HybMT: Hybrid Meta-Predictor based ML Algorithm for Fast Test Vector Generation

Shruti Pandey, Jayadeva and Smruti. R. Sarangi
Department of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi, India

29th Asia and South Pacific Design Automation Conference (ASP-DAC) 2024

Outline

- Introduction and Motivation
- Background
- Methodology
- Results and Conclusion

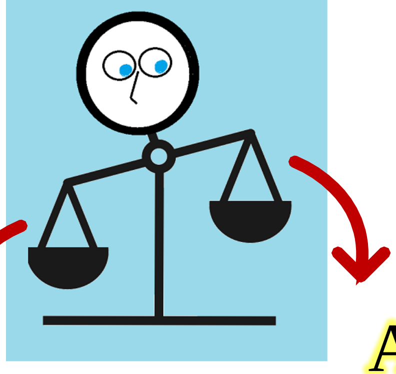
Problem Statement

Circuit

Fault list

→ Automatic Test Pattern Generation (ATPG) algorithm → Test vectors

$$Fault\ coverage = \frac{number\ of\ detected\ faults}{total\ number\ of\ faults}$$



Fault coverage

ATPG runtime

- Equivalent to the SAT problem

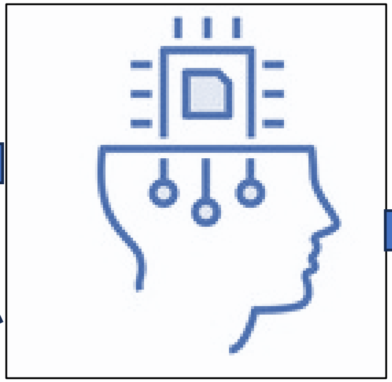
⇒ NP-complete

Problem Statement

NP-complete → Use traditional heuristics

Faster

More accurate



Machine Learning

Excellent decision-making

Minimize the **test vector** generation time using ML subject to **sufficient** fault coverage.

Pitfalls of Prior Work

Have not moved beyond the **4-decade old ISCAS ’85** benchmarks.

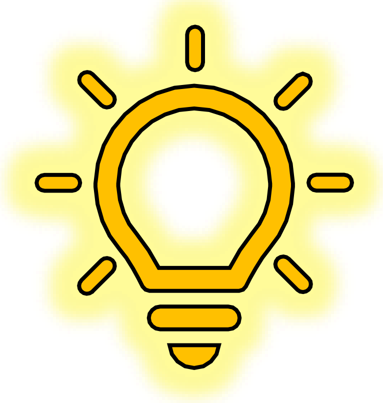
No comparisons with the **commercial tools.**

No work demonstrated **consistent improvement** across all the benchmark circuits.

HybMT Overcomes these Pitfalls

Generates test inputs for recently released benchmark circuits that are up to **70× larger** than the ISCAS’85 circuits.

Outperforms a popular commercial ATPG tool



Shows a consistent speedup across all the benchmark circuits.

Outline

Introduction and Motivation

Background

Methodology

Results and Conclusion

Fundamentals of Generating a Test

Fault-oriented ATPG for single stuck-at-faults (SSFs)

Generating a test for a fault ***s-a-v*** at net '*l*',

1. Fault Activation:

- To activate (excite) the fault.
- Activating the fault means setting *primary input* (PI) values that cause net '*l*' to have value $\overline{v}$.

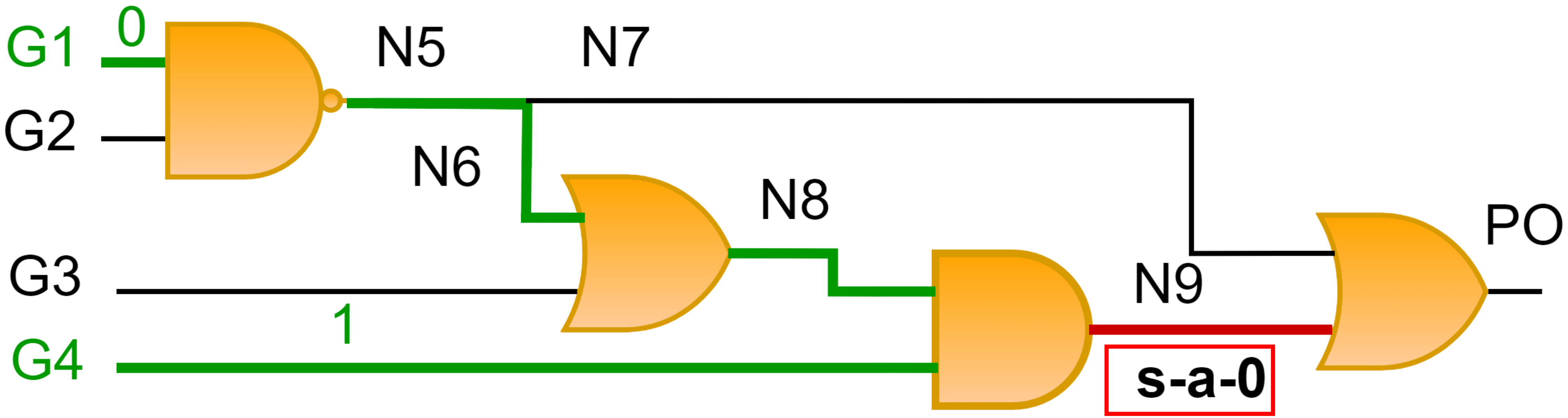
2. Fault Propagation:

- To propagate the fault to a *primary output* (PO).

PODEM: Path Oriented Decision Making

1. Backtrace path: **G4**

2. Backtrace path: **N8,N6,N5,G1**

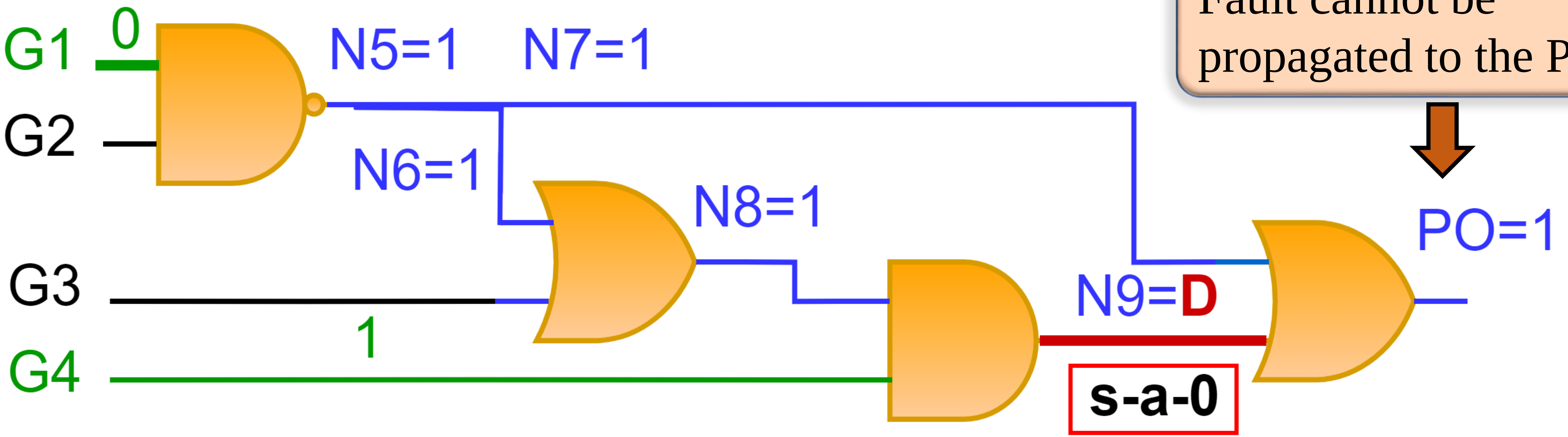


Objective to set 1 at N9

PODEM: Path Oriented Decision Making

A value of 0 at G1 implies **1 at N5,N6,N7,N8,PO** and a **D at N9**

Fault cannot be propagated to the PO



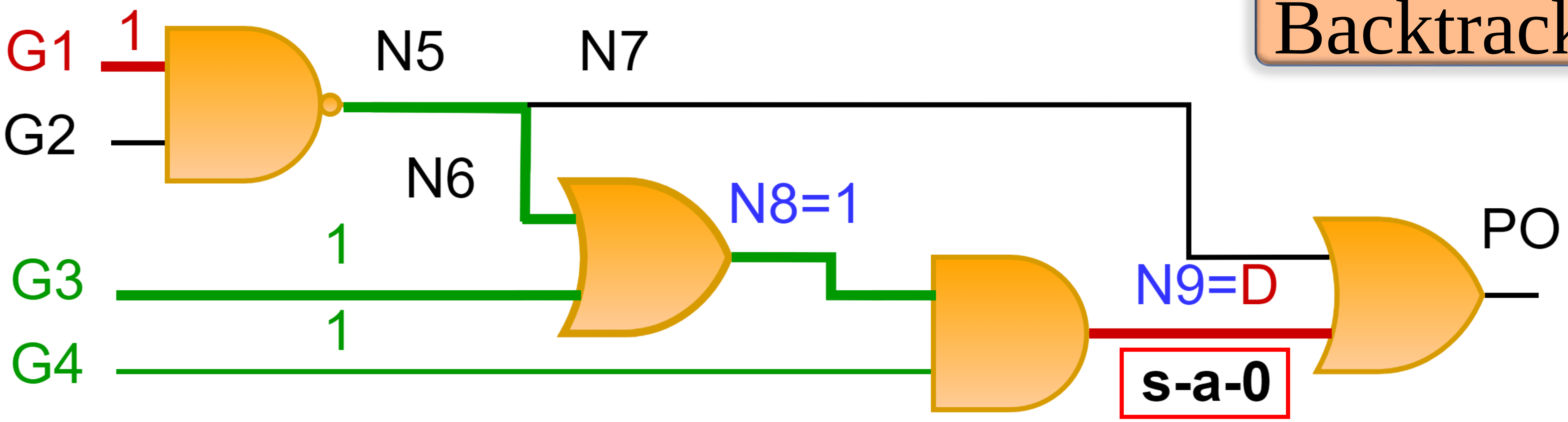
The symbol **D** denotes a composite value of 1/0 for the same net in good/faulty circuits.

Objective (N9,1) met

PODEM: Path Oriented Decision Making

We need to reverse our previous decision of setting 0 at G1.

Backtrack

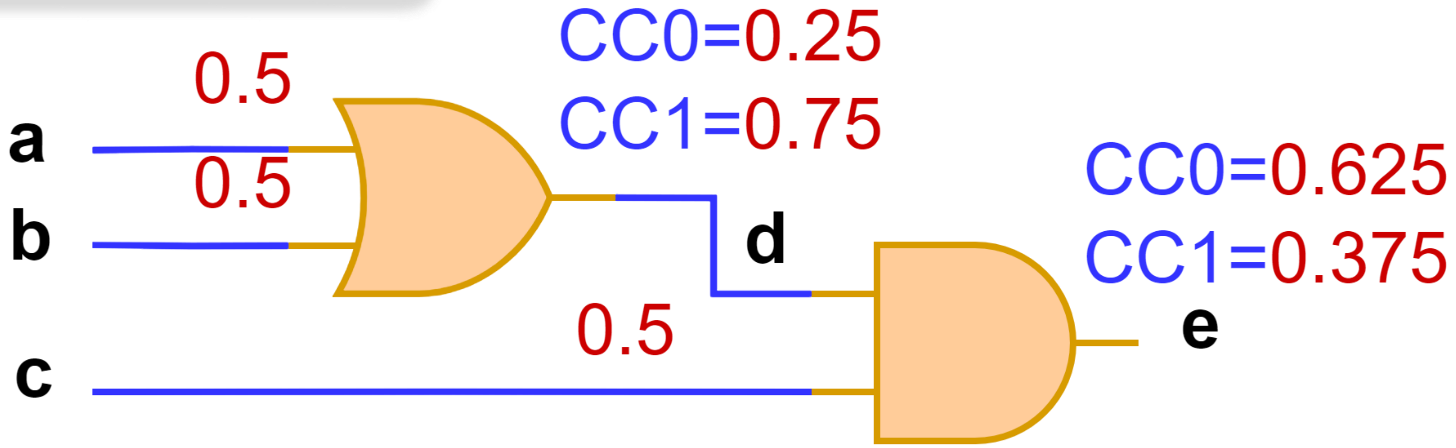


Make decisions such that number of backtracks is minimized.

Controllability

The probability of setting a net to 0 or 1 by making random PI assignments

$CC(x)$

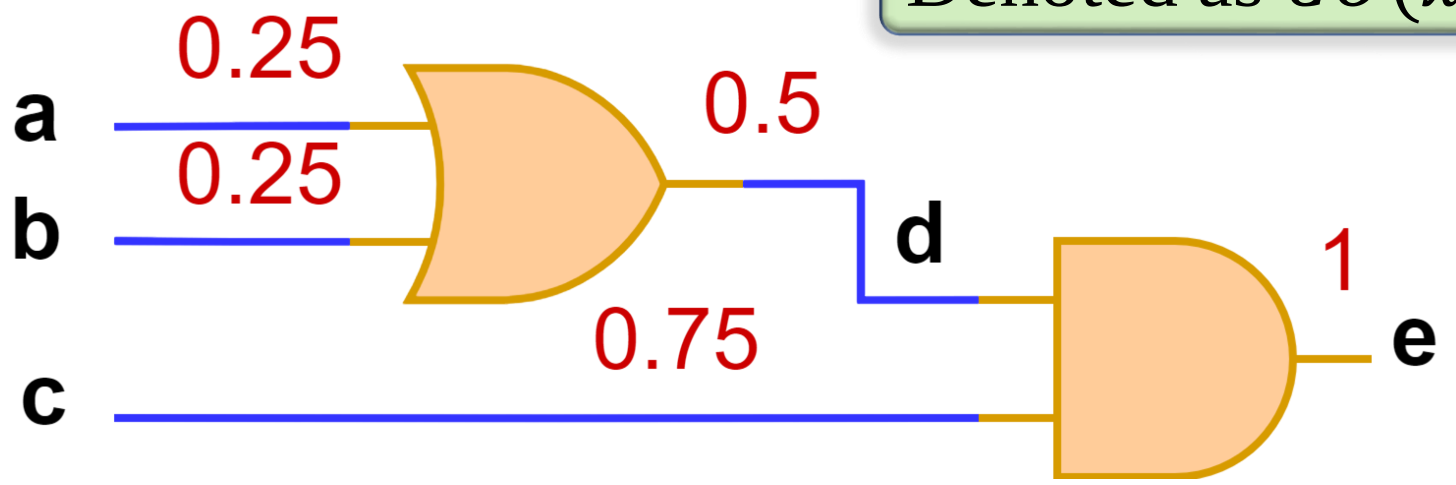


$$CC1(e) = CC1(c) * CC1(d)$$

Observability

The probability of propagating the value at a net to a primary output (PO).

Denoted as $CO(x)$ for a net x



$CO(d) = CO(e) * CC1(c)$

$CO(a) = CO(d) + CC0(b)$

Outline

Introduction and Motivation

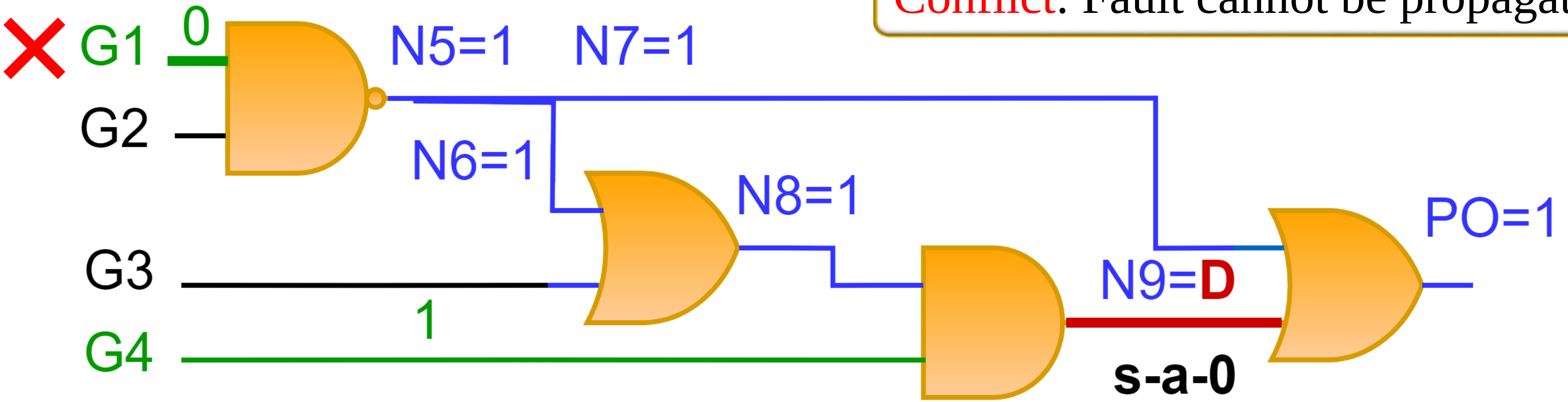
Background

Methodology

Results and Conclusion

Data Generation: Labels of a Net

Conflict: Fault cannot be propagated



If PI assignment leads to a conflict ground truth label for the nets in the backtrace path is 0 else 1

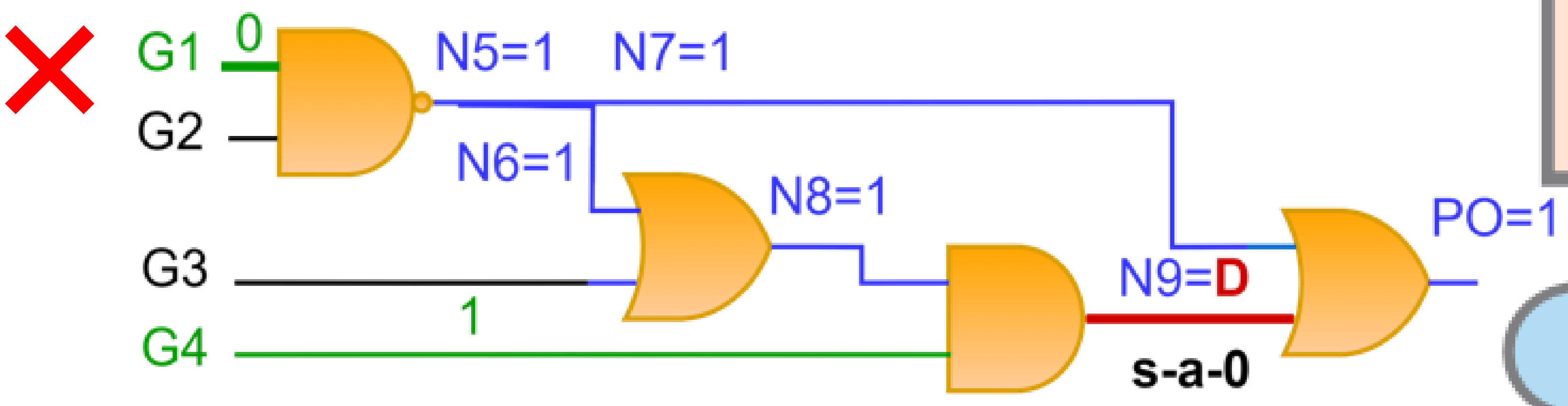
Data Generation: No-backtrack Probability

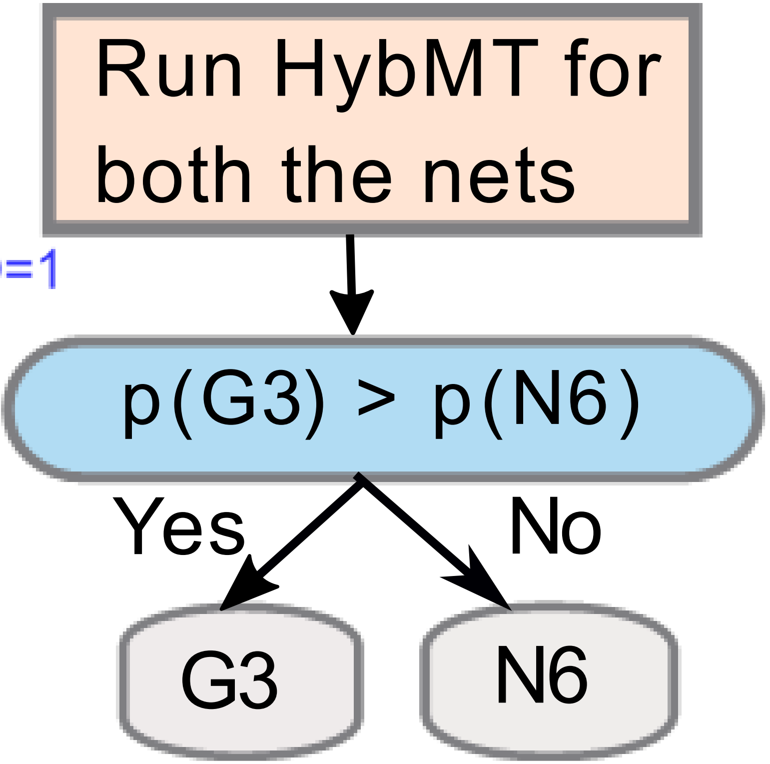
- A circuit net occurs several times while running PODEM.
- Its input features remain the same,
- But the ground truth label keeps changing.

Probability of no-backtrack

$$\frac{(\#\ \textit{ground truth label for circuit net 'l' is 1})}{(\textit{frequency of occurrence of circuit net 'l'})}$$

HybMT based PODEM



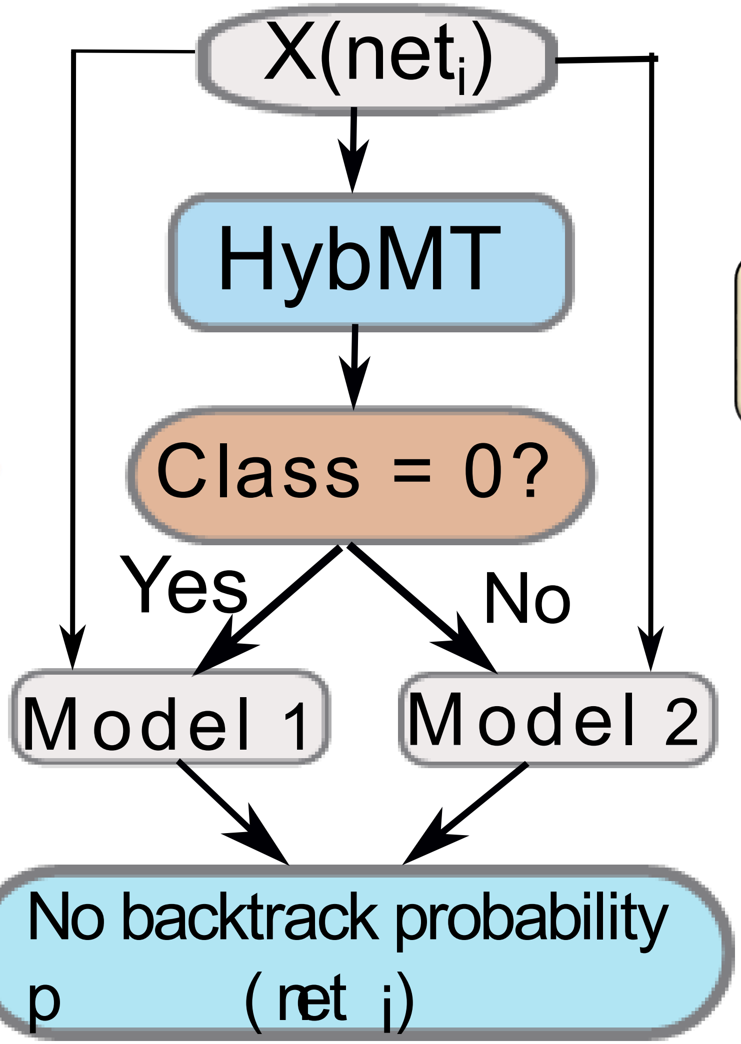


Choose the net with a higher probability of no-backtrack

Working of HybMT

$X(net_i)$ represents the features of net_i

$p(net_i)$ is the probability of no backtrack at net_i



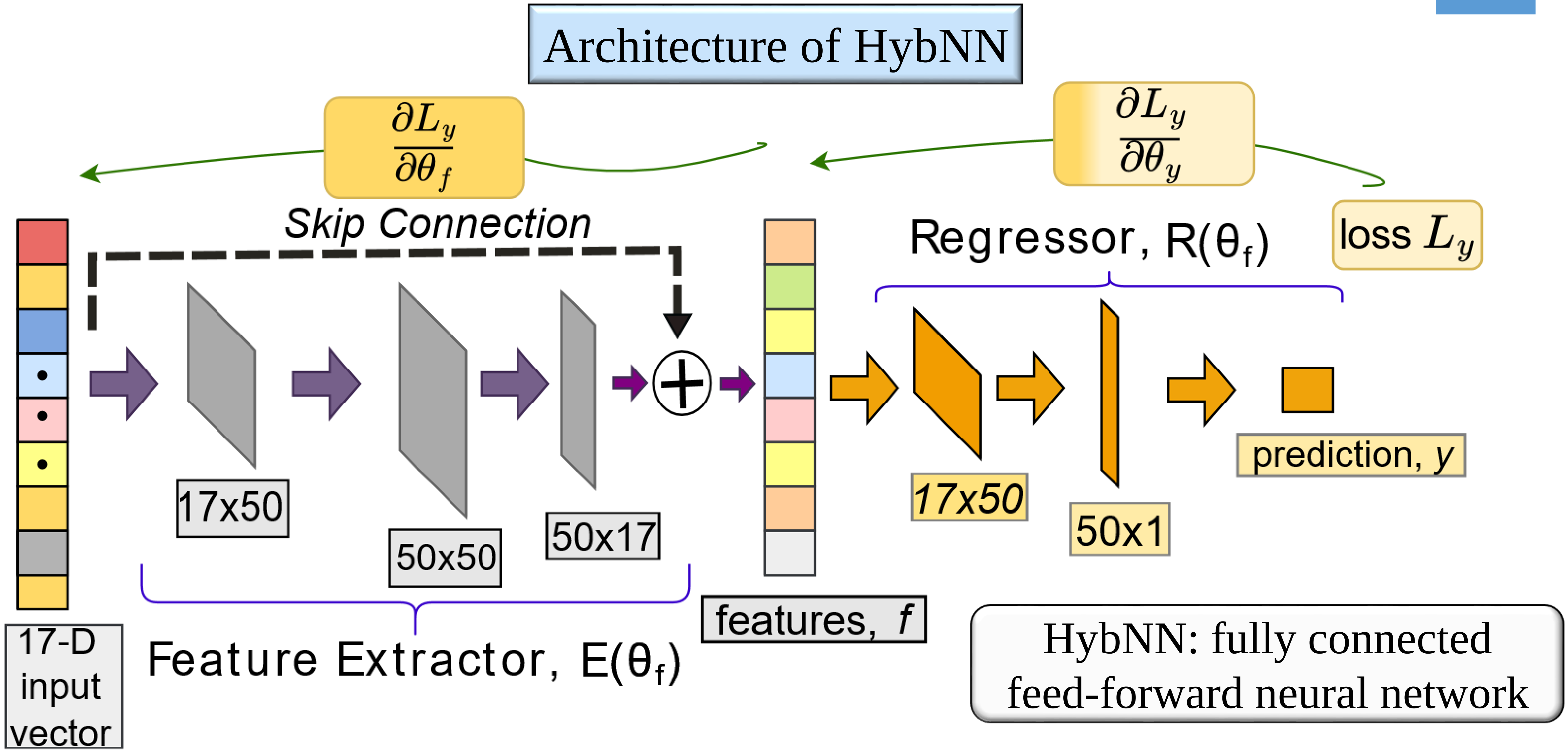
Level 1

Ensemble learning based meta predictor

Level 2

Predicts the no-backtrack probability

Working of HybMT: Lower-level Predictors



Features of the ML Models

1. SCOAP controllability*
2. SCOAP observability**
3. Fanout of the gate

HybMT

Second-level Predictor

*minimum number of PIs required to set the value of a net to a given value

**minimum number of PI assignments that need to be made to propagate the fault to a PO

1. COP controllability
2. COP observability
3. Shortest distance from the primary input (PI) (normalized)
4. Type of the gate (AND, OR etc.)

Outline

Introduction and Motivation
Background
Methodology
Results and Conclusion

Setup

Modified the open-source ATPG framework for PODEM obtained from Fault, a part of Google's OpenLane project.

Loss function: Mean squared error.

Optimizer: Adam with a learning rate of 0.01.

Weight initializer: uniformly distributed in the range (-$\sqrt{k}$, $\sqrt{k}$) where $k = \frac{1}{in\_features}$.

Results: EPFL Benchmark Circuits

+

ATPG CPU time reduction of **56.6%** over a popular commercial tool

Fault Coverage

ATPG CPU time reduction of **126.4%** over the best ML-based ATPG algorithm

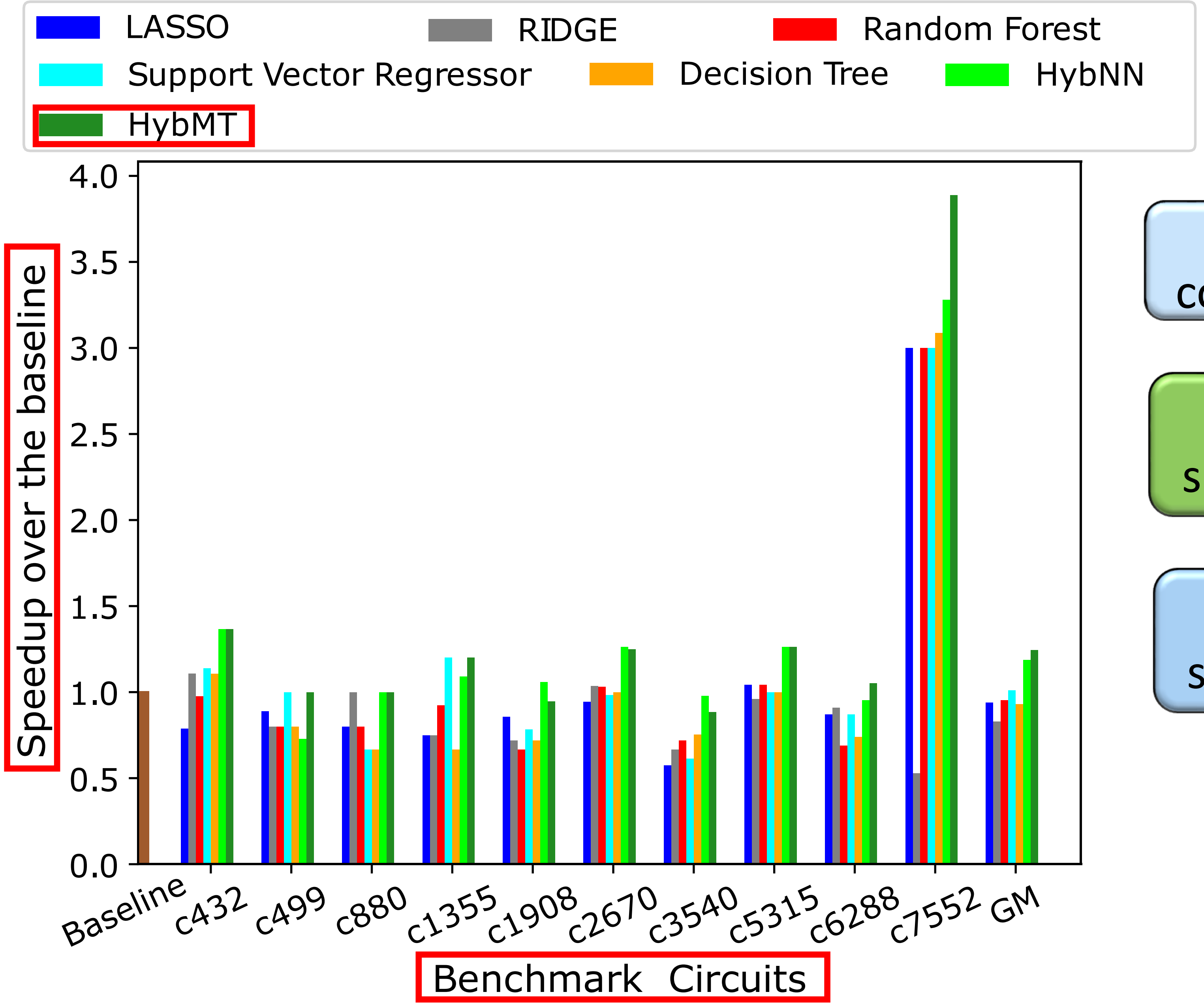
Results: CPU Time and Fault Coverage

First to beat a commercial tool !!

EPFL Benchmark Circuits	CPU Time (in ms)			Fault Coverage (in %)		
	Commercial ATPG Tool	Baseline	HybMT	Commercial ATPG Tool	Baseline	HybMT
Adder	190	2,891	180	100	100	100
Barrel-shifter	340	50	56	100	100	100
Max	950	44	780	100	100	100
Multiplier	2,510	2,545	2,200	100	100	100
Sine	790	3,25,340	708	97	74	97
Square	1,870	647	800	98	98	98

Results

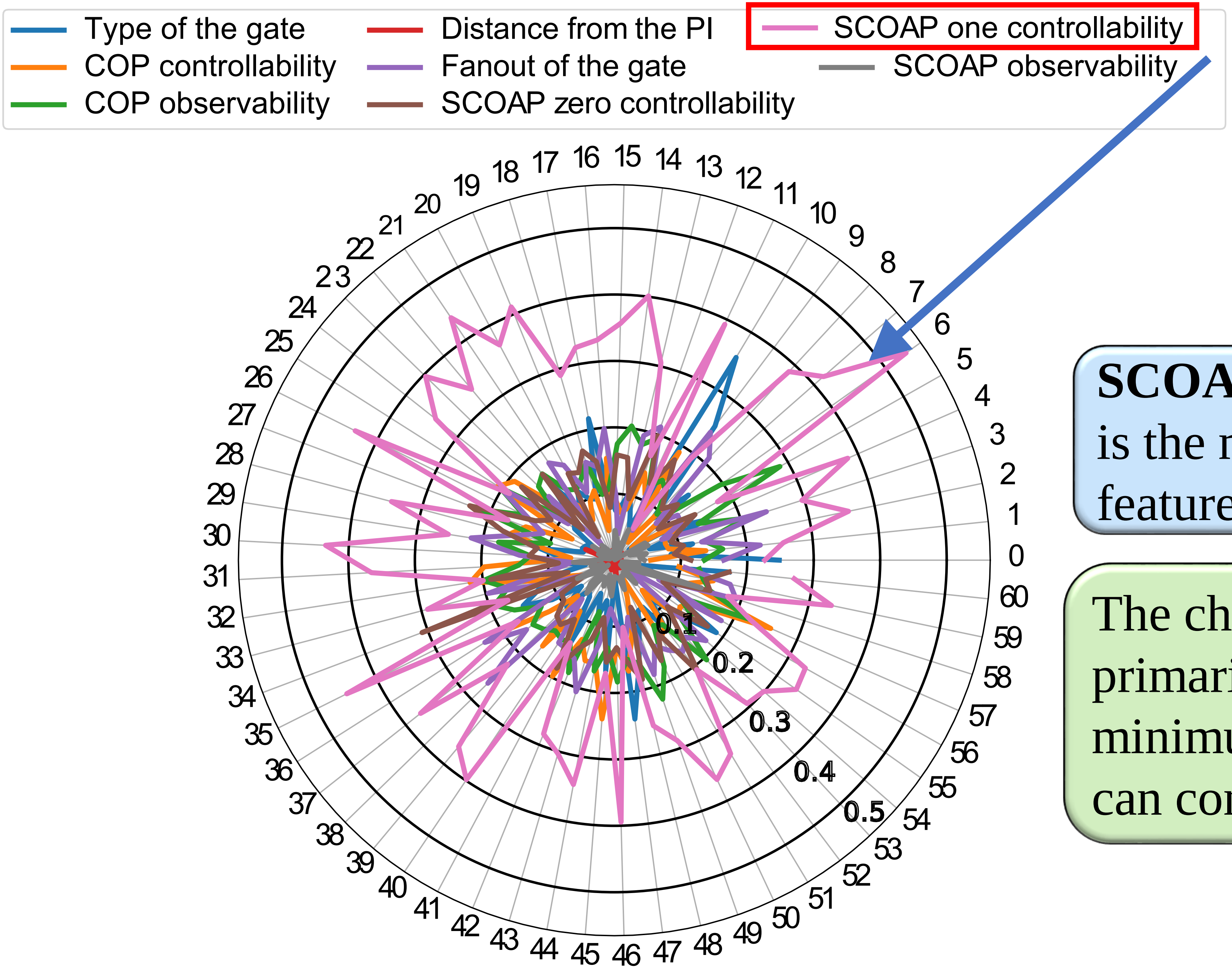
Speedup for ISCAS'85 circuits



High fault coverage (> 90%)

HybMT shows a speedup of 24.4%

HybNN shows a speedup of 18.8%



SCOAP one controlability is the most important feature of all.

The character of a net is primarily determined by the minimum number of PIs that can control it.

Conclusion

A two-level predictor ***HybMT*** for key decision making in PODEM.

Reduces PODEM's CPU time without sacrificing on **fault coverage**.

Obtains a speedup of **56.6%** and **126.4%** over the commercial tool and the state-of–the–art approach respectively.

Properly designed neural networks like **HybNN** are better candidates for guiding backtracing.

THANK YOU

Contact:
shruti.pandey@ee.iitd.ac.in

Questions

This work was funded in part by the Semiconductor Research Corporation (SRC).