DAC Young Fellows

THE CHIPS TO SYSTEMS CONFERENCE
SHAPING THE NEXT GENERATION OF ELECTRONICS

Annotating Slack Directly on Your Verilog: Fine-Grained RTL Timing Evaluation for Early Optimization

Wenji Fang, advisor: Zhiyao Xie, Hongce Zhang
Hong Kong University of Science and Technology

Background and Motivation

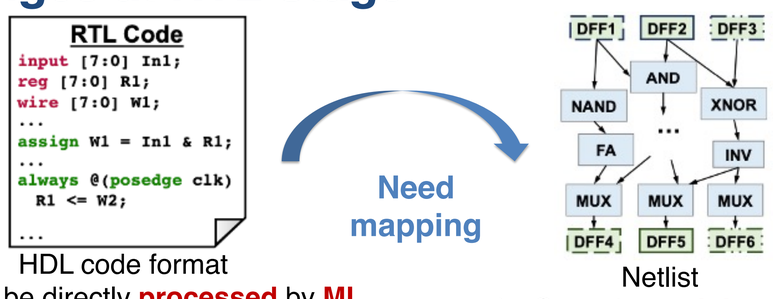
Timing Evaluation in VLSI Design Flow

- **Turnaround** optimization for timing closure
- Existing methods (STA or ML-based): **Available only post-syn**
- RTL-stage timing evaluation: **Only overall WNS/TNS, no early optimization**

Evaluate timing slack earlier at the RTL stage?

- Earlier stage with higher optimization flexibility

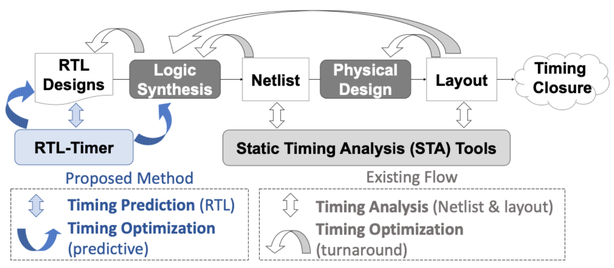
Challenges at RTL Stage



- Cannot be directly **processed** by **ML** or **STA** tools
- Cannot **annotate delay labels**

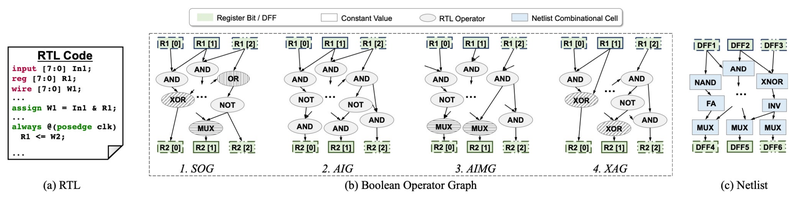
Methodology

VLSI Design Flow Equipped with RTL-Timer



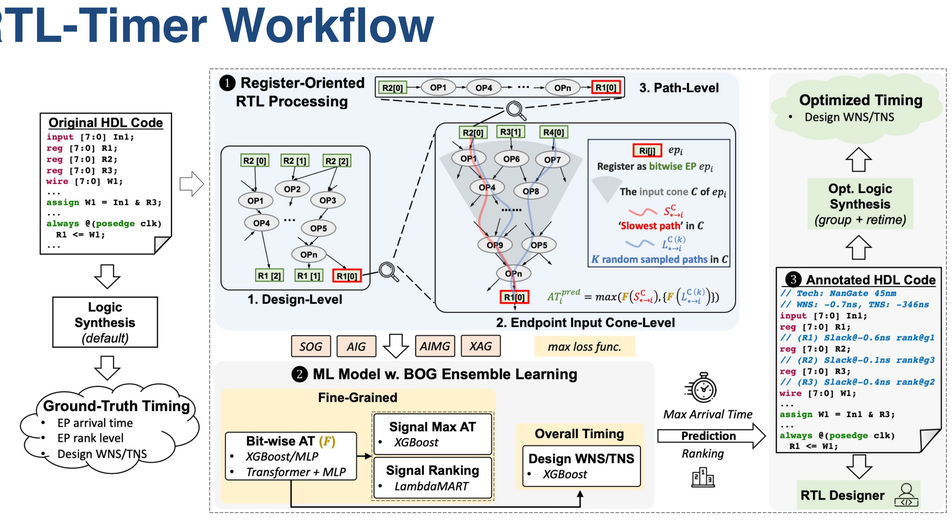
Universal ML-friendly RTL Representation

- **Bit-level**, specialized into different **variants** → **multi-view**
- One-to-one **mapping of registers**
- Treat BOG as a **pseudo netlist** – **We can directly perform STA on it!**



(a) RTL (b) Boolean Operator Graph (c) Netlist

RTL-Timer Workflow



Key Methods

- Fine-grained modeling for **register cone**
  - **STA on BOG** for slowest/ randomly sampled path → max arrival time for register
- Predictive optimization by **constraining synthesis tools**
  - **`Path grouping`** for TNS
  - **`Register retiming`** for WNS

Experimental Results

Setup and Evaluation Metrics

- 21 open-source RTL designs
- Label: post-syn slack on each endpoint register
- Evaluation metrics: R, MAPE, COVR = $\frac{1}{m}\sum_{g=1}^{m}\frac{\#(S_g \cap \hat{S_g})}{\#S_g}\times 100\%$

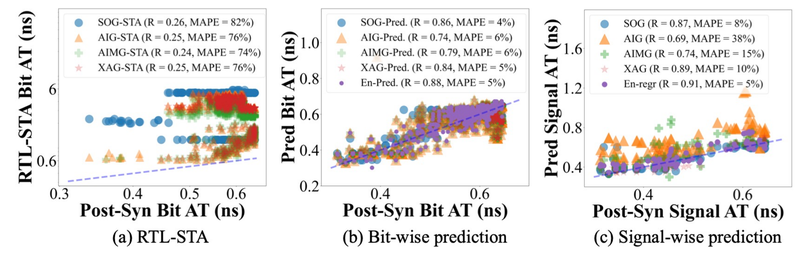
Benchmarks*	#Designs	Design Size Range		HDL Type
		#K Gates	#K Endpoints	
ITC'99	6	9 - 45	0.4 - 1.3	VHDL
OpenCores	4	6 - 56	0.2 - 3.8	Verilog
Chipyard	3	20 - 32	2.5 - 4.1	Chisel
VexRiscv	8	7 - 510	1.2 - 146	SpinalHDL

Modeling Performance

- Fine-grained **slack**
  - **R=0.89, COVR=80%**
  - Ensemble learning: robustness contributed by each variant

Fine-Grained	Method	*R*	MAPE (%)	COVR (%)
Bit-wise	Tree-based w/o sample	0.80	26	59
	MLP	0.71	35	56
	MLP w/o sample	0.65	38	54
	Transformer	0.73	35	57
	Customized GNN	0.25	53	46
	RTL-Timer	0.88	12	66
Signal-wise	Regression w/o bit-wise	0.56	28	56
	Ranking w/o bit-wise	/	/	39
	RTL-Timer (regression)	0.89	15	71
	RTL-Timer (ranking)	/	/	80

	Metrics	SOG	AIG	AIMG	XAG	Ensemble
Bit-wise	Avg. R	0.85	0.75	0.76	0.77	**0.88**
	Std. R	0.18	0.25	0.26	0.21	**0.08**
Signal-wise	Avg. R	0.82	0.81	0.84	0.8	**0.89**
	Std. R	0.15	0.22	0.1	0.1	**0.06**
	Avg. COVR	65	71	72	71	**80**
	Std. COVR	18	19	21	21	**8**



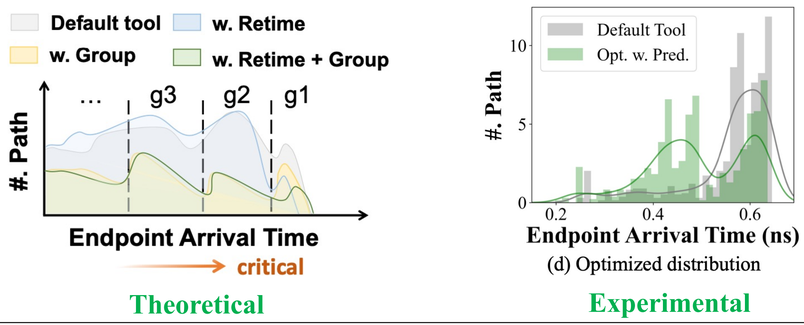
(a) RTL-STA (b) Bit-wise prediction (c) Signal-wise prediction

- Overall **WNS/TNS**
  - Calibrate fine-grained results
  - RTL-Timer outperforms all baselines

Overall	Method	*R*	R^2	MAPE (%)
WNS	SNS [17]	0.73	0.58	33
	MasterRTL [4]	0.89	0.74	15
	RTL-Timer	0.91	0.86	12
TNS	ICCAD'22 [13]	0.65	0.32	42
	MasterRTL [4]	0.96	0.94	34
	RTL-Timer	0.98	0.97	18

Optimization Performance

- Guiding commercial **logic synthesis** tool
  - Improve **WNS (3.1%), TNS (9%)**, maintain area/power
  - Reduce design cycles (concurrently run default and opt flows)
- Impact remains significant after **place (w. place opt):** improve **WNS (3.1%), TNS (6.8%)**
- Slack distribution
  - **`Path grouping`**: Single high peak → two lower peaks (better TNS)
  - **`Register retiming`**: improved WNS



(d) Optimized distribution

Conclusion

RTL-Stage Slack Evaluation for Early Optimization

- **RTL-Timer: evaluate slack on each register at the RTL stage**
  - Ensemble four ML-friendly RTL representations
  - Capture max slack with based on register cone
- **Enable early timing optimization**
  - Annotate slack on HDL code for RTL designers
  - Predictive timing optimization for logic synthesis process

Prospective Work

- Existing ML for EDA solutions are task-specific – **tedious and time-consuming**
- Prospect general circuit model solution – **easily developed**
  - **Pre-training** to learn circuit intrinsic information
  - **Fine-tuning** for specific tasks (e.g., design quality, function)

Contact: wenjifang1@ust.hk

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

About me