Aging-aware Static Timing Analysis with Timing Arc-level Modeling

Jingon Lee[1], Sangwoo Han[2], Ruijing Shen[2], Li Ding[2]

[1]Samsung Electronics, Foundry Business,
[2]Synopsys

SAMSUNG

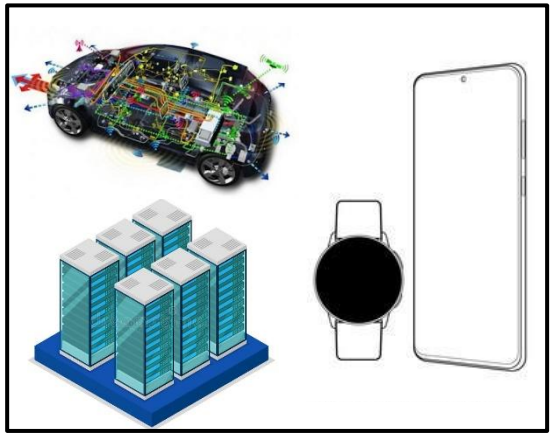
Motivation

- **Device performance degradations due to BTI, HCI induced aging**
  - Transistors gets slower degrade circuit operating frequency
  - Depending on supply voltage, temperature, and lifetime

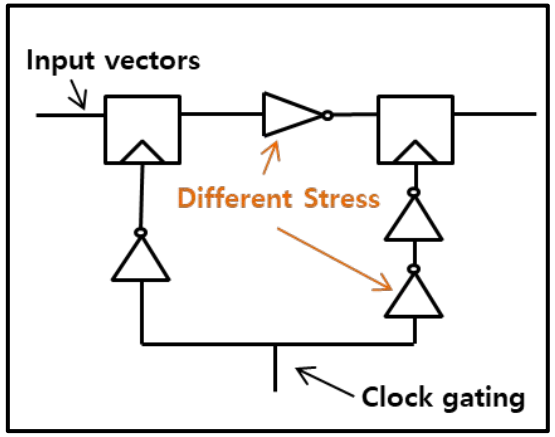
- **Estimating aging induced timing variation is challenging due to diverse factors**
  - Customer specific aging conditions (depending on what product specification is)
  - Cell, instance-level diversity (different input/output transition time, load)
  - Instance's timing arc-level diversity (multiple input output timing arcs with different Tr. topology)

  Wide range of difference in performance degradation and hard to consider at design stage.

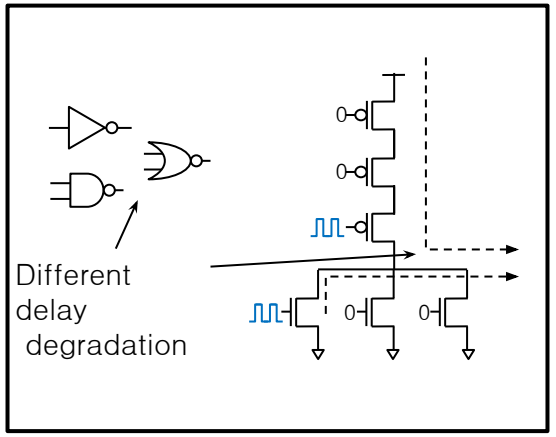
Product level



Circuit (path) level



Cell level

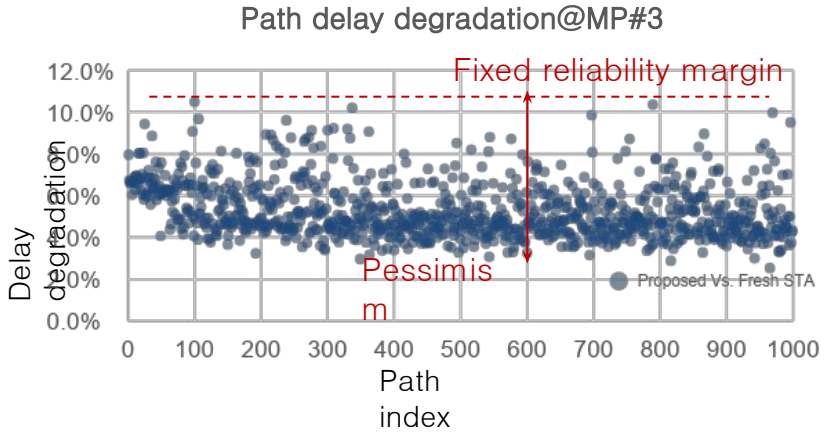


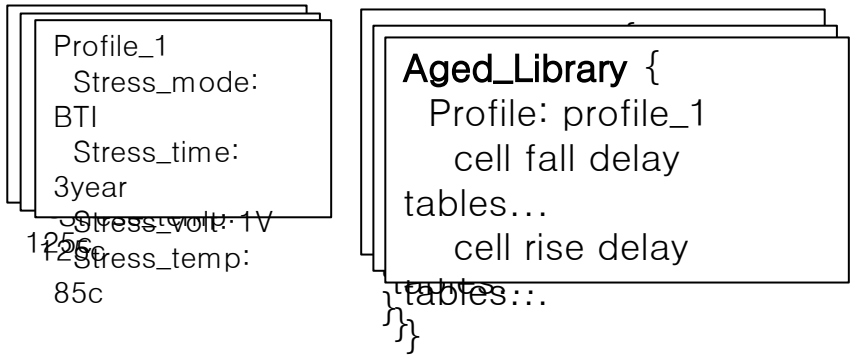
Motivation

- **Aging aware SPICE simulation: Timing consuming**

- **Static timing analysis: Limited accuracy or cost-ineffective**
  - **Fixed timing derate (margin)**
    - Pessimistic: assume the same amount of aging degradation on instances
  - **Aged library characterization based method**
    - Time consuming: Requires aged library characterization on multiple scenarios using time-consuming aging simulation
    - Expensive cost: Aged libraries of multiple scenarios that is only limited to single purpose.

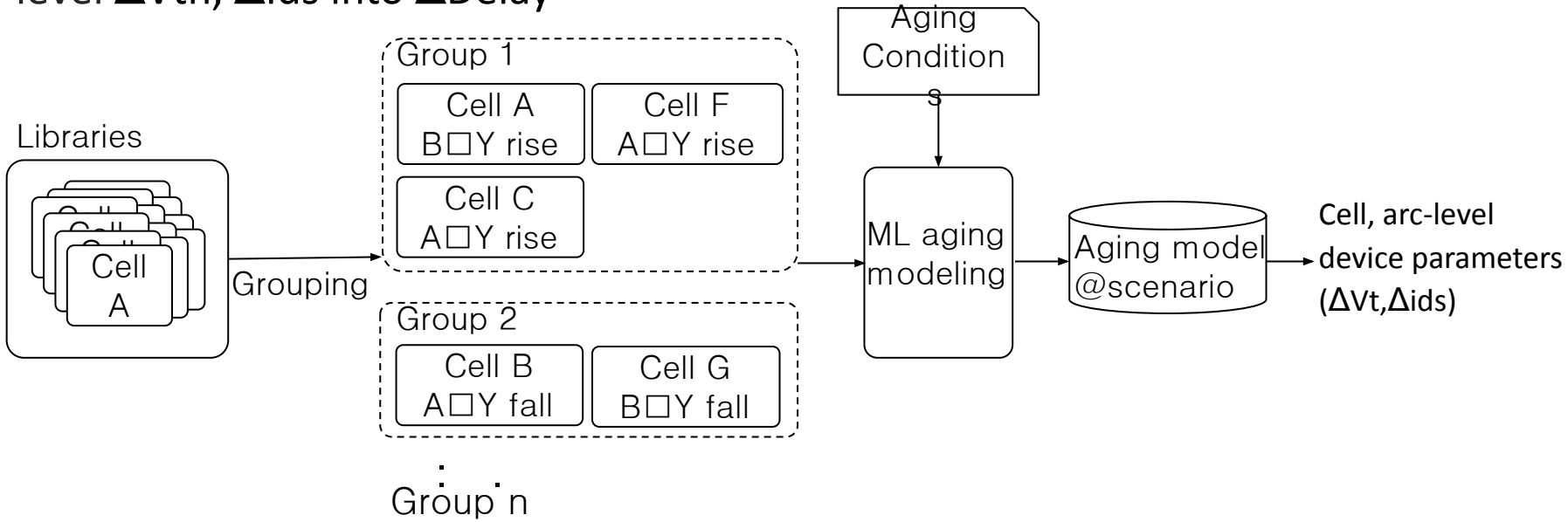
- **Estimation on aging can be achieved by modeling, not by direct characterization**





Proposed: Aging-aware STA with ML based Arc-level Aging Model

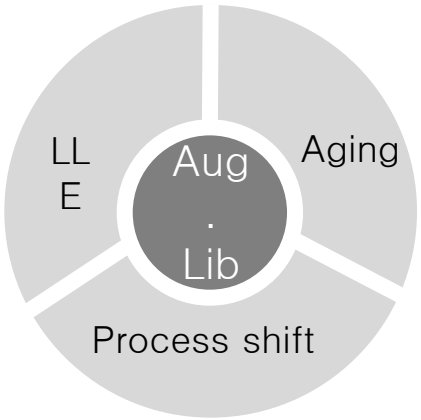
- **Objective: Building a timing model at cell & arc-level and using it at STA**
- **Constraints:**
  - **SPICE like accuracy**
  - **Easy to use**
- **Method:**
  - Grouping arcs based on characteristics
  - Measuring aging at multiple scenarios for each groups and ML-based modeling
    - Cell's timing arc-level aging is modeled into device parameters such as ΔVth, Δids
  - Converting arc-level ΔVth, ΔIds into ΔDelay



Proposed: Aging-aware STA with ML based Arc-level Aging Model

- **Augmented Library**
  - **Contains delay sensitivity with respect to device parameters**
    - Device parameters: Mobility shift, Vth shift..
    - Delay sensitivity: $delay_{shifted}$ - $delay_{norminal}$ / $delay_{nominal}$
  - **Extends timing analysis dimension from two to three**
  - **Versatile**
    - Captures Local layout effect, which is modeled in SPICE model as Vth, μ_0 shift
    - Process shift, which also modeled into device parameter shifts
    - Aging

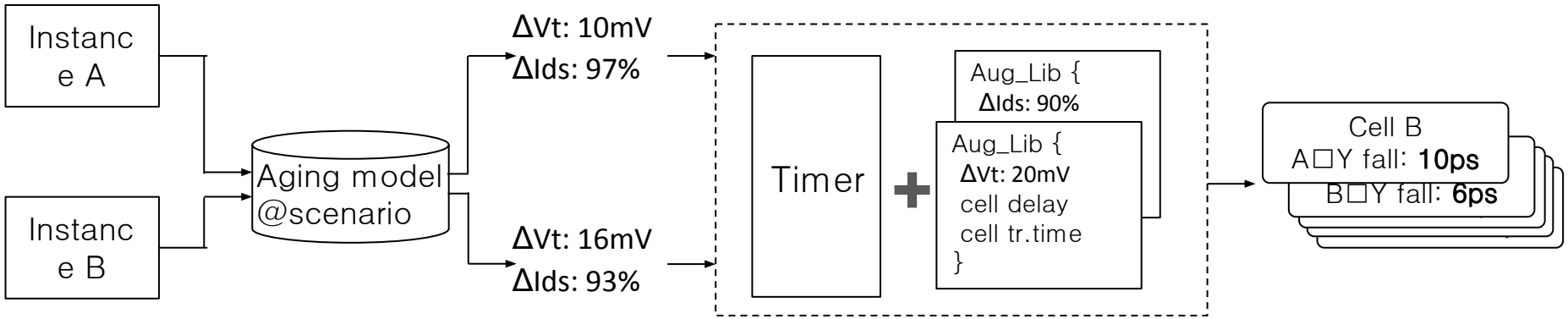
```
Cell timing {
  index1:Δvth 20mV
  index2: input tr time
  index3: load

Delay | tr_time sensitivity {
        intr1 intr2 intr3
 load1  0.21    0. 41     0.55
 load2  0.31     0.61     0.90
 load3  0.61     0.82   1.17 }
```



Application of augmented lib.

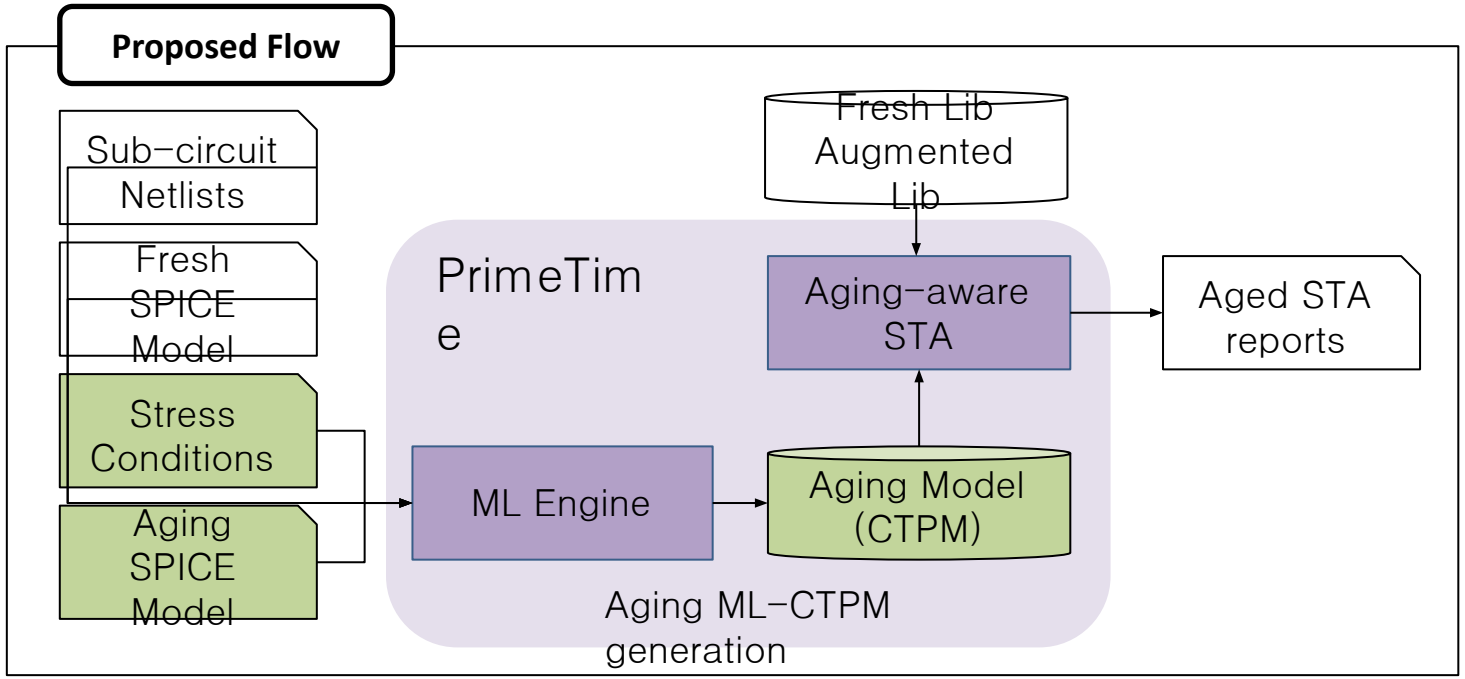
Proposed: Aging-aware STA with ML based Arc-level Aging Model

- **Estimate aged delay of a cell**
  - Estimate aging impact on ΔVth, Δids
  - Convert it into Δdelay using augmented library
  - Aging under different transition time and load can be captured
  - Arc-level (tr-topology) diversity can be captured



Proposed Flow

- **Generating aging model**
  - Inputs: Fresh lib, SPICE model, cell-level SPICE netlists, aging model, aging conditions
  - Outputs: Aging model (compact timing power model; CTPM)
  - Steps
    - Grouping cell and automatically simulating delay degradation by aging SPICE sim.
    - Setting a specific stress scenario or scenarios for regression on various stress scenarios.
    - ML powered modeling



Proposed Flow

- **Running STA with aging model CTPM**
  - Input: Fresh lib, augmented lib, aging CTPM, STA collaterals (sdc, SPEF ..)
  - Output: Aging awared timing reports
  - Steps
    1. STA setting remains same
    2. Read_ctpm & library mapping
    3. update timing: Delay of a cell is re-calculated using Δ param. and timing sensitivity at augmented
- **Reporting aged STA results**

Example Script

```
#enable CTPM
source enable_pvt_explorer.tbc
link_library lib_4nm_0p5400v_m25c
read_verilog design.v
link_design top_design
read_parasitics parasitic.spef

#augmented lib mapping
define_sensitivity_lib_mapping lib_4nm_0p5400v_m25c ₩
-side_file augmented_lib_4nm_0p5400v_m25c

#read CTPM
read_ctpm agingmodel.ctpm
#define aging condtion
set_user_specified_aging 0.85v/125c/10y/100%
update_timing
```

Results

- **Lib arc-level delay validation**
  - Error: -0.05% (avg.), 1.08% (std.) @ 0.54V, -25c
  - Estimate aged cell delay from CTPM and compared 1000 cells' rise, fall delay of aged library

- **Design-level aging-aware STA evaluation**
  - Design: 4nm CPU (running at 0.54V), three multiple aging conditions

Library arc-level delay validation						
run_name	Rise			Fall		
	avg	Std.	99th_percentile	avg	Std.	99th_percentile
Delay-preCTPM	-5.95%	4.74%	16.97%	-2.19%	10.13%	25.75%
Delay-postCTPM	**-0.02%**	**0.70%**	**1.66%**	**-0.08%**	**1.46%**	**3.48%**
Slew-preCTPM	-8.27%	5.30%	20.60%	-2.29%	7.09%	18.78%
Slew-postCTPM	**0.06%**	**1.22%**	**2.89%**	**-0.01%**	**2.36%**	**5.51%**

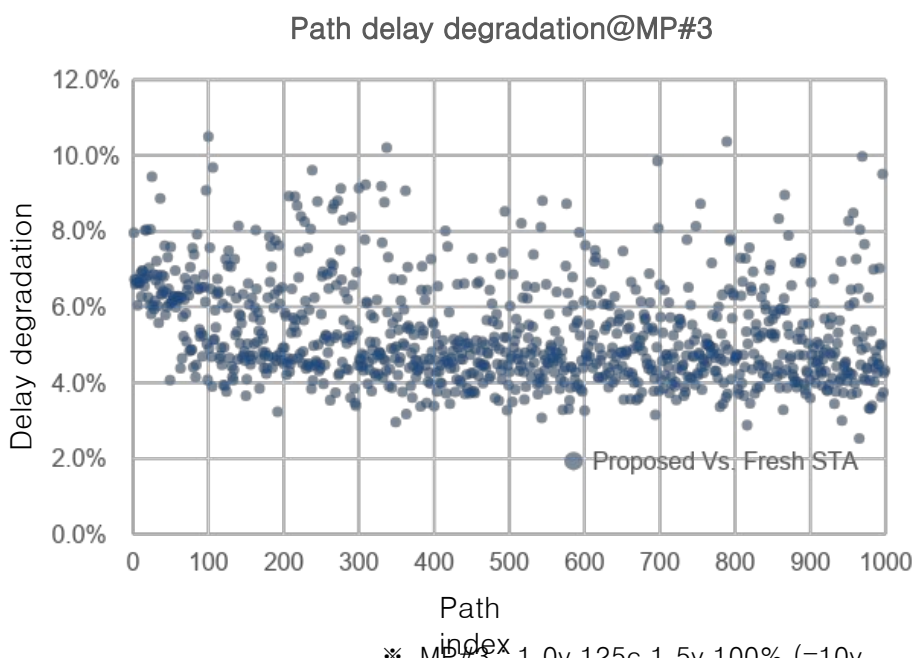
4nm rvt 0p54v m25c
(Stress: 0.9v 125c 10y w/ HCI,BTI and BTI Recovery)

4nm CPU		#paths	Mean(%)	Std(%)	Max(%)	Min(%)	Pass rate (%)	Outlier
MP#1	PT_FRESH vs SPICE Aging	1000	-1.5	0.5	-0.3	-3.3	100%	0
	PT_CTPM vs SPICE Aging	**1000**	**-0.1**	**0.3**	**1.3**	**-1.1**	**100%**	**0**
MP#2	PT_FRESH vs SPICE Aging	1000	-4.2	0.8	-2.7	-7.3	73%	273
	PT_CTPM vs SPICE Aging	**1000**	**-0.4**	**0.7**	**2.8**	**-1.9**	**100%**	**0**
MP#3	PT_FRESH vs SPICE Aging	1000	-5.3	0.9	-3.5	-8.8	17%	826
	PT_CTPM vs SPICE Aging	**1000**	**-0.6**	**0.8**	**3.1**	**-2.1**	**100%**	**0**

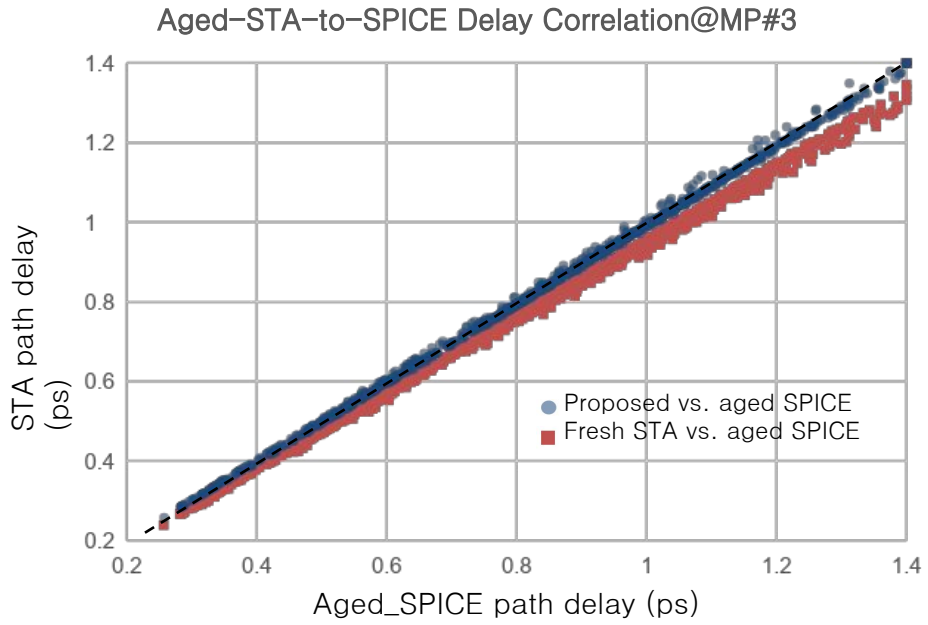
※ MP#1 : 0.79v 125c 10y 100%
※ MP#2 : 0.9v 125c 10y 100%
※ MP#3 : 1.0v 125c 1.5y 100% (=10y 15%)
Outlier definition: err > 4.5% and 4.5ps

Results

- **Design-level aging-aware STA evaluation**
  - Comparing aged SPICE to Fresh (PT_Fresh) and proposed (PT_CTPM)
  - Path delay degradation ranged from 2.5% to 10% are well estimated. (@MP#3)
    - Error: -0.6% (avg.), 0.8% (std.), 3.1% (max.), -2.1% (min.)
    - 1000 paths, ref: aging SPICE@MP#3



※ MP#3 : 1.0v 125c 1.5y 100% (=10y 15%)
Outlier definition: err > 4.5% and 4.5ps



Summary

- **Estimating aging aware timing simulation & analysis is challenging**
  - Aging aware timing simulation: runtime burden
  - Design pessimism: derate based STA
  - Expensive solution: aged-lib. based STA

- **Proposed: Modeling (estimation) + Analysis**
  - Modeling aging effect of each cell and timing arc by machine-learning engine
  - Consuming versatile augmented library to estimate aged delay
    - Reduce cost and effort to library characterization
    - Can also be used in in parameter variation-aware timing analysis
  - Highly accurate aging Estimation & timing analysis
    - Library timing arc-level delay error: -0.05% (avg.), 1.08% (std.)
    - Design-level path delay error: -0.6% (avg.), 0.8% (std.), 3.1% (max.), -2.1% (min.)

Reference: R. Shen and L. Ding (2021), "Method to model timing behavior using augmented sensitivity data for physical parameters", US patent No. 17/540,774, Filed on Dec 2, 2021.