Multi-Stage CSM Timing Waveform Propagation Accelerated with NLDM Assistance

Shih-Kai Lee, Pei-Yu Lee, and Iris Hui-Ru Jiang

SYNOPSYS®

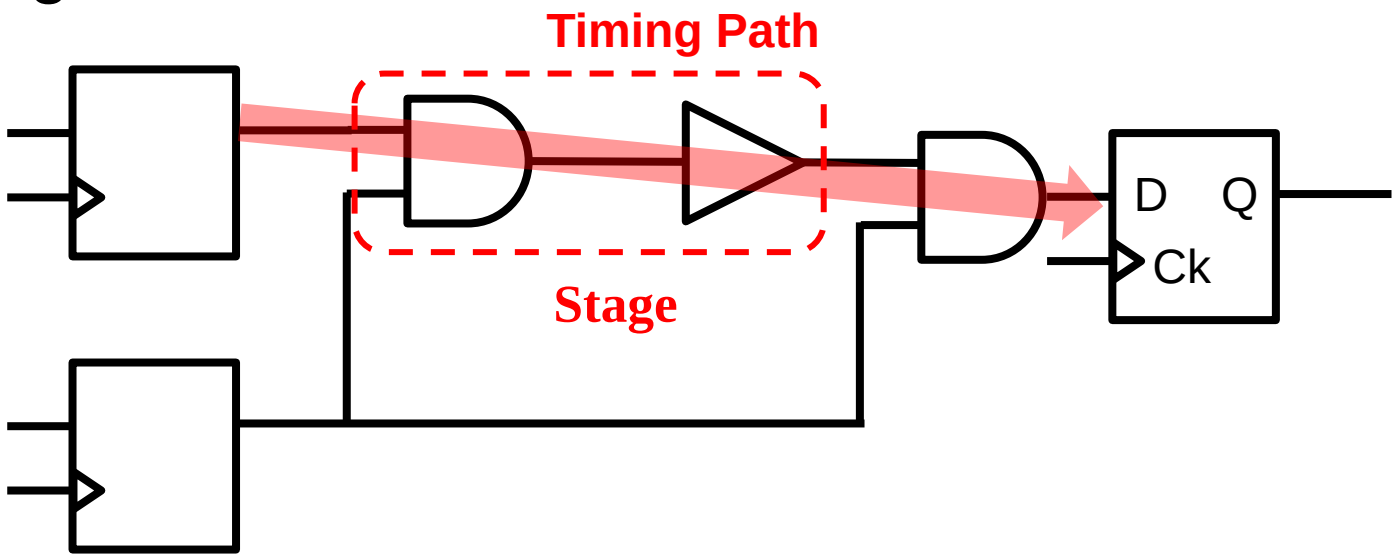
MEDIATEK
everyday genius

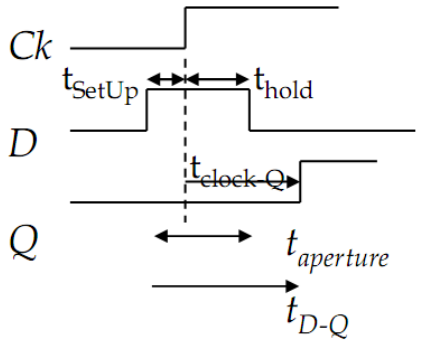
National Taiwan University
國立臺灣大學

Static Timing Analysis (STA)

- Essential at every stage to achieve timing closure in modern IC design
  - **Gate-Level** STA is widely adopted for timing ECO and signoff
  - Find **critical paths** violating **timing constraints (setup/hold)**

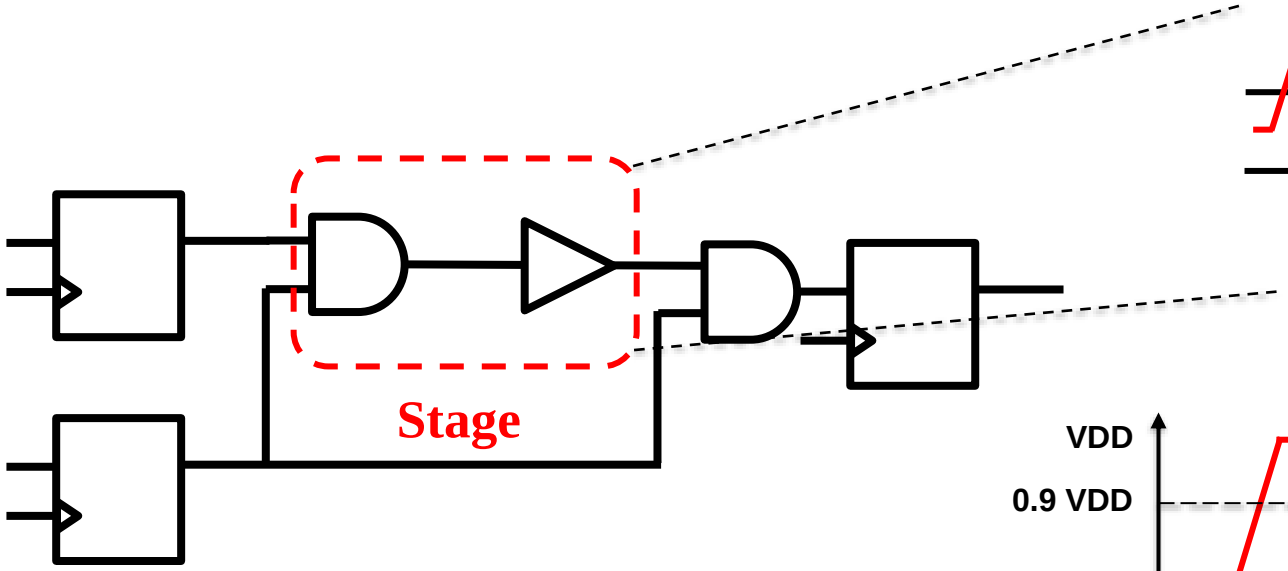
- **Gate-Level** STA propagates **delay** and **slew** along the **timing paths** with **Stage Timing Calculator**

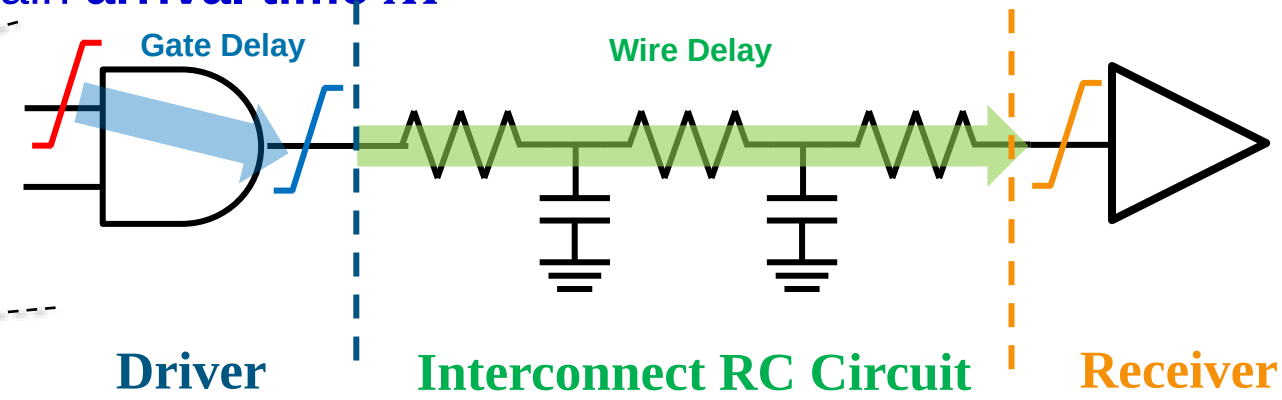


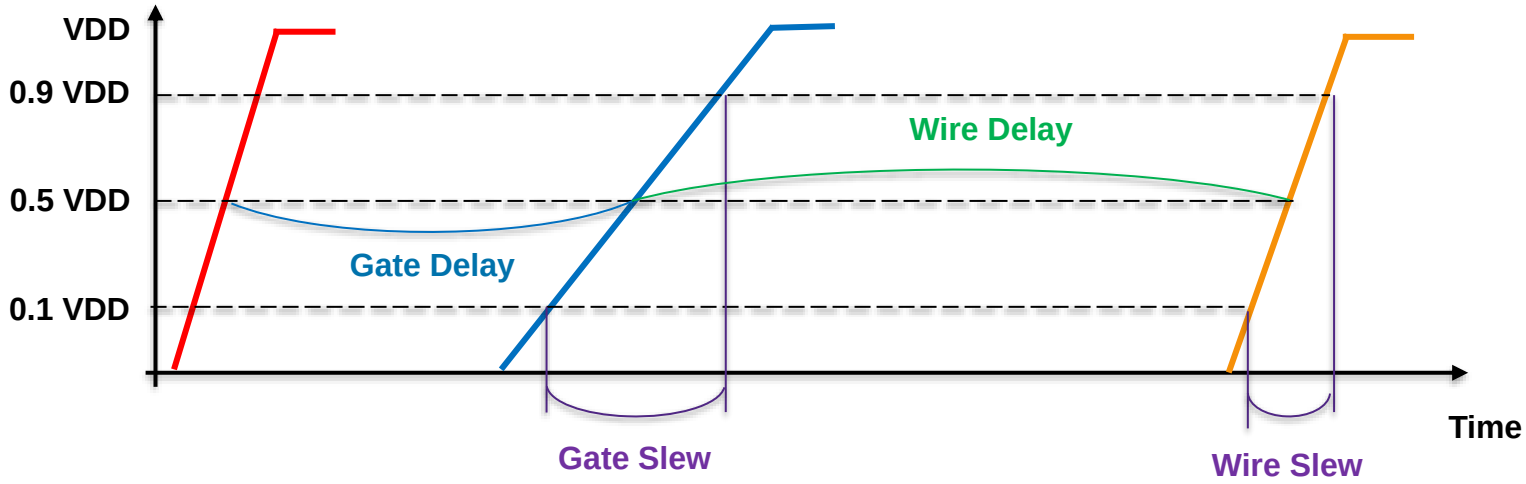


Stage Timing Calculator

- **Stage = a driver gate + a set of receivers + an interconnect RC circuit**
- Compute **gate delay/slew** and **wire delay/slew**
  - Sum up delay along the timing path to obtain **arrival time *AT***

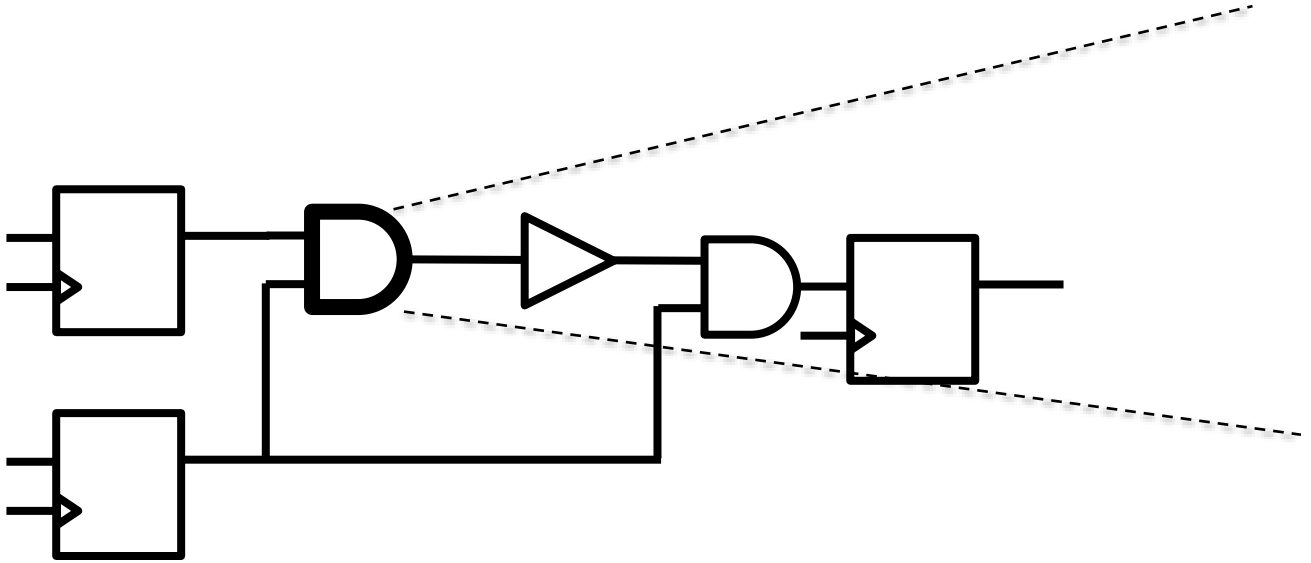


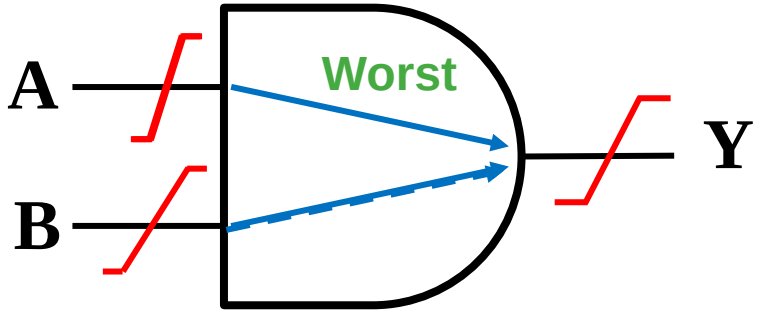




Graph-Based STA

- **Graph-based** STA provides **conservative** but **more efficient** analysis
  - Propagate the **worst** timing only





Timing Calculation in STA

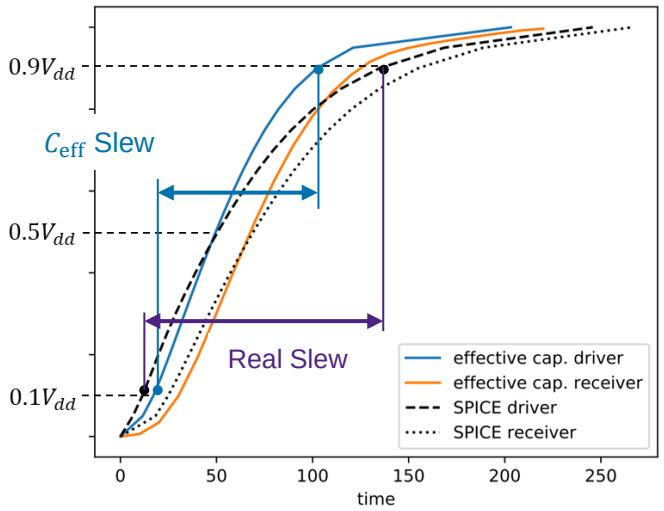
- Non-Linear Delay Model (NLDM)
  - Lookup table for **delay** and **slew**
  - Fixed receiver capacitance
    - **Miller effect**
  - **Effective capacitance**
    - **Long tail effect**
- Current Source Timing Model (CSM)
  - Cadence **ECSM** and Synopsys **CCS**
  - Tabulate the **output current** of the driver cell
  - Variable capacitor with tabulated capacitance

Delay Table

C_l \ T_r		
	0.5	0.8
	0.7	0.9

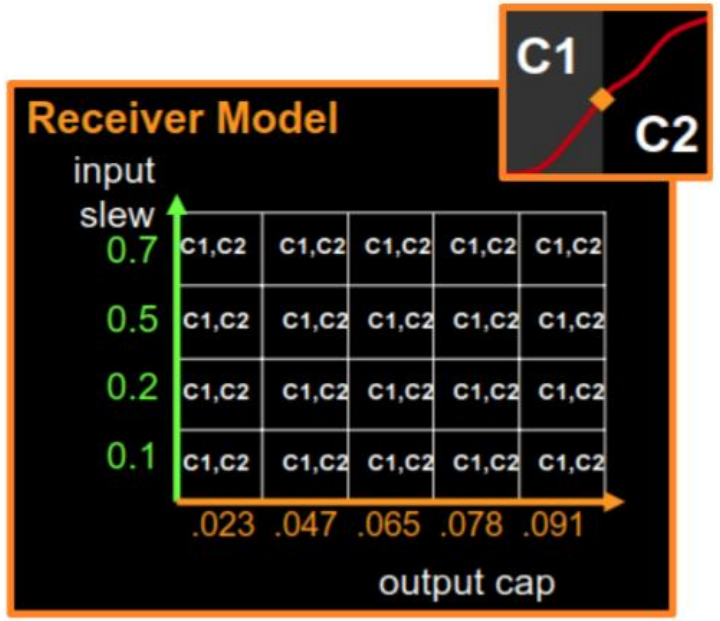
Slew Table

C_l \ T_r		
	0.5	0.8
	0.7	0.9

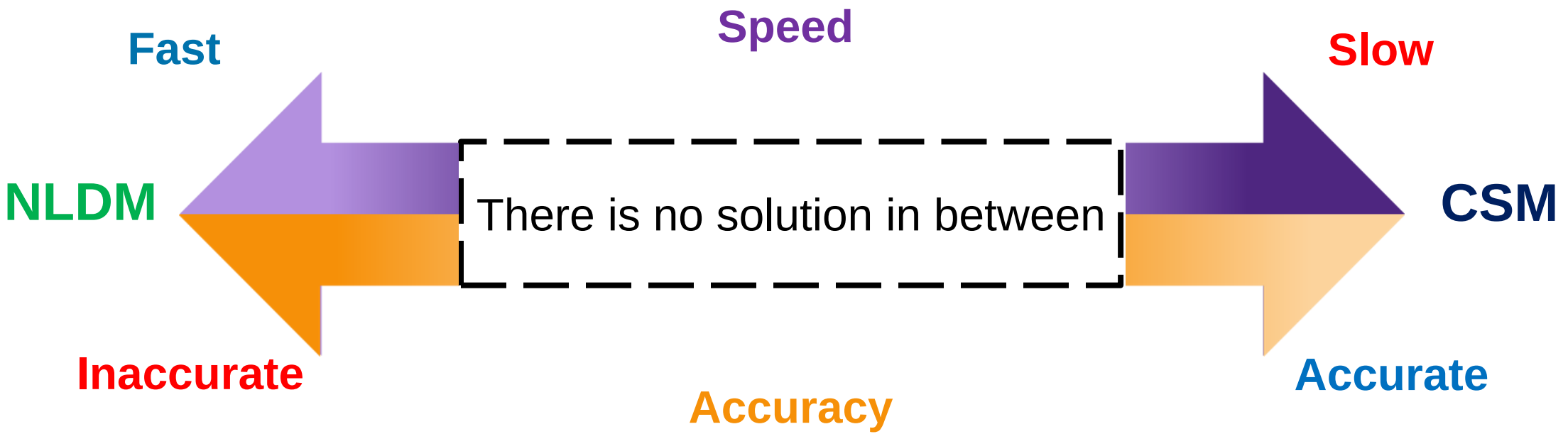


Current Table

C_l \ T_r		

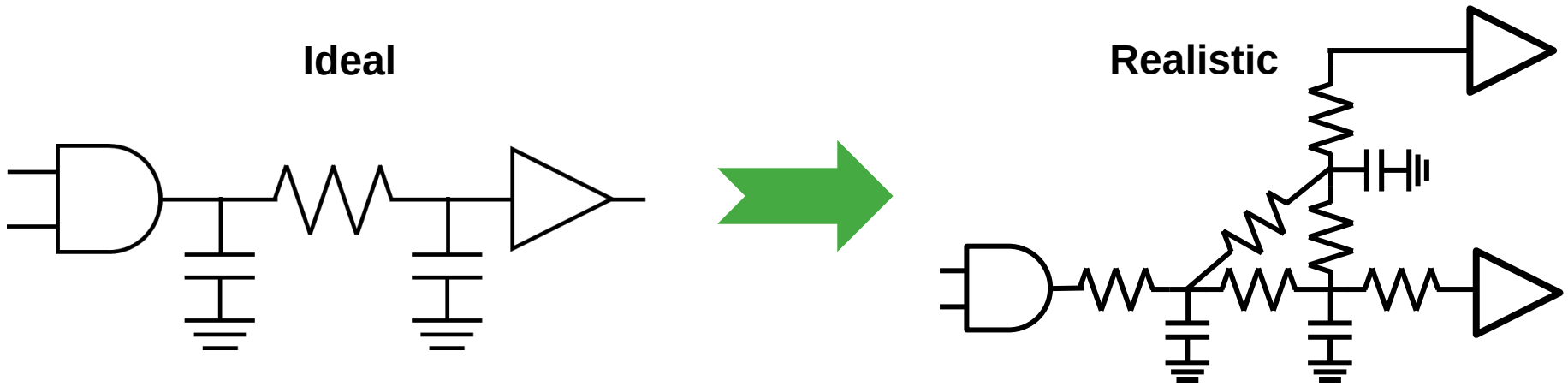


Motivations



Related Works

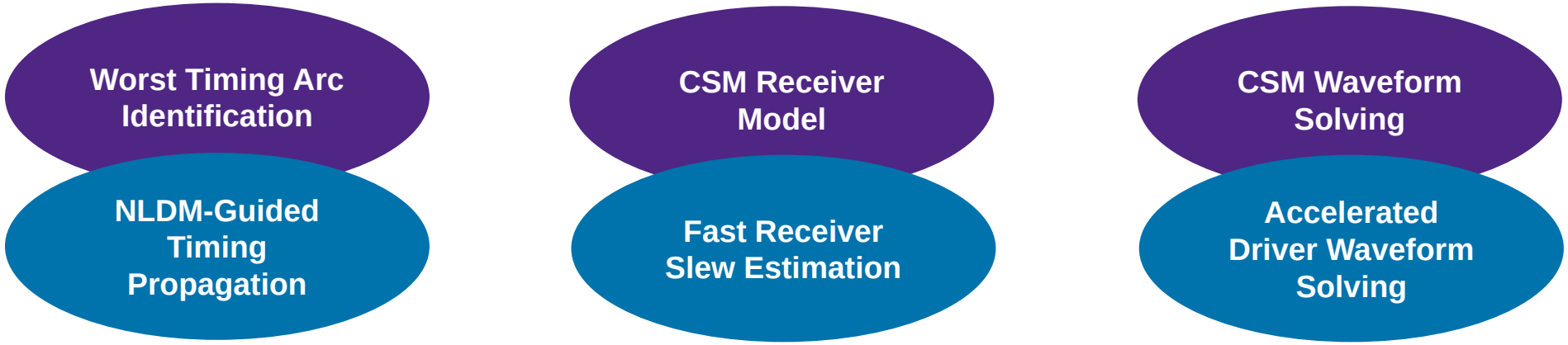
- DAC ’08 (MVTM) [1]
  - Only considered driver models
  - No clear method provided to deal with general circuits
- TVLSI ’21 [2]
  - Considered both advanced driver and receiver models
  - Limited to **π-shaped circuits** and single receiver
- In realistic, STA need to handle **arbitrary RC circuits** and **multiple receivers**



[1] P. Feldmann et. al., “Driver waveform computation for timing analysis with multiple voltage threshold driver models,” 45th DAC, 2008.
[2] Dimitrios et. al., “Gate delay estimation with library compatible current source models and effective capacitance,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2021.

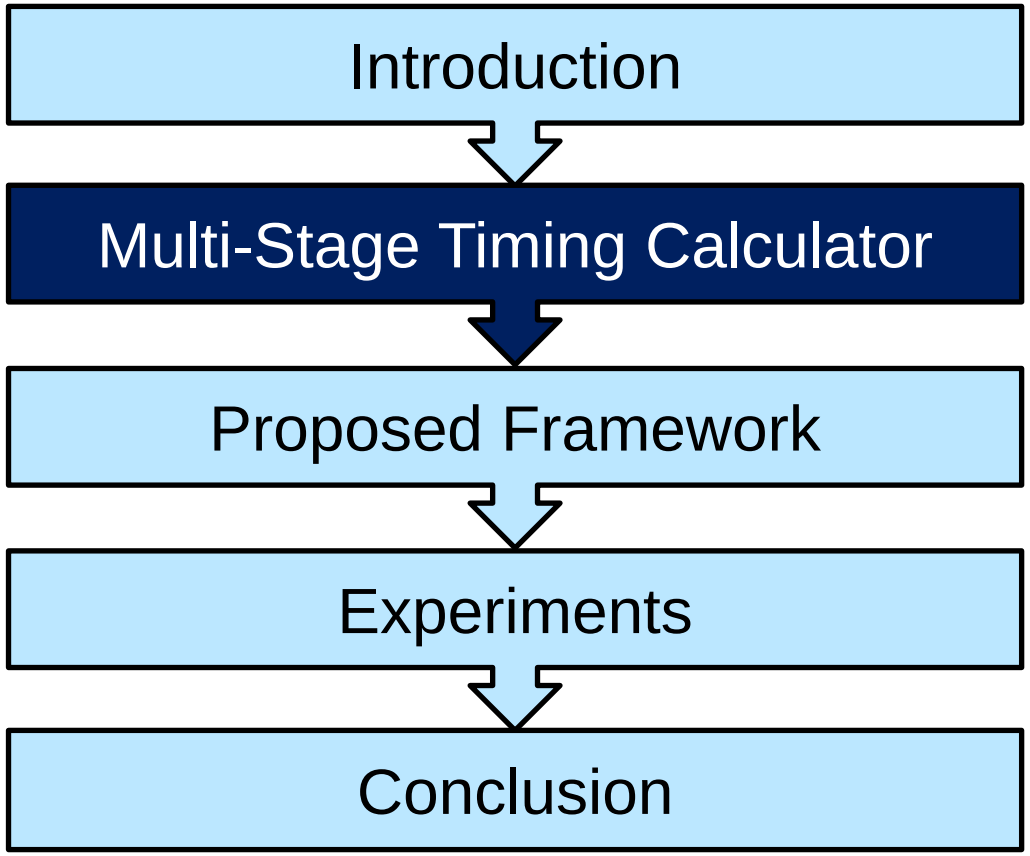
Contributions

- We demonstrate a CSM-compliant graph-based STA framework, capable of propagating **complete waveforms** along **multiple stages** with **generalized RC interconnects**
- We explore the potential of using NLDM as a guide to accelerate CSM-based STA, **compatible with any CSM-based stage timing calculator**

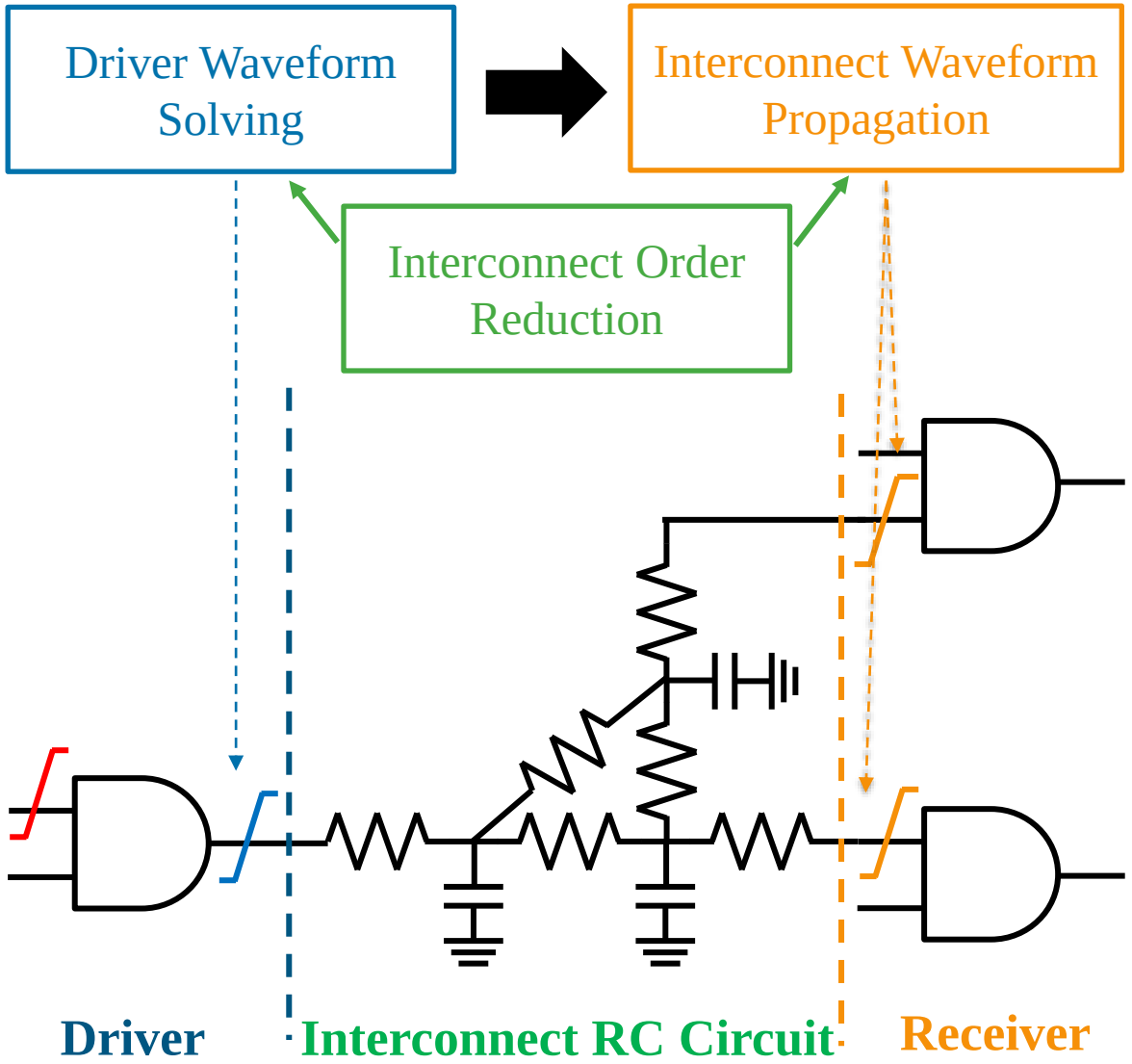


- Experimental results show that the proposed techniques significantly improve the performance with minimal accuracy loss

Outline


Introduction
Multi-Stage Timing Calculator
Proposed Framework
Experiments
Conclusion

Single Stage Timing Calculation

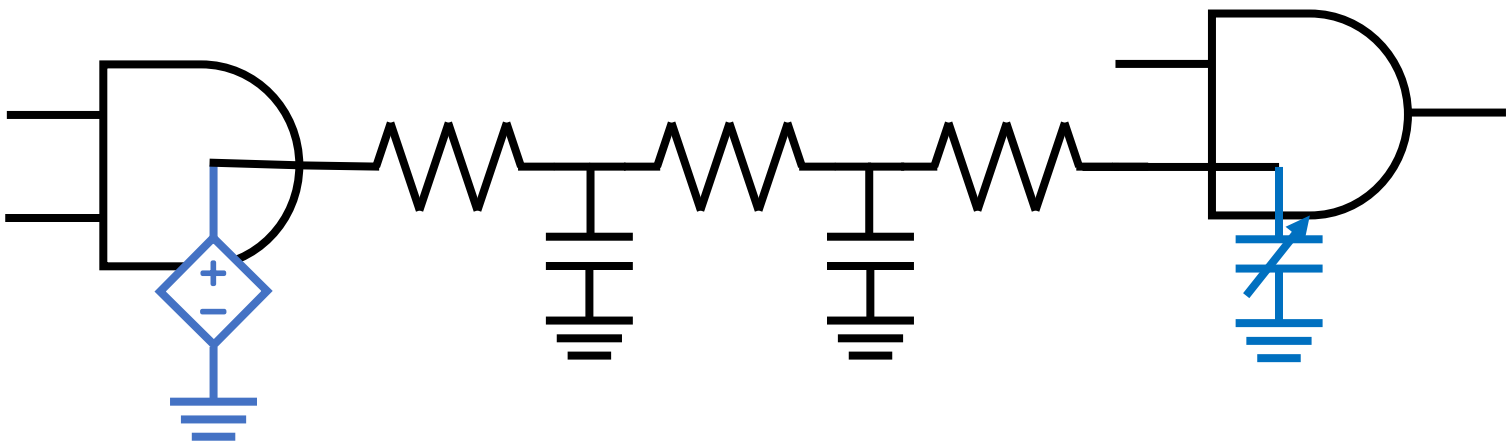


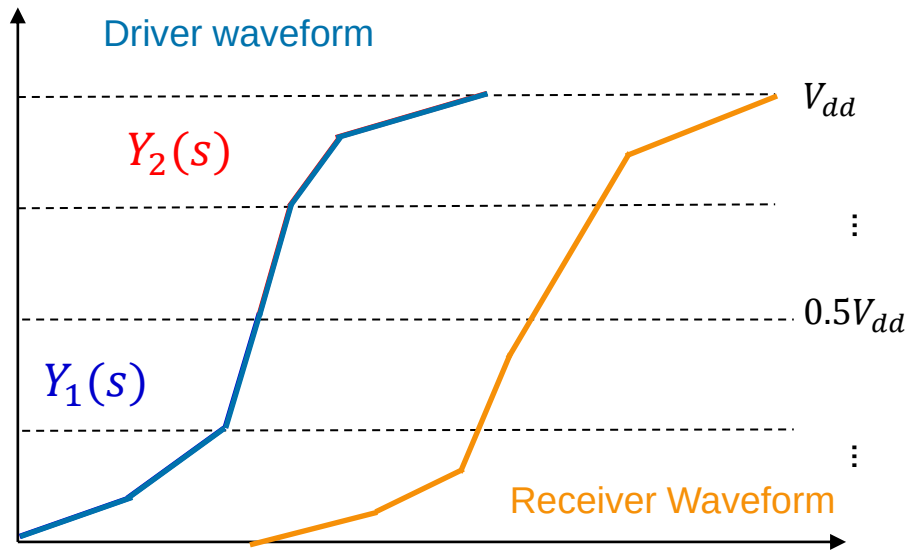
Waveform Propagation

- Multiple Voltage Threshold Driver Model (MVTM) [1]
  - **Divide output voltage waveform** into segments with predefined voltage thresholds
  - Derive analytical solution to voltage waveform with **first-order poles**
- Pole Analysis via Congruence Transformations (PACT) [3]
  - Capable of **arbitrary RC circuits**
- Modified Nodal Analysis (MNA) circuit simulation

$$Y(s) = \sum_{i=1}^{n} \frac{sz_i}{s - p_i}$$

$V(t)$

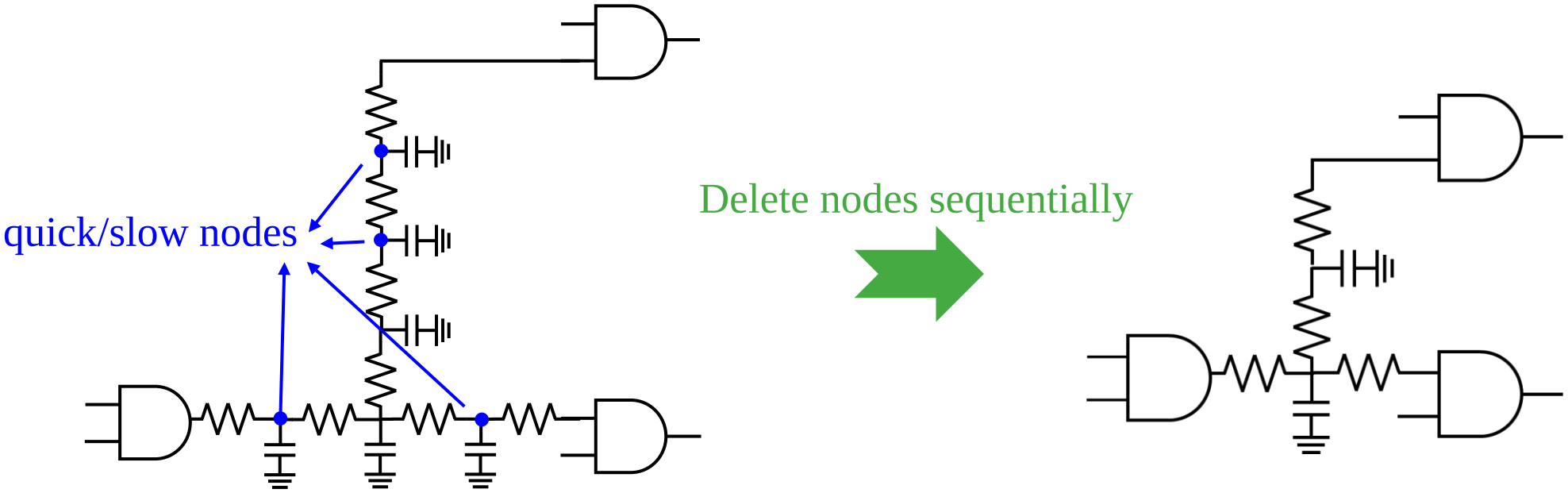




[3] K. J. Kerns and A. T. Yang, "Stable and efficient reduction of large, multiport RC networks by pole analysis via congruence transformations," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, no. 7, pp. 734-744, July 1997.

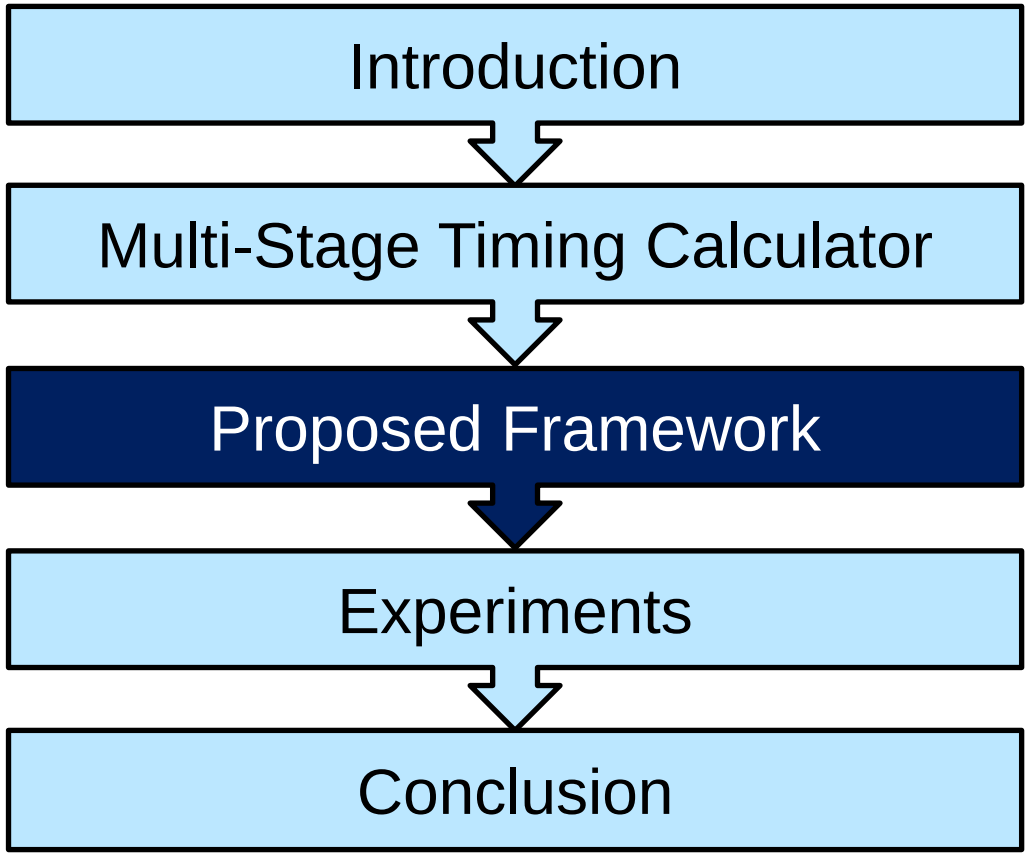
Interconnect Order Reduction

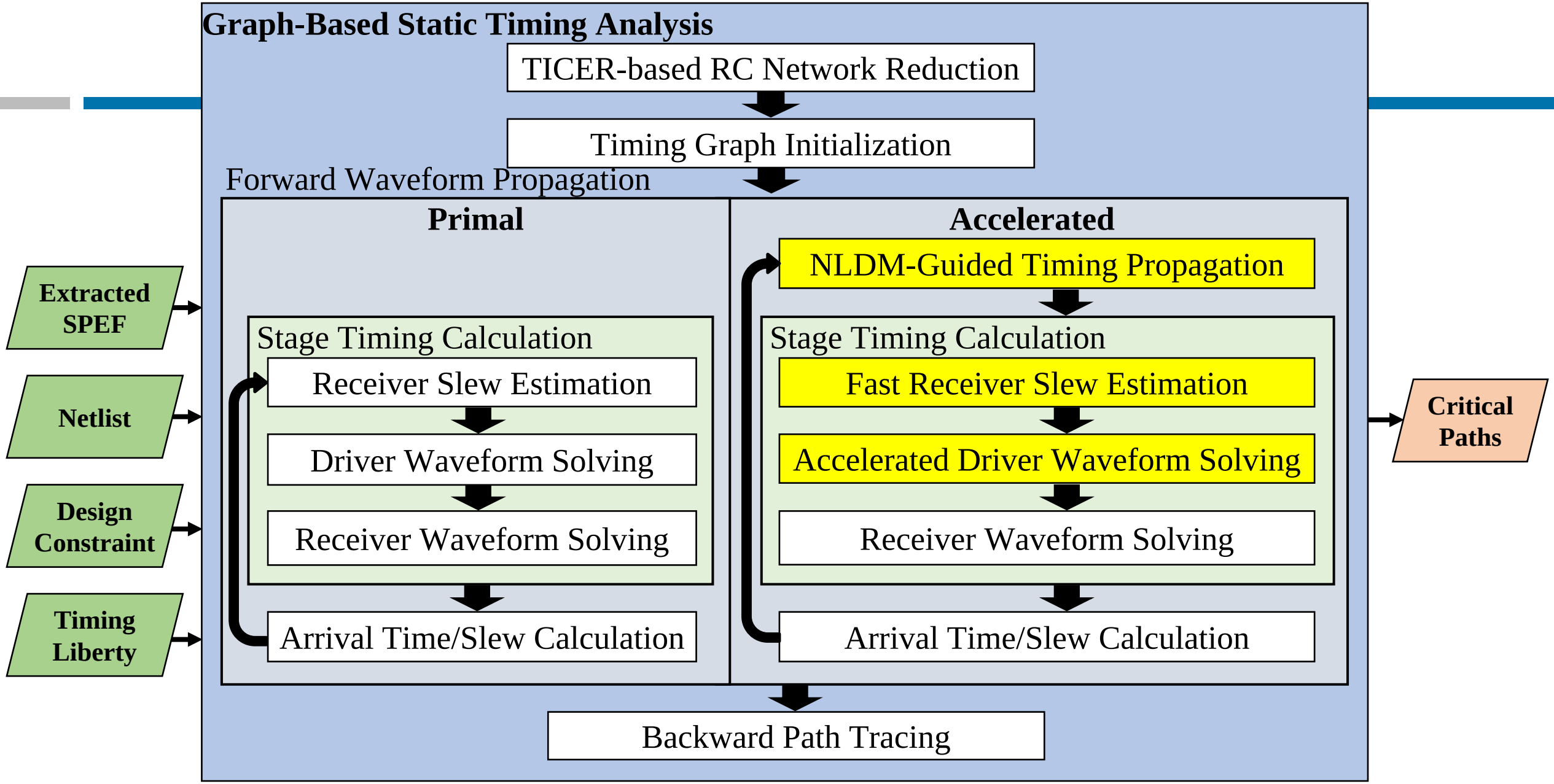
- Time Constant Equilibration Reduction (TICER) [4]
  - Delete nodes with extreme time constants
    - Consider **quick** nodes short, **slow** nodes as open
  - Graph-based implementation for linear time complexity



[4] B. N. Sheehan, “TICER: Realizable reduction of extracted RC circuits,” 1999 IEEE/ACM International Conference on Computer-Aided Design.

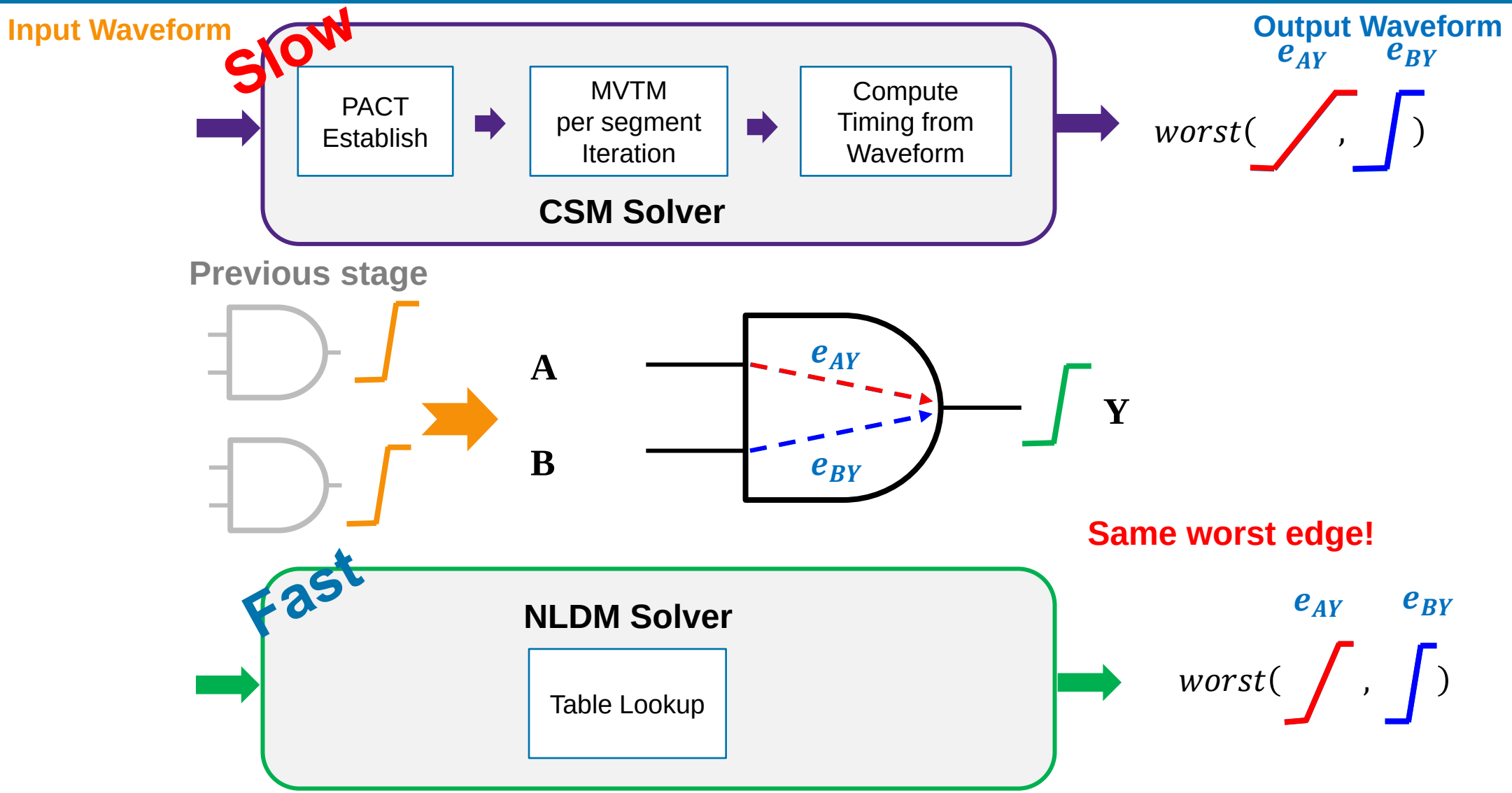
Outline


Introduction
Multi-Stage Timing Calculator
Proposed Framework
Experiments
Conclusion


Graph-Based Static Timing Analysis
TICER-based RC Network Reduction
Timing Graph Initialization
Forward Waveform Propagation
Primal
Stage Timing Calculation
Receiver Slew Estimation
Driver Waveform Solving
Receiver Waveform Solving
Arrival Time/Slew Calculation
Accelerated
NLDM-Guided Timing Propagation
Stage Timing Calculation
Fast Receiver Slew Estimation
Accelerated Driver Waveform Solving
Receiver Waveform Solving
Arrival Time/Slew Calculation
Backward Path Tracing
Extracted SPEF
Netlist
Design Constraint
Timing Liberty
Critical Paths

Acceleration #1:
NLDM-Guided Timing Propagation

Worst Timing Arc with NLDM



NLDM-Guided Timing Propagation

- Identify the worst timing arc by NLDM

Algorithm 1 NLDM-Guided Timing Propagation

Input: Timing Graph G and driver output pin O
Output: Worst arrival time at and slew on output pin O

1: Calculate the delay and slew of each timing arc in $O.Fanin$ using NLDM
2: Identify the worst timing arc $e_{\text{slew}}^{\text{worst}}$ and $e_{\text{at}}^{\text{worst}}$
3: Calculate driver waveform on edge $e_{\text{slew}}^{\text{worst}}$ and $e_{\text{at}}^{\text{worst}}$ using CSM
4: Update arrival time by $e_{\text{at}}^{\text{worst}}$ and slew by $e_{\text{slew}}^{\text{worst}}$ on output pin O

Propagate CSM waveform only

- Matching ratio of identified worst timing arcs by NLDM and CSM

Circuit		usb_funct	vga_lcd	leon3mp	leon2
Arrival Time	Matching Ratio	99.79%	99.97%	99.80%	99.93%
	Unmatched Error (ps)	0.17	0.15	0.24	0.21
Slew	Matching Ratio	94.71%	97.41%	92.87%	91.75%
	Unmatched Error (ps)	0.47	0.66	0.70	0.96

Almost every gate selects the correct arc

Performance and Path Coverage

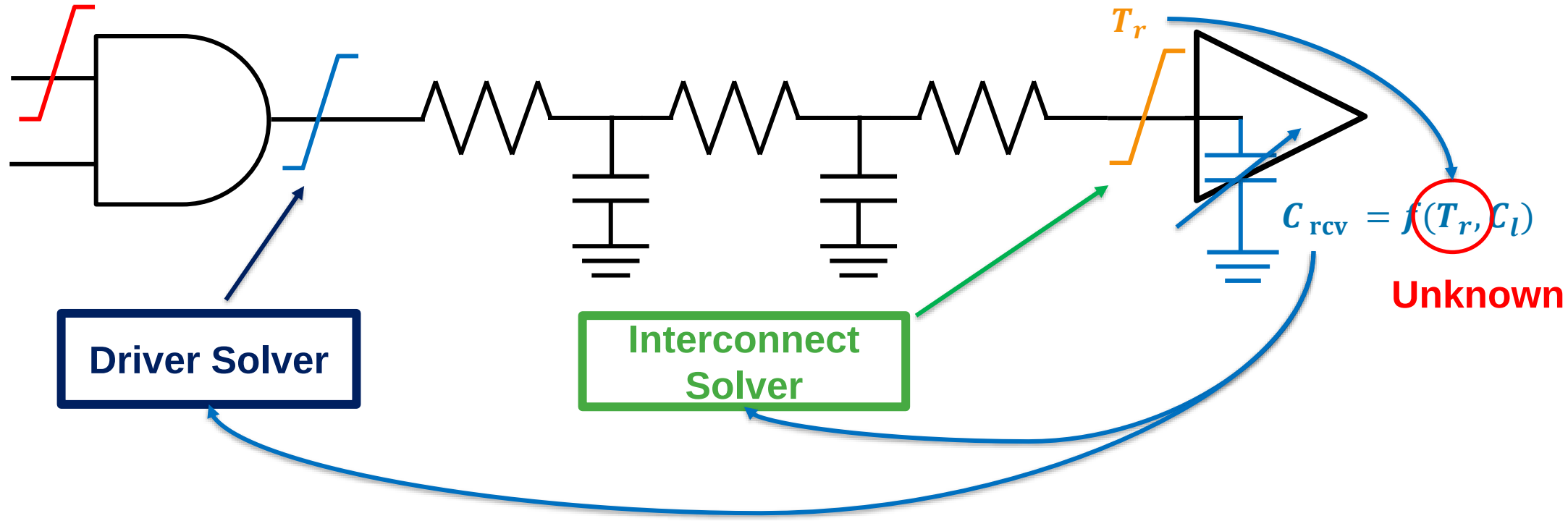
- Performance and path coverage ratio against Primal flow with and without NLDM-guided timing propagation based on top 500 critical paths
- **Better path coverage** compared to NLDM flow and a **1.66X runtime speedup** compared to Primal flow

Circuit		usb_funct	vga_lcd	leon3mp	leon2
CSM (Primal)	Matching Ratio	100.00%	100.00%	100.00%	100.00%
	Runtime (s)	28.37	249.39	2235.53	3535.61
	Normalize.	1.55	1.59	1.66	1.91
NLDM	Matching Ratio	93.40%	97.30%	56.80%	99.20%
	Runtime (s)	1.26	11.71	114.50	151.95
	Normalize.	0.07	0.07	0.08	0.08
Proposed	Matching Ratio	100%	100.00%	94.00%	100.00%
	Runtime (s)	18.271	156.512	1349.4	1853.95
	Normalize.	1.00	1.00	1.00	1.00

Acceleration #2:
Fast Receiver Slew Estimation

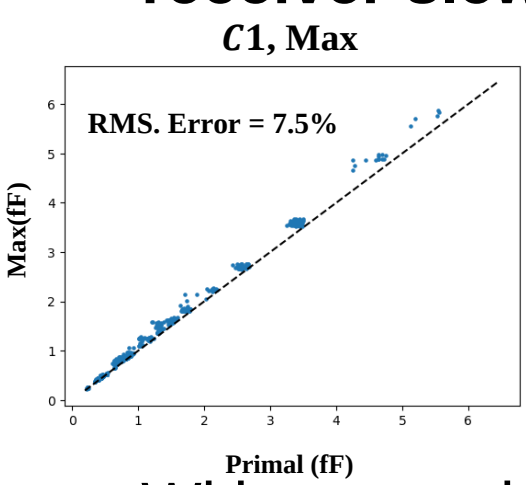
CSM Receiver Model

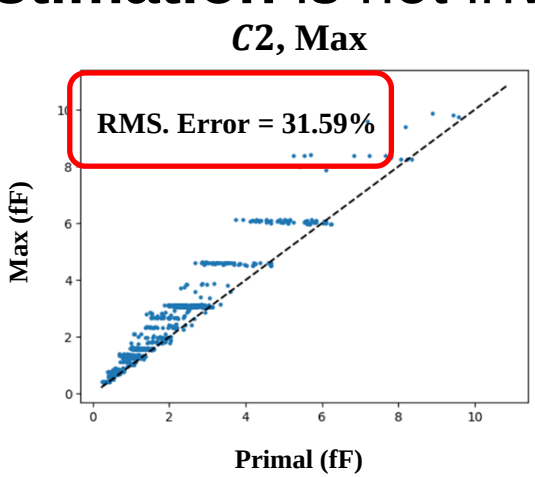
- Input slew is required for CSM receiver model table lookup, but **receiver slew is unknown** before propagation
  - The propagation is repeated until convergence

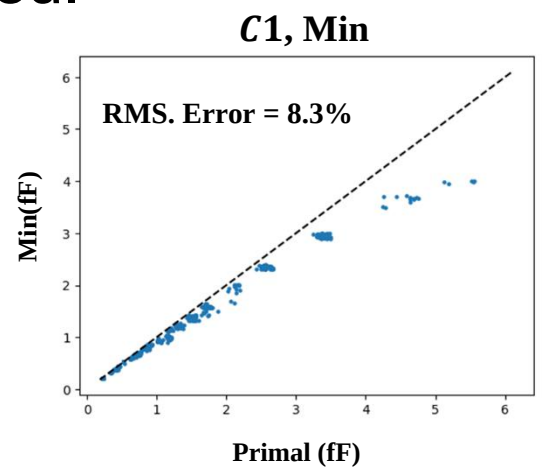


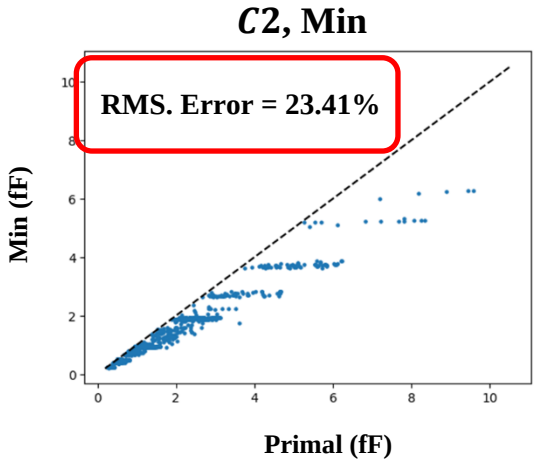
Necessity of Receiver Slew

- **Significant error** in **receiver capacitance** occurs, especially for C_2, when **receiver slew estimation** is not invoked.

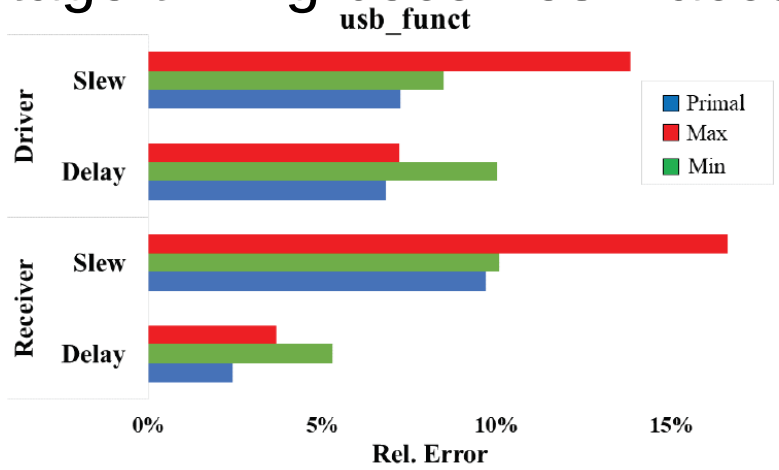




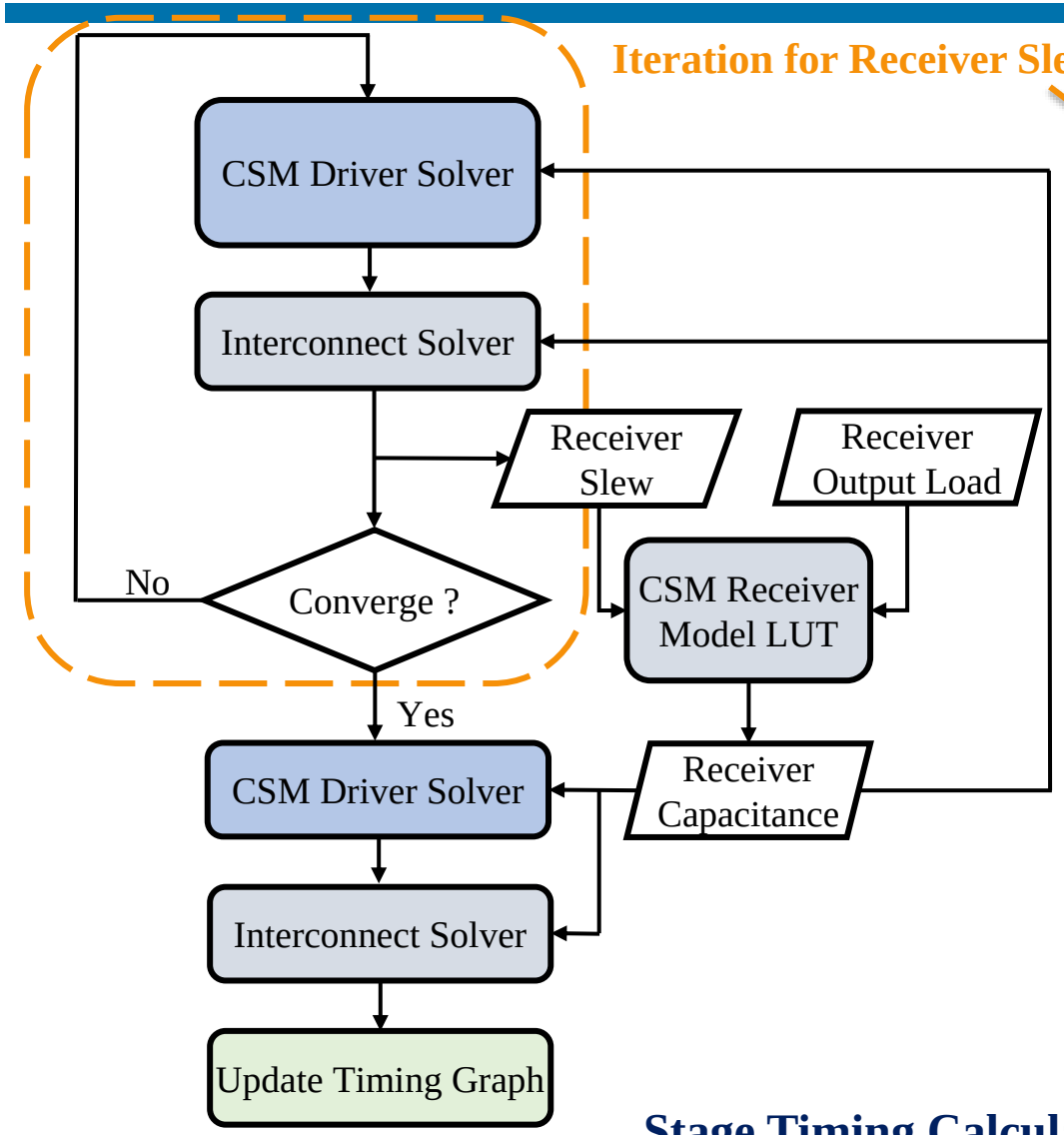


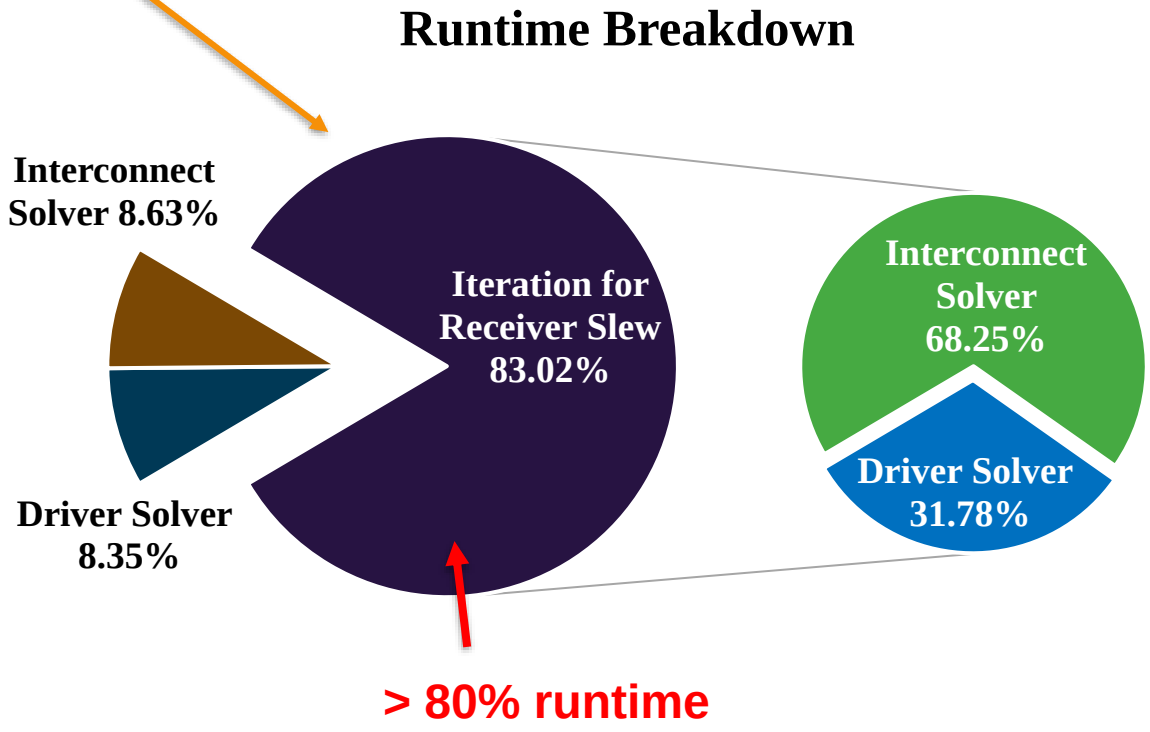


- Without receiver slew estimation, the stage timing becomes inaccurate and unstable, for both delay and slew.



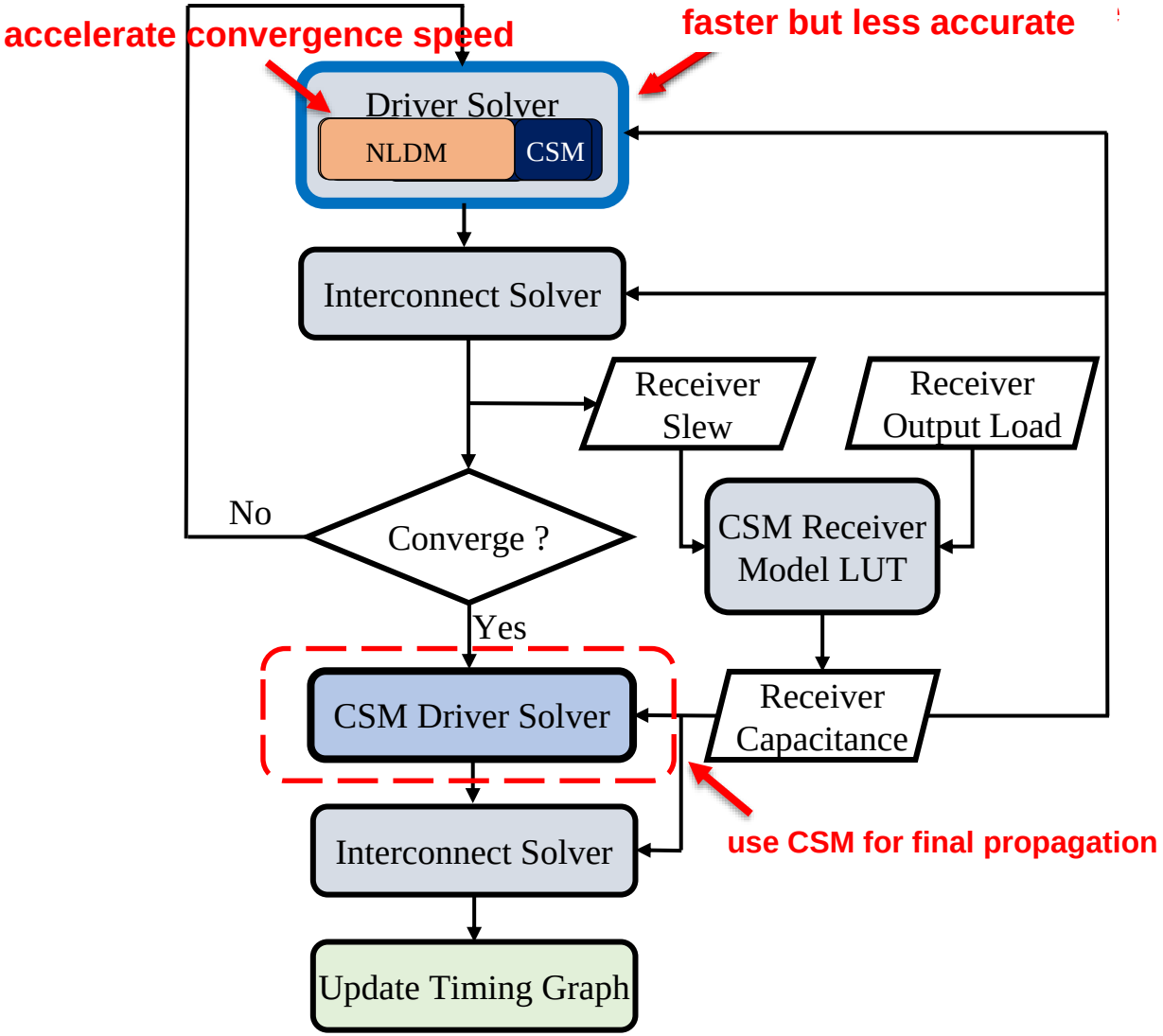
Flow Chart for Receiver Slew Estimation





Stage Timing Calculator

Fast Receiver Slew Estimation



- Accelerate converge speed of receiver slew with **NLDM assistance**
  - **Tradeoff** between accuracy and runtime control **by the percentage of NLDM usage**

Receiver Slew Estimation

- **100% NLDM usage** achieves the best tradeoff

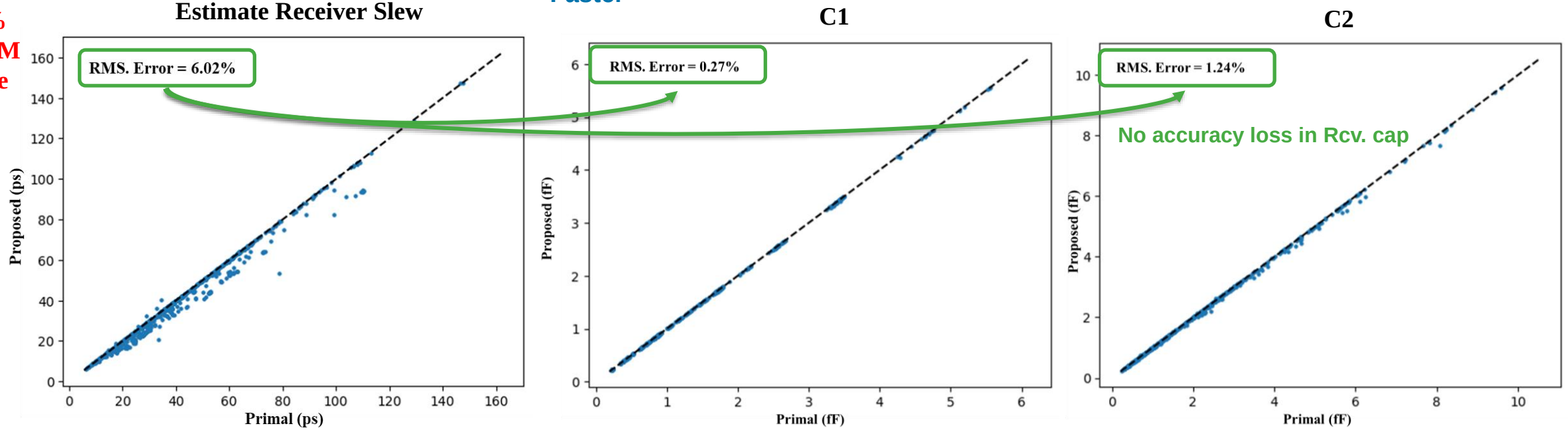
		usb_funct			vga_lcd			leon3mp			leon2		
Percentage of NLDM usage		0%	50%	100%	0%	50%	100%	0%	50%	100%	0%	50%	100%
RMS. Rel. Error	Driver Delay	6.82%	6.82%	6.74%	3.06%	3.07%	3.03%	3.03%	3.02%	2.98%	5.01%	5.01%	4.98%
	Driver Slew	7.25%	7.24%	7.52%	7.35%	7.34%	7.60%	14.38%	14.32%	14.55%	7.94%	7.65%	8.08%
	Receiver Delay	2.40%	2.40%	2.39%	5.44%	5.46%	5.31%	4.92%	4.93%	4.81%	4.95%	4.94%	4.89%
	Receiver Slew	9.70%	9.69%	9.97%	8.23%	8.21%	8.52%	12.65%	12.58%	12.99%	8.24%	8.17%	8.56%
Runtime		28.37	21.49	19.54	249.39	200.4	173.74	2235.53	1670.58	1500.41	2235.53	1670.58	1500.41
Norm.		1.45	1.10	1.00	1.44	1.15	1.00	1.49	1.11	1.00	1.49	1.11	1.00

less than 0.5% error

Faster

100% NLDM Usage

Estimate Receiver Slew

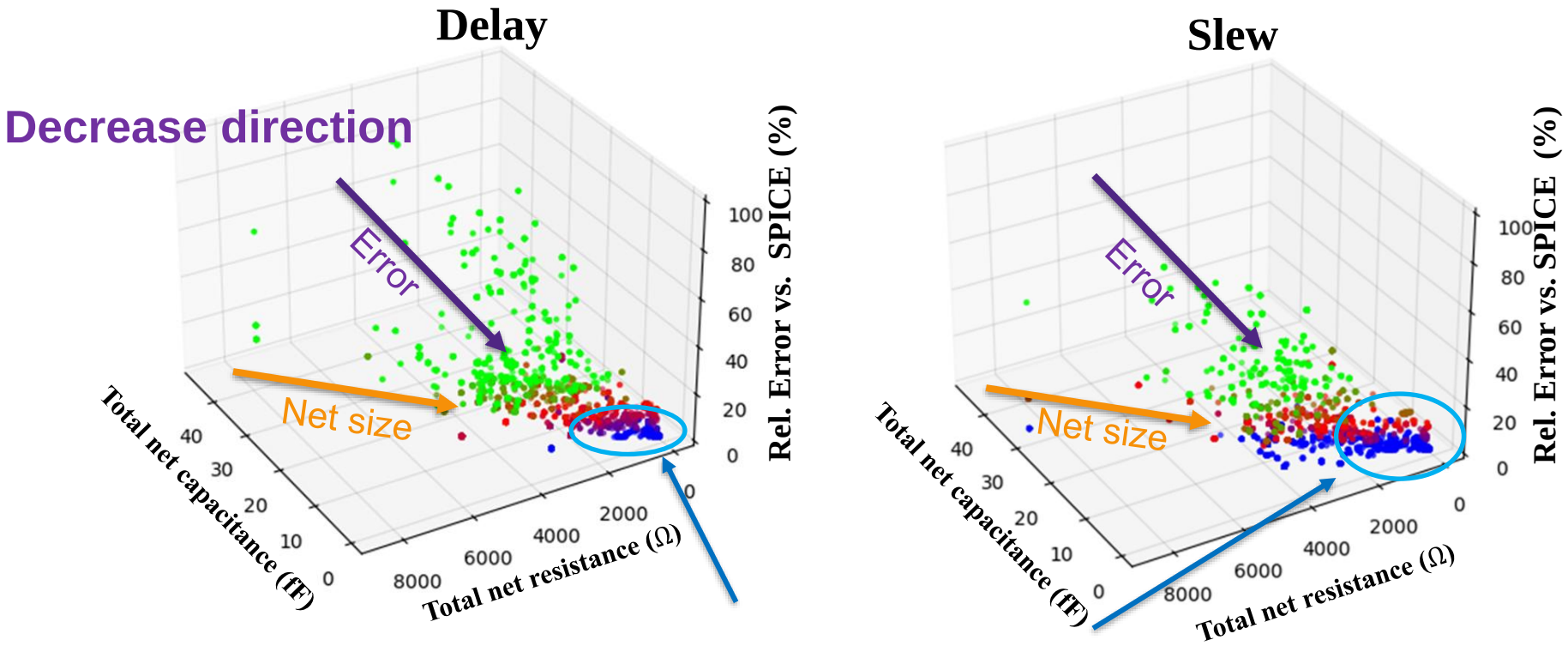


Acceleration #3:
Accelerated Driver Waveform Solving

Timing Accuracy with NLDM

- The relative error of NLDM **decreases** as the total **net resistance** and **capacitance** become **small**.

Relative Error versus Spice

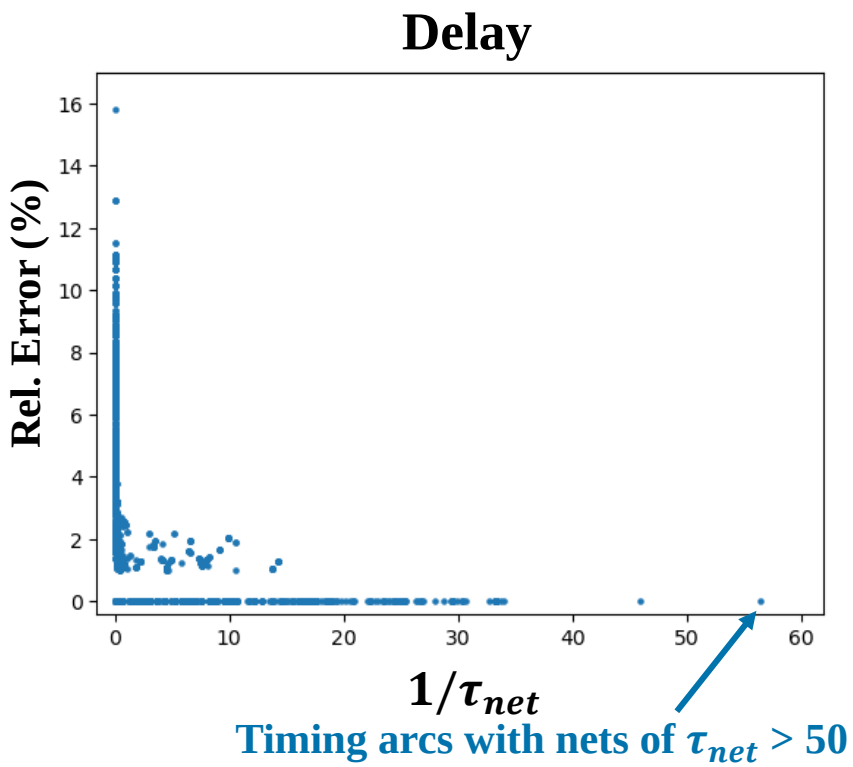


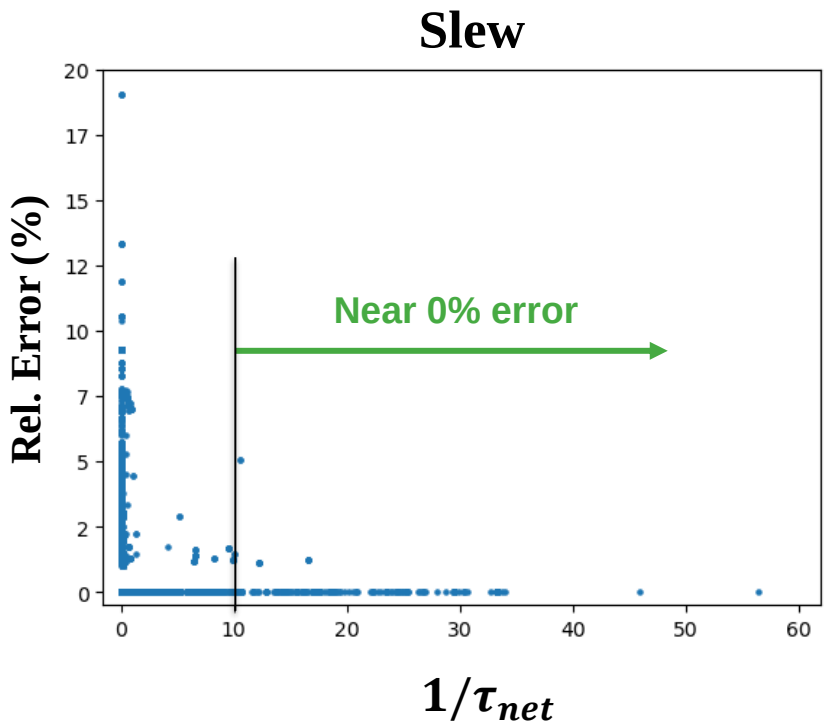
Small net with small error

Time Constant for Nets

- Define **time constant τ_{net}** to represent the net size

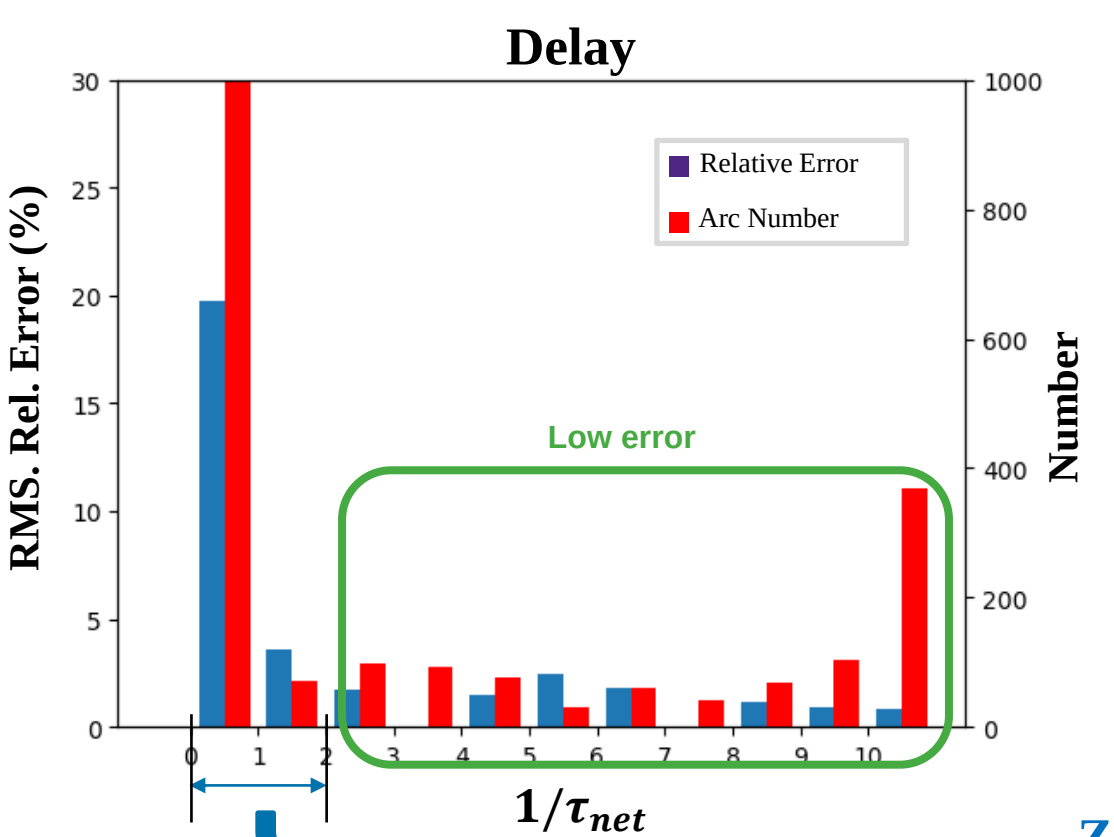
$$\tau_{net} = \sum_i C_i * \sum_j R_j$$

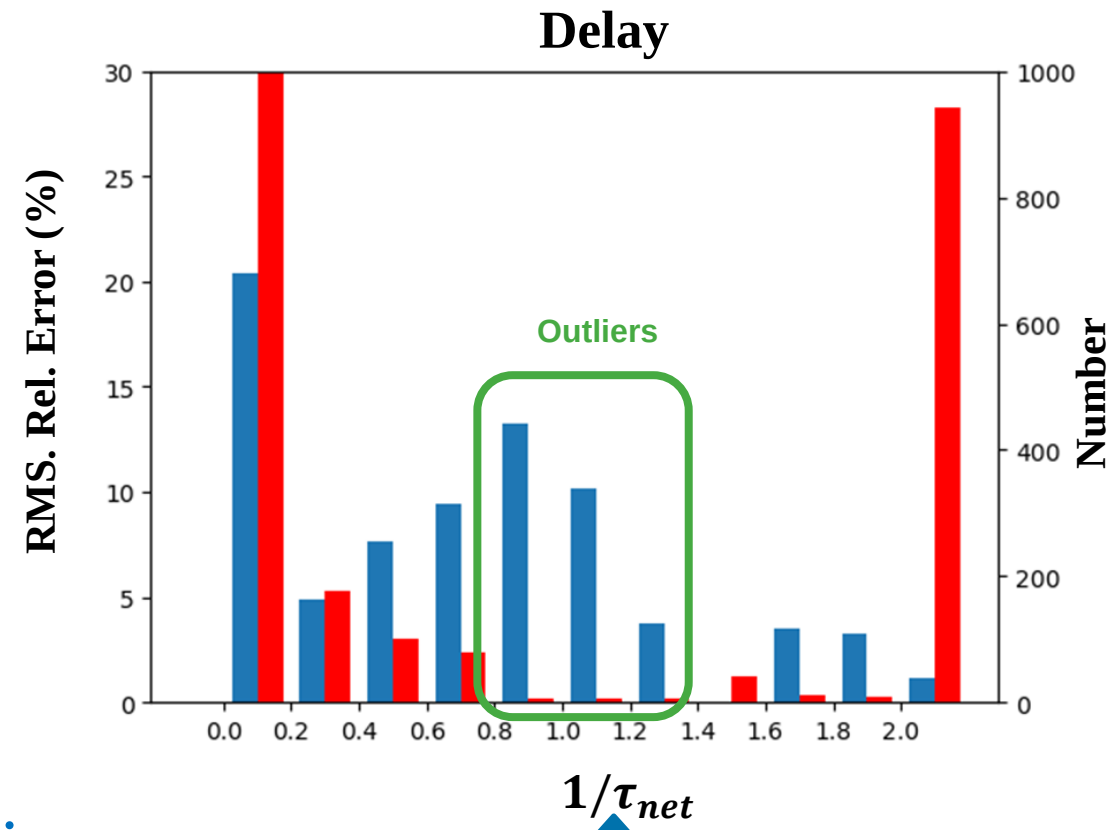




NLDM Accuracy with Time Constant

- In distribution of the error with respect to $1/\tau_{net}$, outliers are infrequent and can be ignored

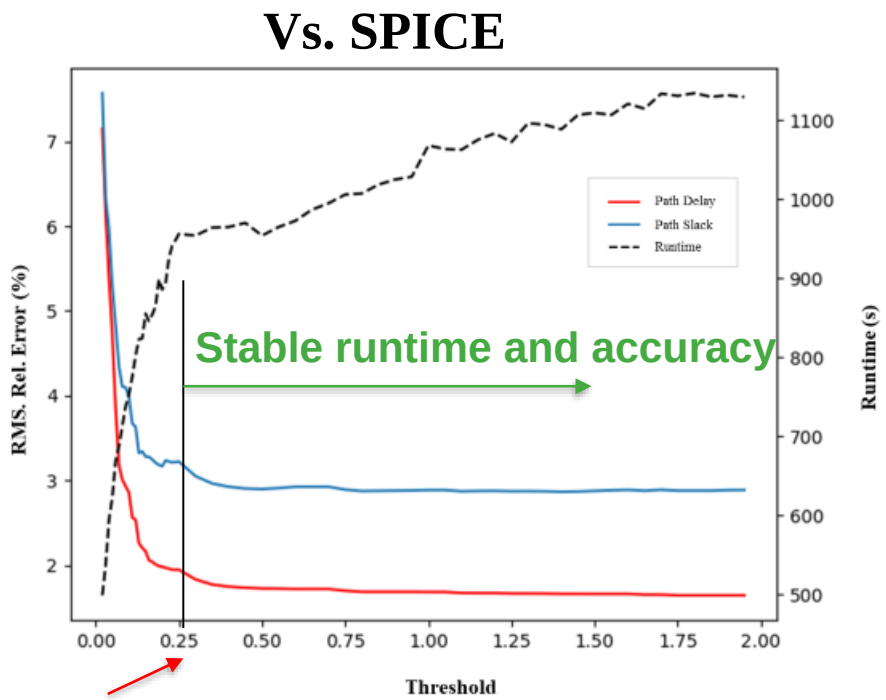


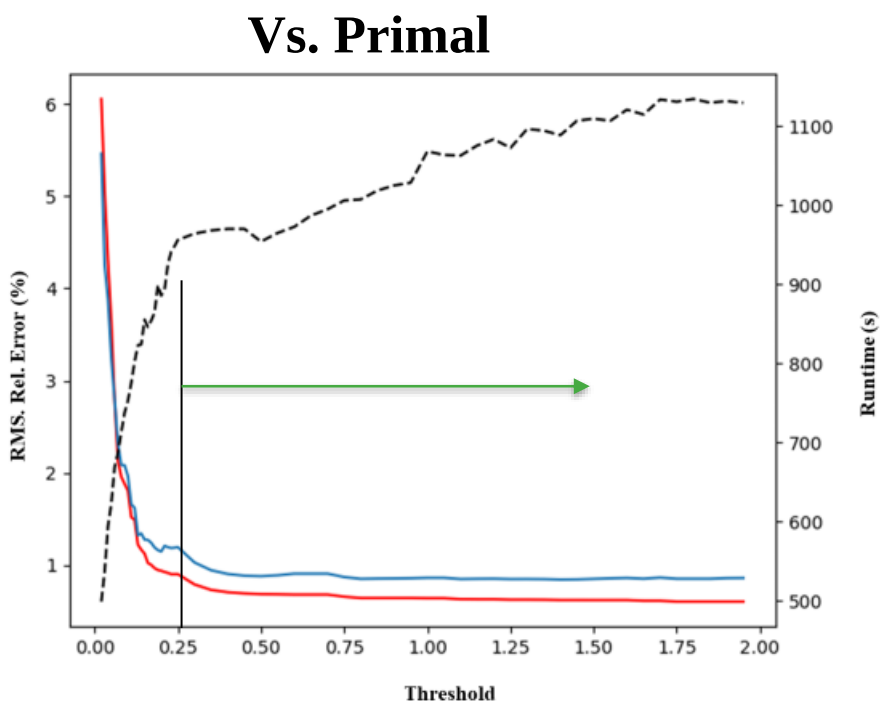


Zoom in

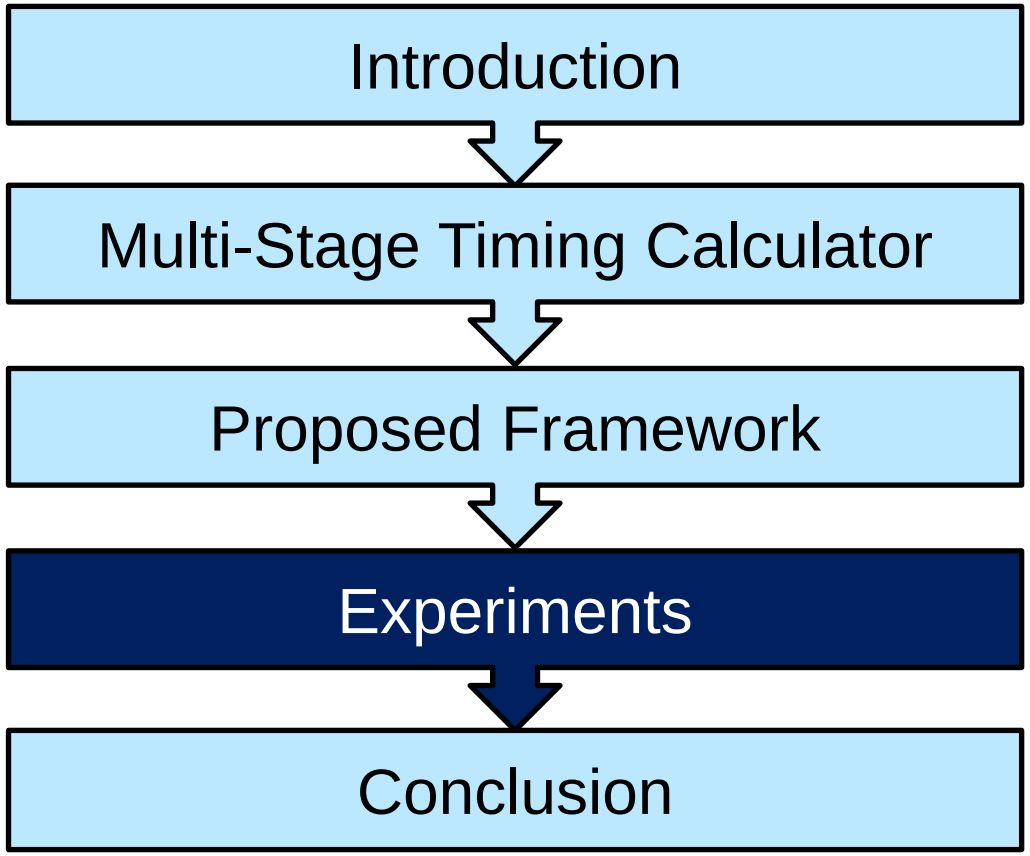
Accelerated Driver Timing Solving

- Runtime and accuracy of path delay with different thresholds for $1/\tau_{net}$
- A smaller threshold makes more nets being propagated with NLDM timing.
- The runtime and relative error become stable after the threshold greater than 0.25.





Outline


Introduction
Multi-Stage Timing Calculator
Proposed Framework
Experiments
Conclusion

Experimental Setup

- TAU 2021 Delay Calculation Tool Kit (DCTK) for parsing timing libraries and Eigen library for matrix operations

- Benchmarks adopted from **TAU2019** contest
  - Re-synthesize with **ASAP 7nm PDK** using Innovus
  - Extract SPEF files using Quantus RC
  - Calculate golden with Xyce Spice calculator

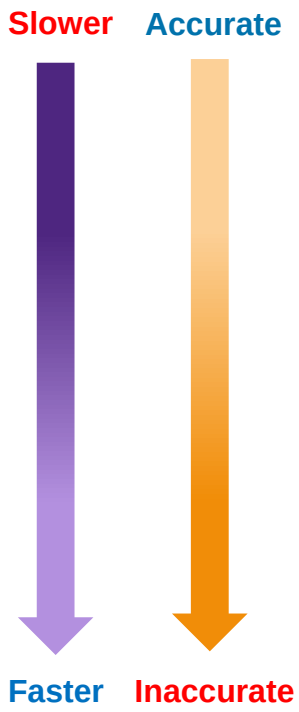
	usb_funct	vga_lcd	leon3mp	leon2
Clock Period (ps)	2000	3000	2200	64000
Number of Nets	9892	96455	723130	882561
Number of Gates	9896	96537	723208	882646
Number of Endpoints	1746	17079	108839	149381

STA Performance Comparison

- Compare different frameworks and features
  - (1) NLDM-guided timing propagation
  - (2) Fast receiver slew estimation
  - (3) Accelerated driver waveform solving

Average **98% path coverage** and **3.97X speedup**

		usb_funct	vga_lcd	leon3mp	leon2
Primal (CSM)	**Path Coverage**	100%	100%	100%	100%
	Runtime (s)	28.37	249.39	2235.53	3535.61
	Normalize	4.98	3.60	3.74	3.55
(1)	**Path Coverage**	100%	100.00%	94.00%	100.00%
	Runtime (s)	18.27	156.51	1349.40	1853.95
	Normalize	3.21	2.26	2.26	1.86
(1) + (2)	**Path Coverage**	99.80%	100.00%	90.80%	100.00%
	Runtime (s)	11.24	100.93	896.99	1211.47
	Normalize	1.97	1.46	1.50	1.22
(1)+(2)+(3)	**Path Coverage**	99.20%	100.00%	90.30%	99.00%
	Runtime (s)	5.7	69.24	598.03	996.24
	Normalize	1.00	1.00	1.00	1.00
NLDM	**Path Coverage**	93.40%	97.30%	56.80%	99.20%
	Runtime (s)	1.26	11.71	114.49	151.94
	Normalize	0.22	0.17	0.19	0.15



Path Accuracy and Multi-threaded Runtime

Path Slack/Delay Accuracy

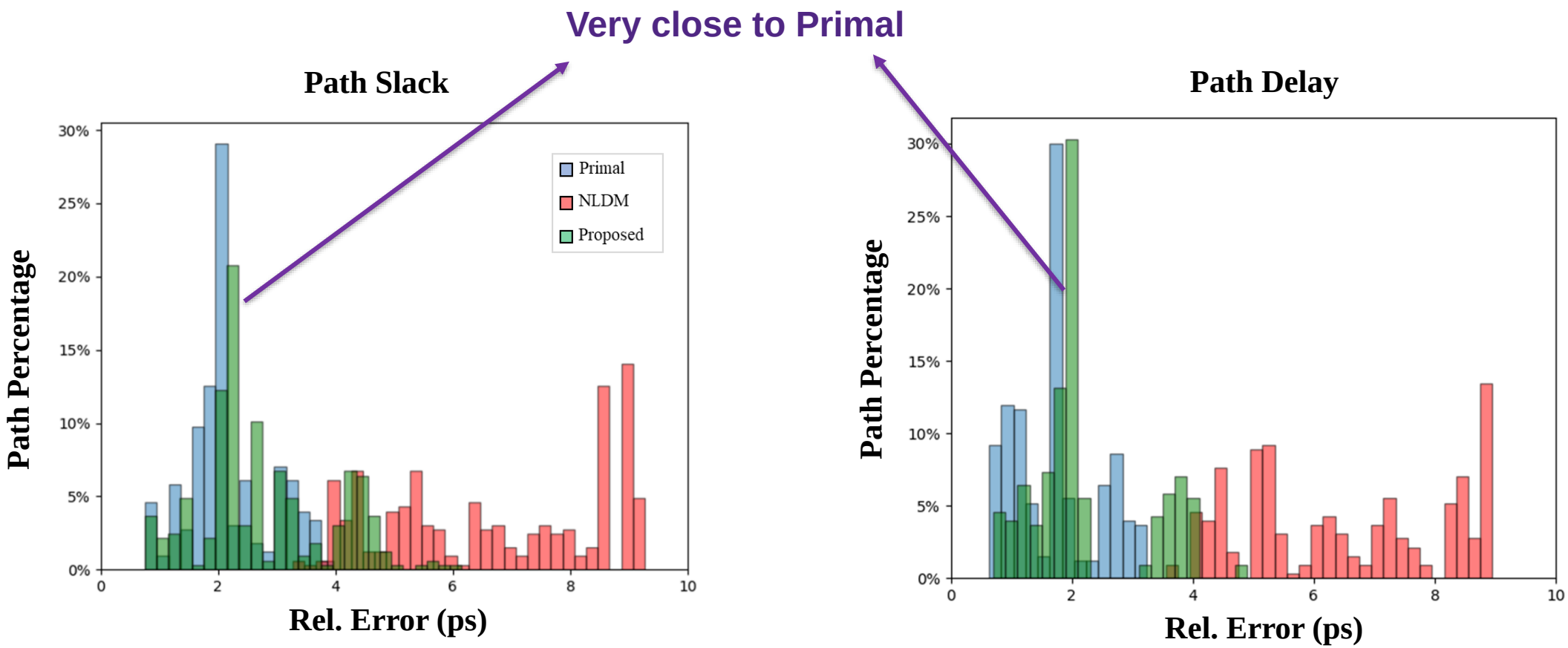
		RMS. Rel. Error		RMS. Abs. Error	
		Path Slack	Path Delay	Path Slack	Path Delay
usb_funct	Primal	3.12%	2.61%	22.53	21.83
	Proposed	4.02%	3.40%	35.83	37.42
	NLDM	5.40%	5.26%	38.97	43.94
vga_lcd	Primal	1.28%	1.04%	29.54	27.08
	Proposed	1.34%	1.23%	30.79	32.04
	NLDM	4.37%	4.50%	101	117.84
leon3mp	Primal	2.56%	1.57%	32.41	32.07
	Proposed	3.18%	1.93%	40.34	39.48
	NLDM	7.51%	7.07%	95.46	144.55
leon2	Primal	1.99%	1.81%	127.56	129.69
	Proposed	2.12%	1.97%	136.41	141.59
	NLDM	8.85%	8.66%	568.47	622.07

Runtime Speedups

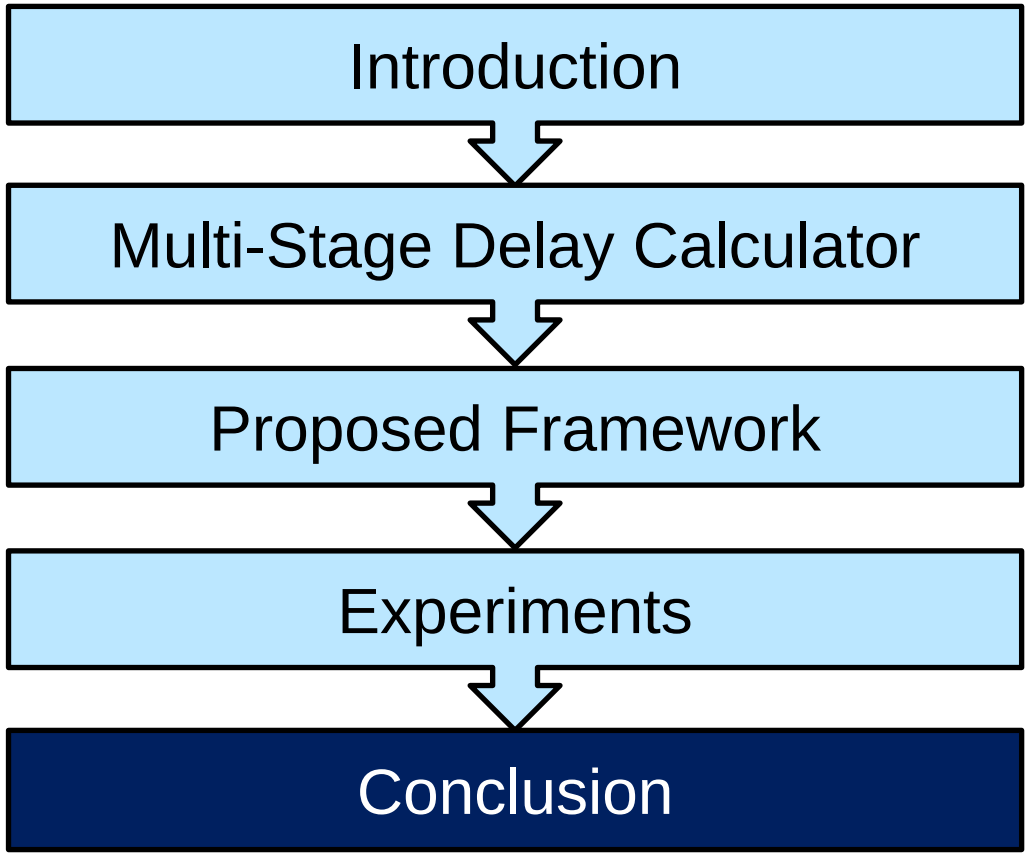
- Multi-threading with 16 threads

Circuit		usb_funct	vga_lcd	leon3mp	leon2
Proposed	Speedups	3.31	3.16	3.89	6.44
NLDM	Speedups	2.80	2.39	2.20	3.21

Path Timing Error Distribution



Outline


Introduction
Multi-Stage Delay Calculator
Proposed Framework
Experiments
Conclusion

Conclusion

- The proposed STA framework offers a solution that strikes a tradeoff between performance and runtime using CSM with multi-stage waveform propagation
  - We investigate three different bottlenecks in the primal framework and propose corresponding techniques to handle them
- The proposed techniques preserver the timing accuracy and the criticality of extracted paths while the performance is greatly improved
  - Achieve average **98%** path coverage while improve the performance **3.96X** faster
  - Can be applied on any CSM-compatible STA with **different stage timing calculator**
- Future work includes the integration with an STA engine considering other process effects at advanced technology nodes.


Thank You!