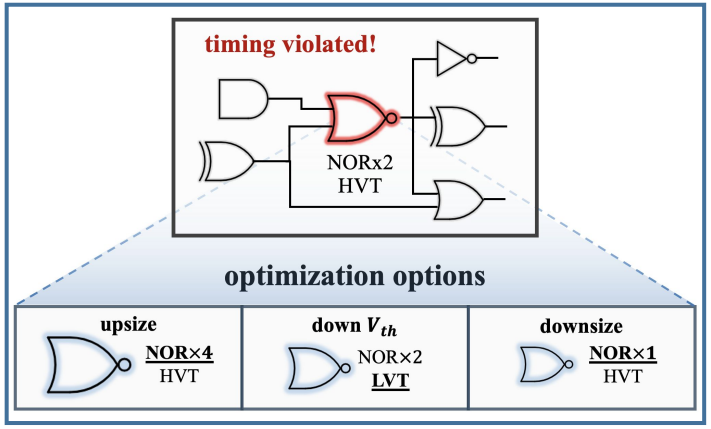


Limitations: Lack of functionality consideration, Highly dependent on the original RTL

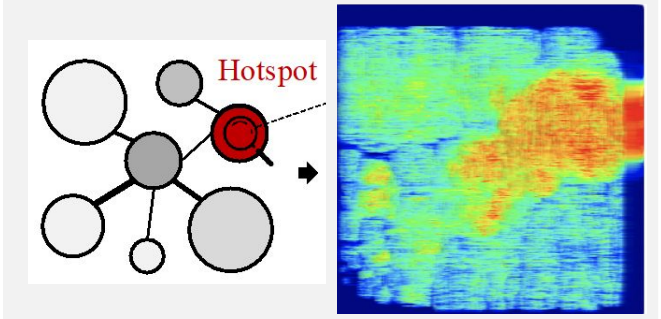
Direction: RTL augmentation with functionality consideration

Challenges and Future Direction

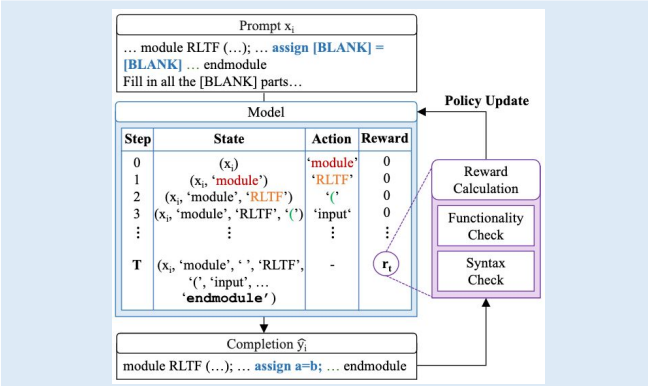
1. Timing Opt



2. Routability Opt



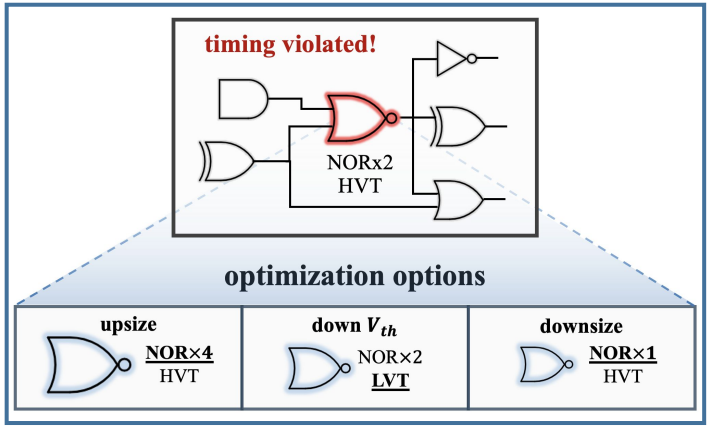
3. RTL Generation



	1. Timing Opt	2. Routability Opt	3. RTL Generation
Limitations	Specifically fitted for contest benchmark	Lack of macro consideration, Hard to generate large designs	Lack of functionality consideration, Highly dependent on the original RTL
Direction	Reduce gap between real and artificial design, Handle realistic delay model	Support macros, Hierarchical design, Enable large design gen	RTL augmentation with functionality consideration

Challenges and Future Direction

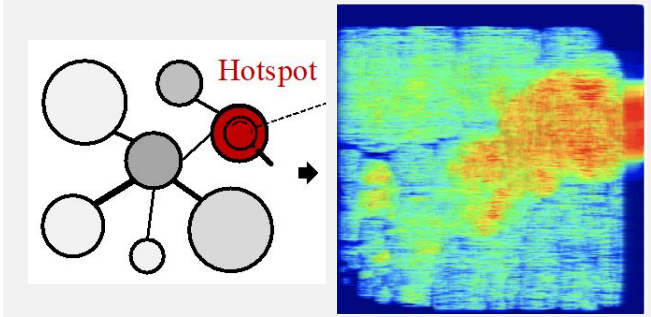
1. Timing Opt



Limitations: Specifically fitted for contest benchmark

Direction: Reduce gap between real and artificial design, Handle realistic delay model

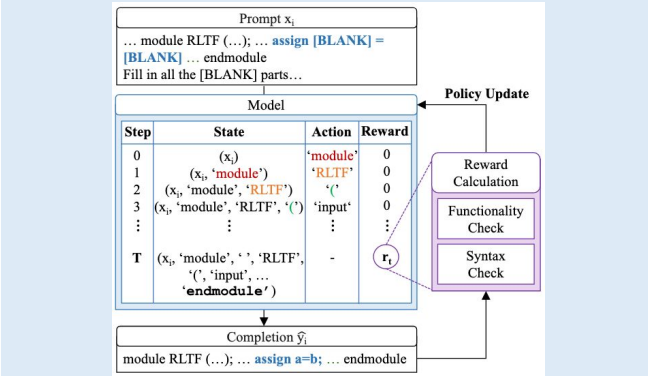
2. Routability Opt



Limitations: Lack of macro consideration, Hard to generate large designs

Direction: Support macros, Hierarchical design, Enable large design gen

3. RTL Generation



Limitations: Lack of functionality consideration, Highly dependent on the original RTL

Direction: RTL augmentation with functionality consideration

Thank you