Statistical Static Timing Analysis

A New Approach to Deal with Increased Process Variability in Advanced Nanometer Technologies

London, *ICCV*, May 12th, 2009

Cristiano FORZAN and Davide PANDINI

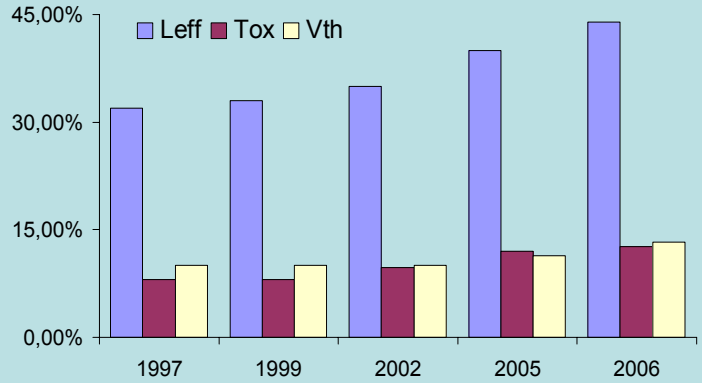
Central CAD and Design Solutions, Agrate Brianza, Italy

Outline

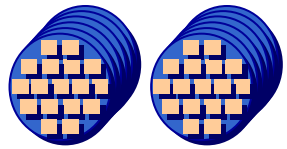
- Introduction and motivation
- Corner-case analysis limitations
- Statistical static timing analysis
- Statistical library characterization
- Mismatch characterization
- Pilot project on SSTA
- Conclusions

Statistical Static Timing Analysis (SSTA)

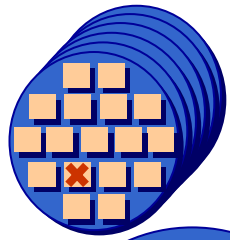
3σ parameter total variation relative to nominal value [Source: S. Nassif, IBM]



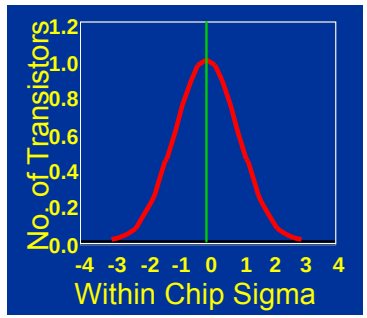
Lot-to-Lot



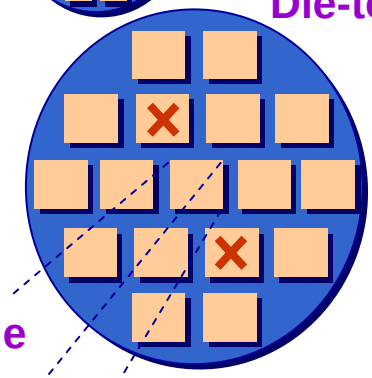
Wafer-to-Wafer



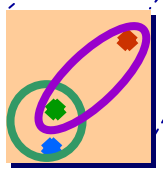
Within-Die Variation



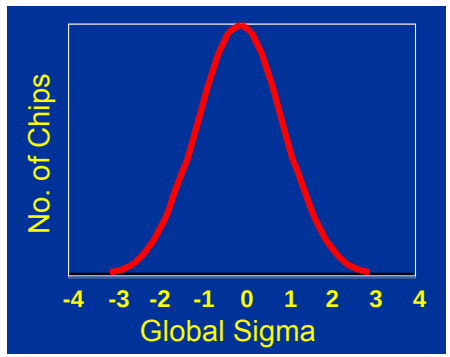
Die-to-Die

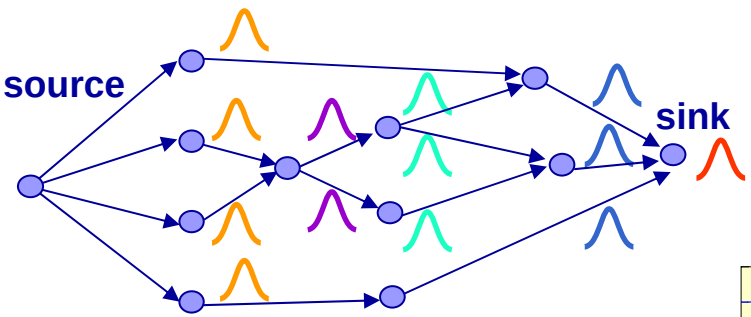


Within-Die

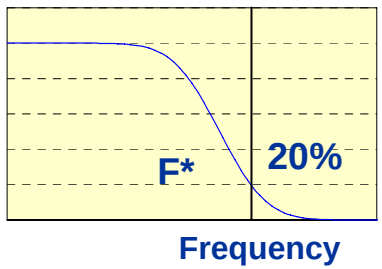


Die-to-Die Variation

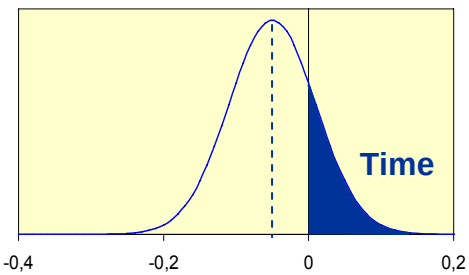




Design Yield



Design Slack



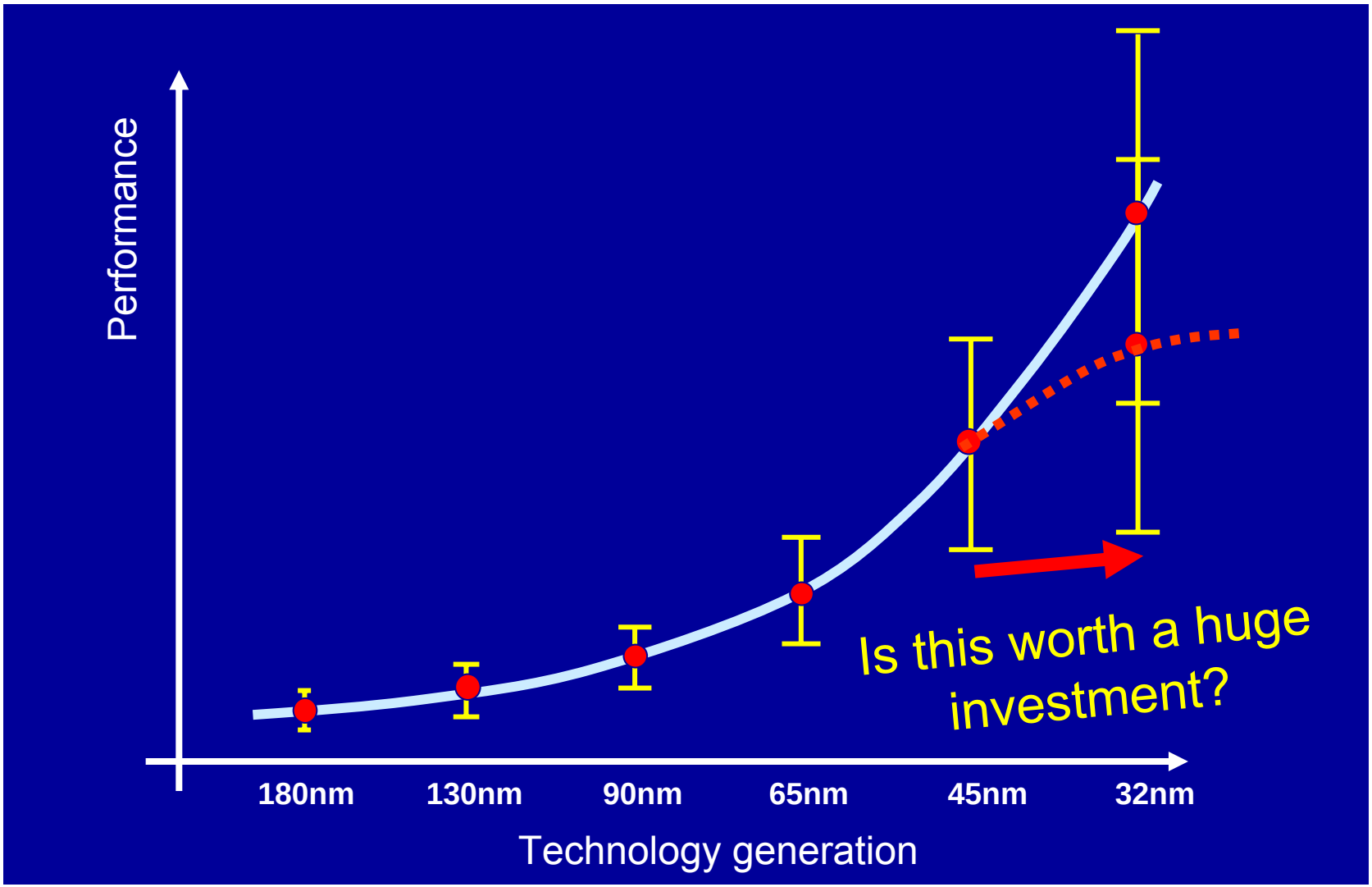
Variations Impact on Delay

Parameter	***Delay Impact***
BEOL metal (Metal mistrack, thin/thick wires)	-10% → +25%
Environmental (Voltage islands, IR-drop, temperature)	±15 %
Device fatigue (NBTI, hot electron effects)	±10%
V_t and T_{ox} device family tracking (Can have multiple V_t and T_{ox} device families)	± 5%
Model/hardware uncertainty (Per cell type)	± 5%
N/P mistrack (Fast rise/slow fall, fast fall/slow rise)	±10%
PLL (Jitter, duty cycle, phase error)	±10%

Requires 2^{20} timing runs or [-65%,+80%] guard band!

Source: L. Stok, IBM

Technology Trends



Corner-case Static Timing Analysis

- Process parameter variations are modeled in STA with corner-case analysis
  - *Best*-, *nominal*-, *worst*-case corner SPICE parameters are extracted
  - STA is performed several times each time using one corner (number of corners is increasing exponentially)
  - Statistical variations in the underlying silicon are hidden
- Traditional STA accounts for uncertainty with *worst*-case analysis
  - It determines each device and interconnect delay under worst process and operating conditions
  - It assumes that if a circuit functions according to design specs under worst operating conditions then it will operate under normal conditions as well

Corner-case Approach Limitations

- Number and magnitude of variations is increasing: can be pessimistic and risky
- Cannot properly consider within-die variations: use of heuristic derating factors or additional margins
- Difficult to determine *worst*-case for devices and interconnects at the same time
- Cannot provide feedback to designers
- Cannot predict parametric yield curve

- It is the solution we have today, but we must work aggressively on statistical timing analysis

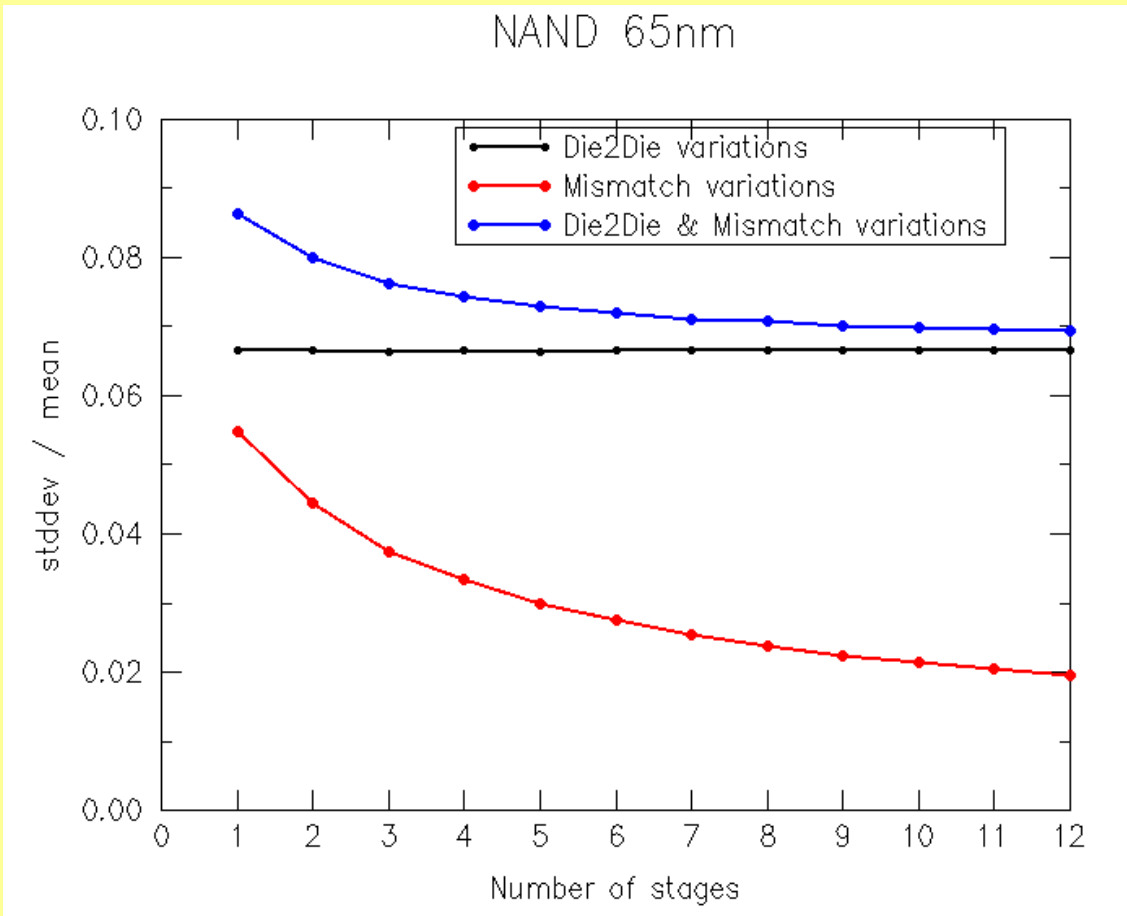
Why SSTA? Model the REAL Corner

- There is no single process corner which gives the worst timing results
- Process corner is a function of:
  - Input slew/output load
  - Cells
  - Path length
  - Slack vs. arrival/required
  - Design vs. path

- SSTA can find the process corner for each path and for the entire design

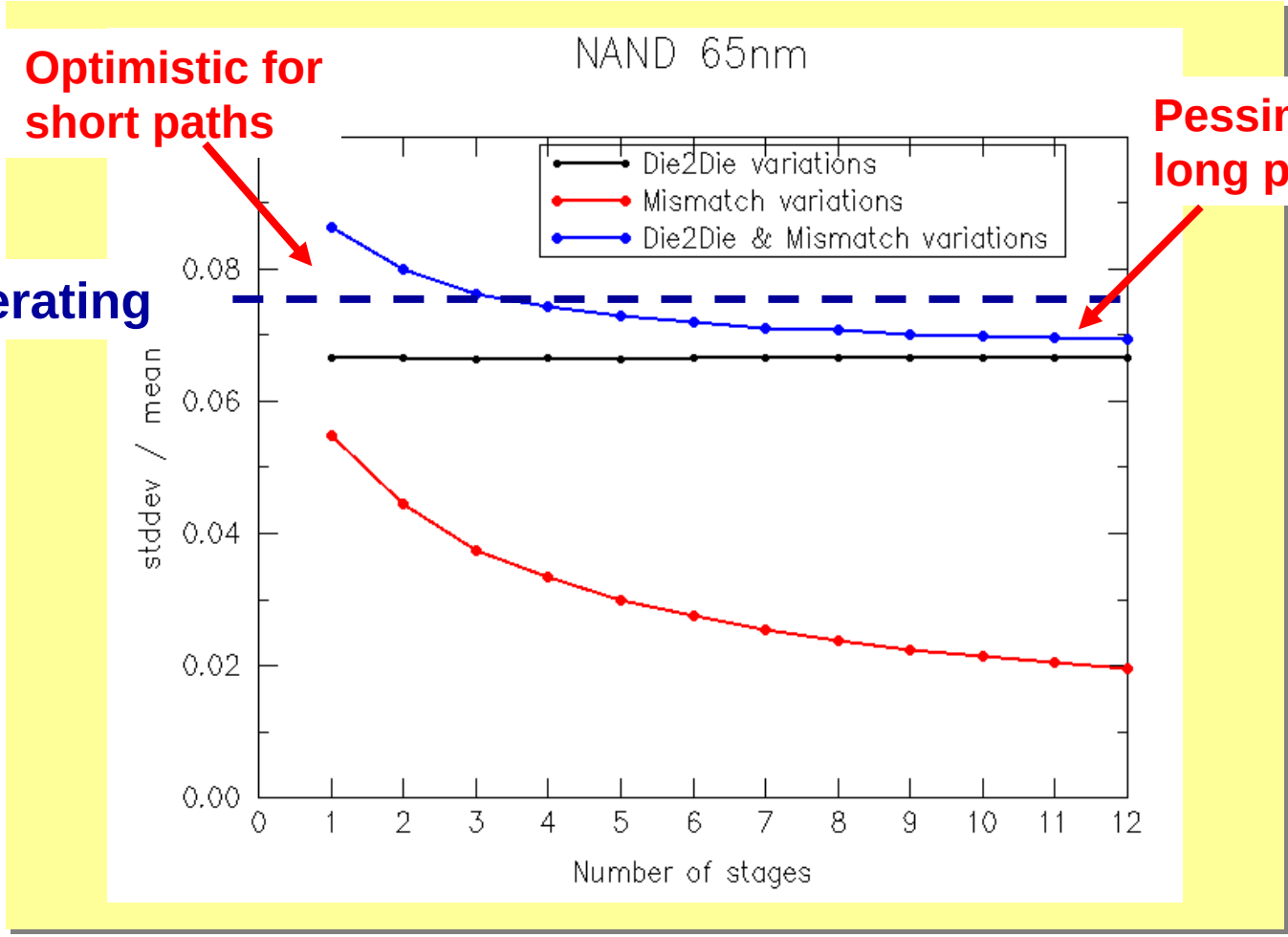
Path Length

- For different path lengths we measured D2D variations, mismatch variations, and the *rms* sum



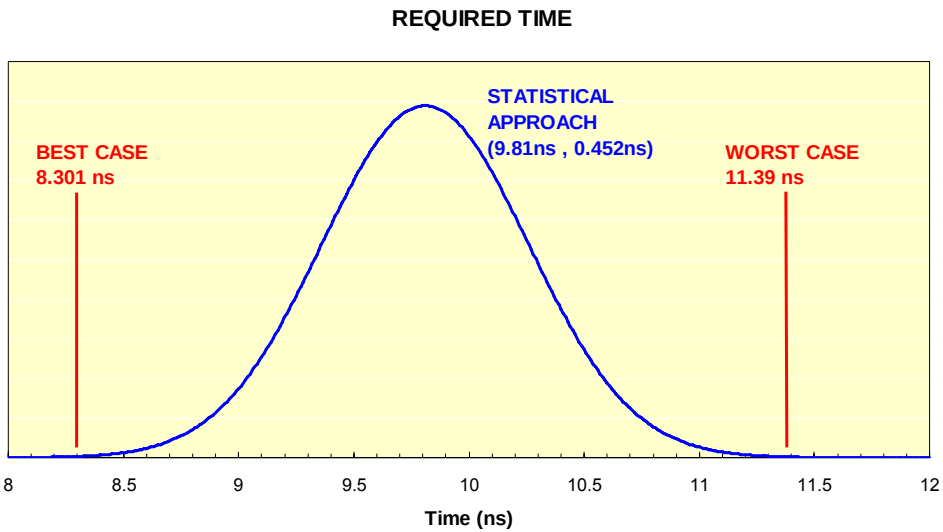
Path Length

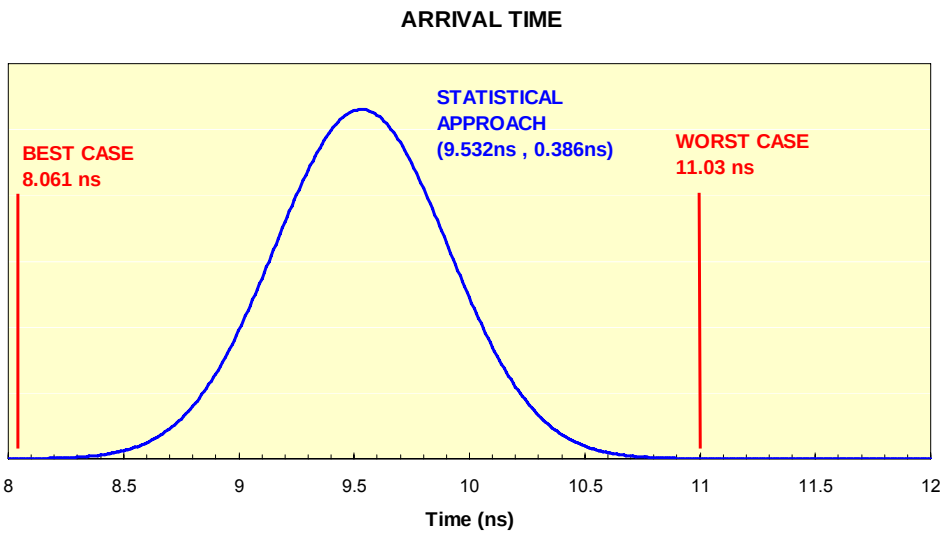
- Derating cannot model OCVs due to mismatch correctly



Slack vs. Required/Arrival Times

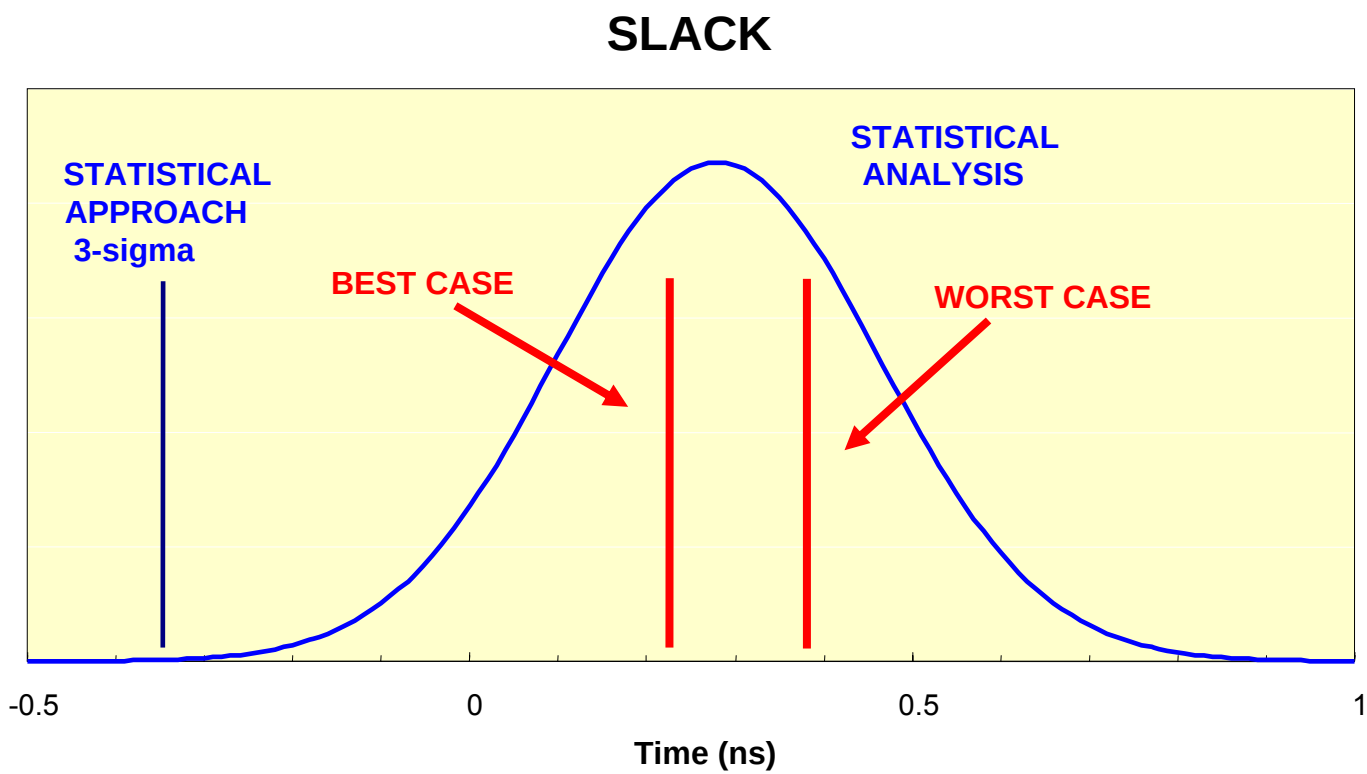
- Statistical path analysis vs. *worst*-case path analysis
- Required and arrival times are bounded by *worst*- / *best*-corners
- Slack is: (*required – arrival*). Is it bounded?





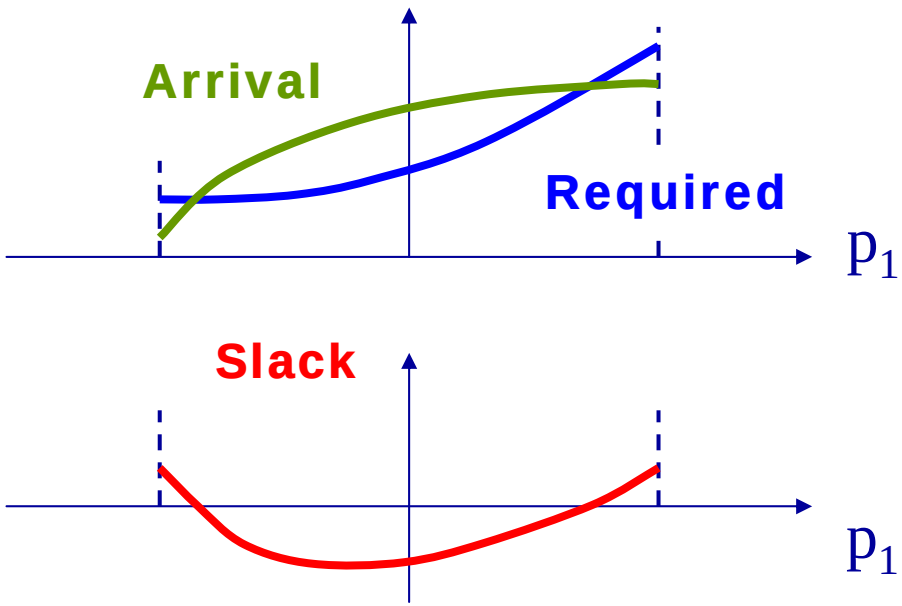
Slack Corner is Not the Process Corner

- 3σ slack is NOT bounded by *best-* / *worst*-case



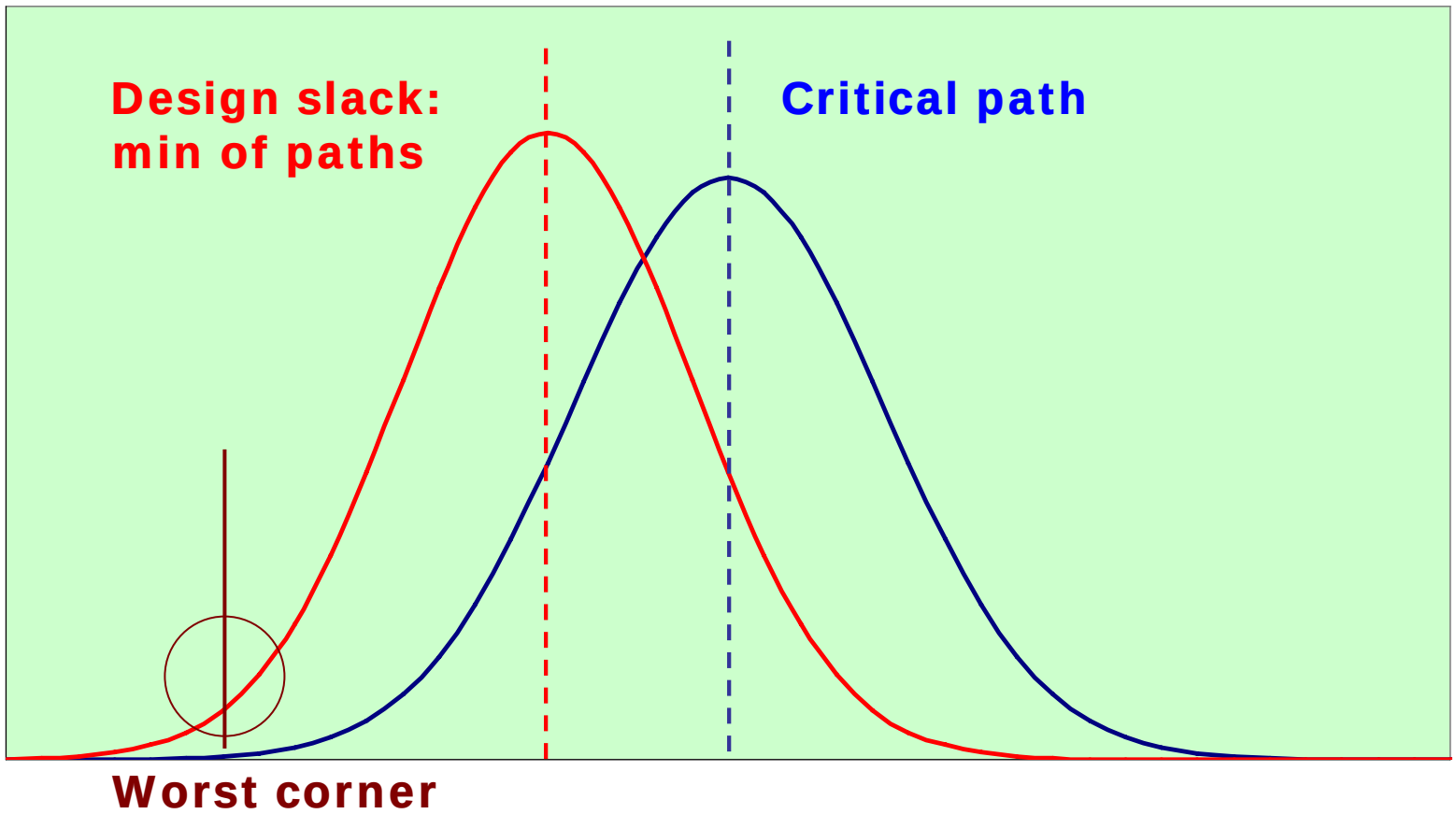
Slack vs. Required/Arrival

- 3σ slack is not necessarily at *worst-* / *best*-corners
- Typically due to uncorrelated required/arrival paths
- Required and arrival times are monotonic wrt process variations while slack is not



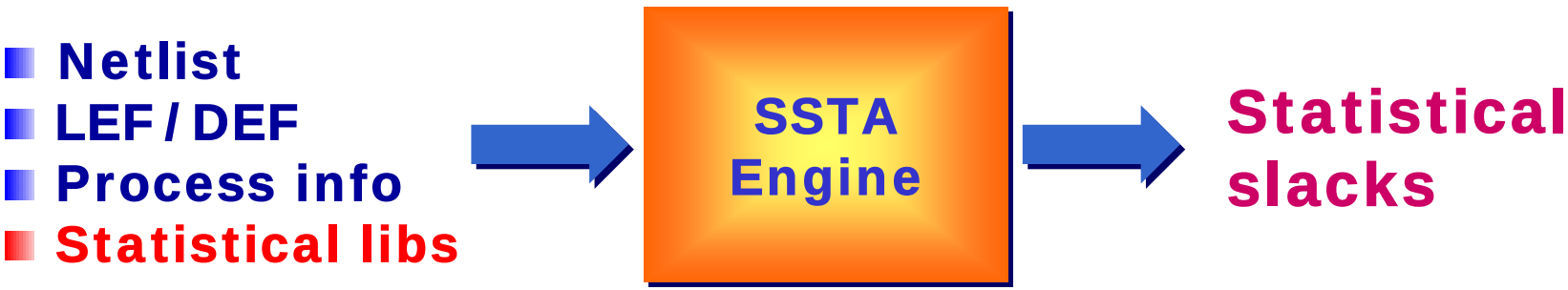
Design Slack vs. Path Slack

- Max/min of multiple paths will shift the nominal value
  - It is *design-specific*



Need for Statistical Static Timing Analysis

- Today sign-off is corner- or case-based
  - Would require about 2^{20} timing runs to cover all corners!
- Cumbersome, risky, and pessimistic all at the same time!
- Our design/synthesis methods do not properly target robustness, nor do our timing tools measure robustness or assess robust design
- Solution: statistical timing and optimization



- SSTA requires a standard cell model that accounts for the process parameter variations

Statistical Library Characterization

- New and tough challenges for standard cell library characterization
  - Better accuracy
    - Must account for global, systematic, and random variations
    - Transistor-level details exposed to the characterization tool
  - Characterization time
    - Increasing number of characterization grid points
    - More library cells
    - Larger number of Power Supply / Temperature corners (NBTI effect)
    - Huge number of circuit simulations
  - Large file sizes
    - Difficult to handle/transfer

Statistical Library Format

- A unified format should be adopted
  - Si2 - Open Modeling Coalition
  - TSMC Reference Flow
    - Use the existing Liberty format without modification
    - Use multiple Liberty files to store delay and/or sensitivities at different values of the process parameters
  - Extreme DA
    - Use an extension of the Liberty format
  - SYNOPSYS PrimeTime-VX
    - Use Liberty Composite Current Source
  - Others…

Extreme DA Statistical Library Format

- The look-up tables collect both the nominal and the variations of the quantities (delay, transition times, input pin capacitance, …) wrt the selected variational parameters

```
lu_table_template (“table_5”) {
   …
   variable_3 : “variational_parameter”
   …
   index_3 : (“NOMINAL nsigma_poly_cdvar nsigma_active_cdvar
nsigma_go1n_dtoxm nsigma_nsvt_vth0 nsigma_psvt_vth0”);
}
…
cell_rise (“table_5”) {
   values(“13.8, 0.569, 0.0379, 0.034, 0.009, 0.424”,
“54.70, 2.198, 0.279, -0.001, 1.75, 0.324”, …);
}
```

Deviation of the cell_rise wrt a variation of nsigma_poly_cdvar parameter

Library Cell Characterization with Sensitivities

- SSTA tool must recharacterize libraries for physical variations and accuracy
- All quantities are characterized with variations

$$delay = delay_{\text{nom}} + f(p_1, p_2, \cdots, p_k)$$

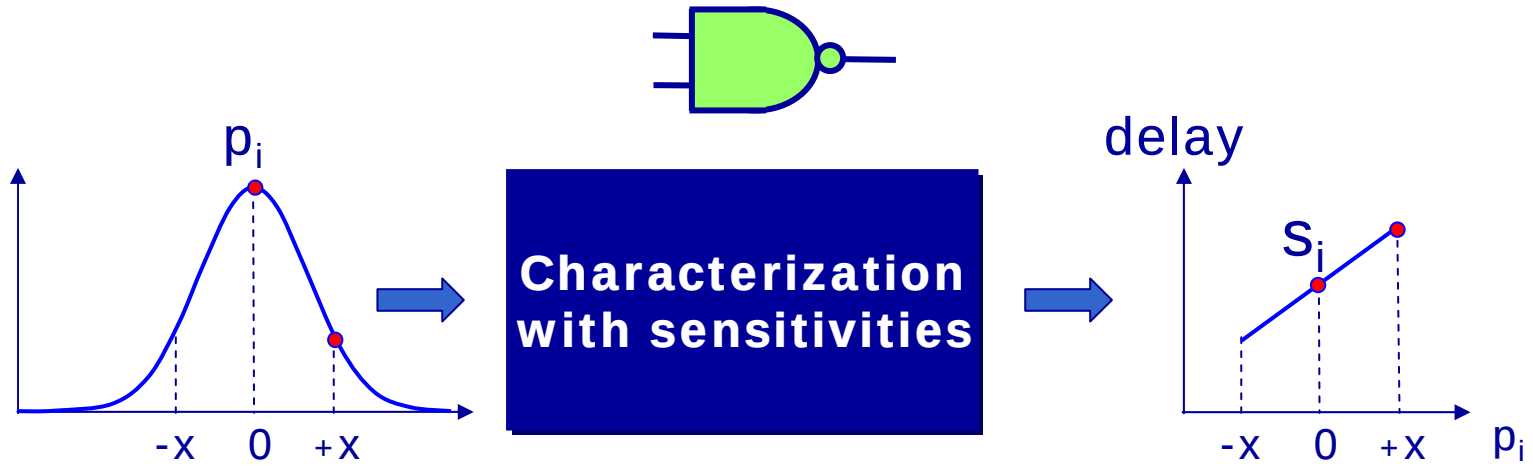
- For small changes in the process parameters we can write a Taylor expansion of the form (Linear sensitivity approach)

$$delay = delay_{\text{nom}} + \sum_{i=1}^{k} \frac{\partial delay}{\partial p_i} \cdot \Delta p_i$$

Sensitivity

Library Cell Characterization with Sensitivities

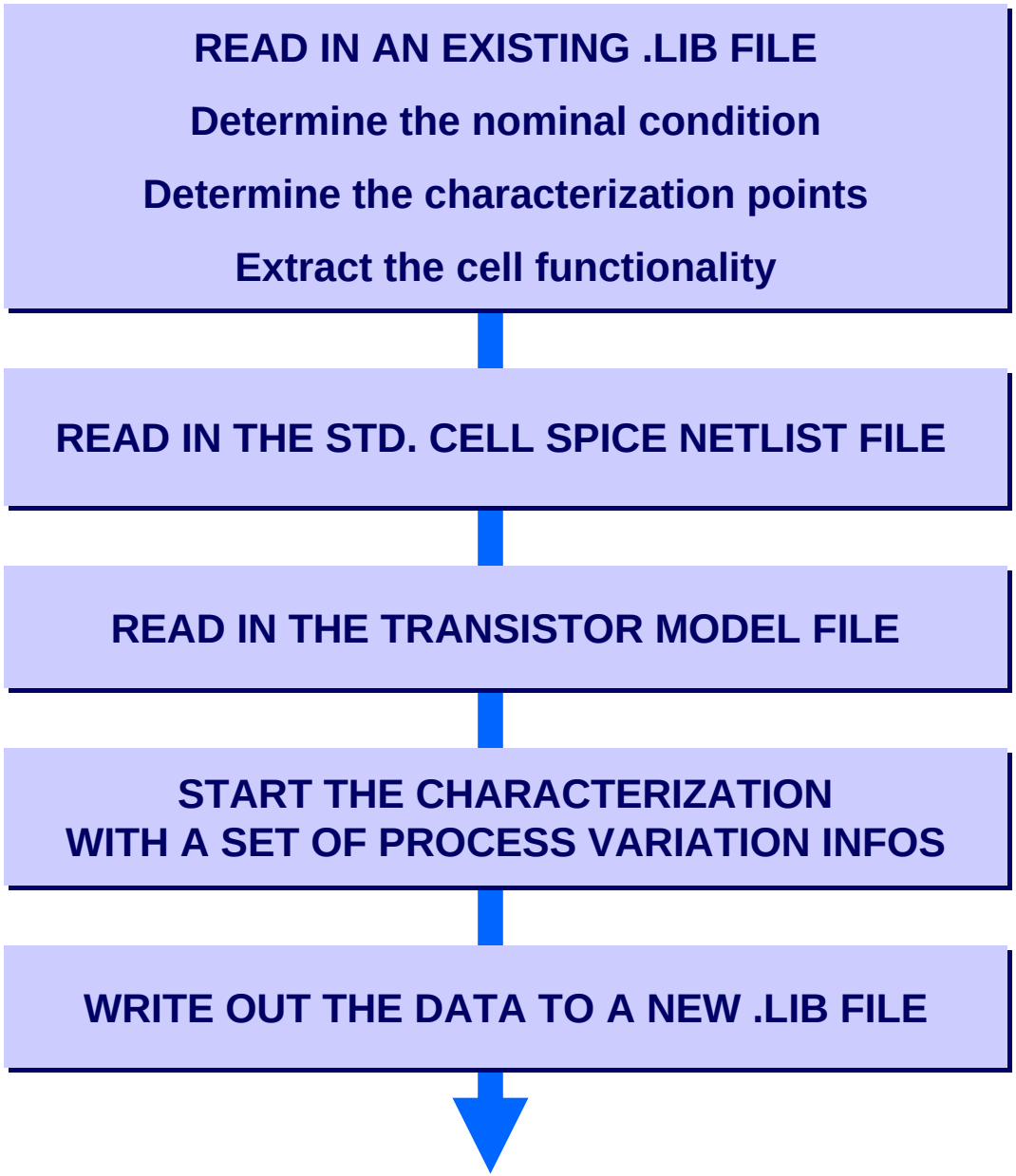
- For each timing quantity (propagation delay, output slew, …) both the nominal value and its sensitivities wrt the selected parameters are measured



- During the delay calculation phase, the delay can then be computed based on the following relationship

$$delay = delay_{\text{nom}} + \sum_{i=1}^{k} S_i \cdot \Delta p_i$$

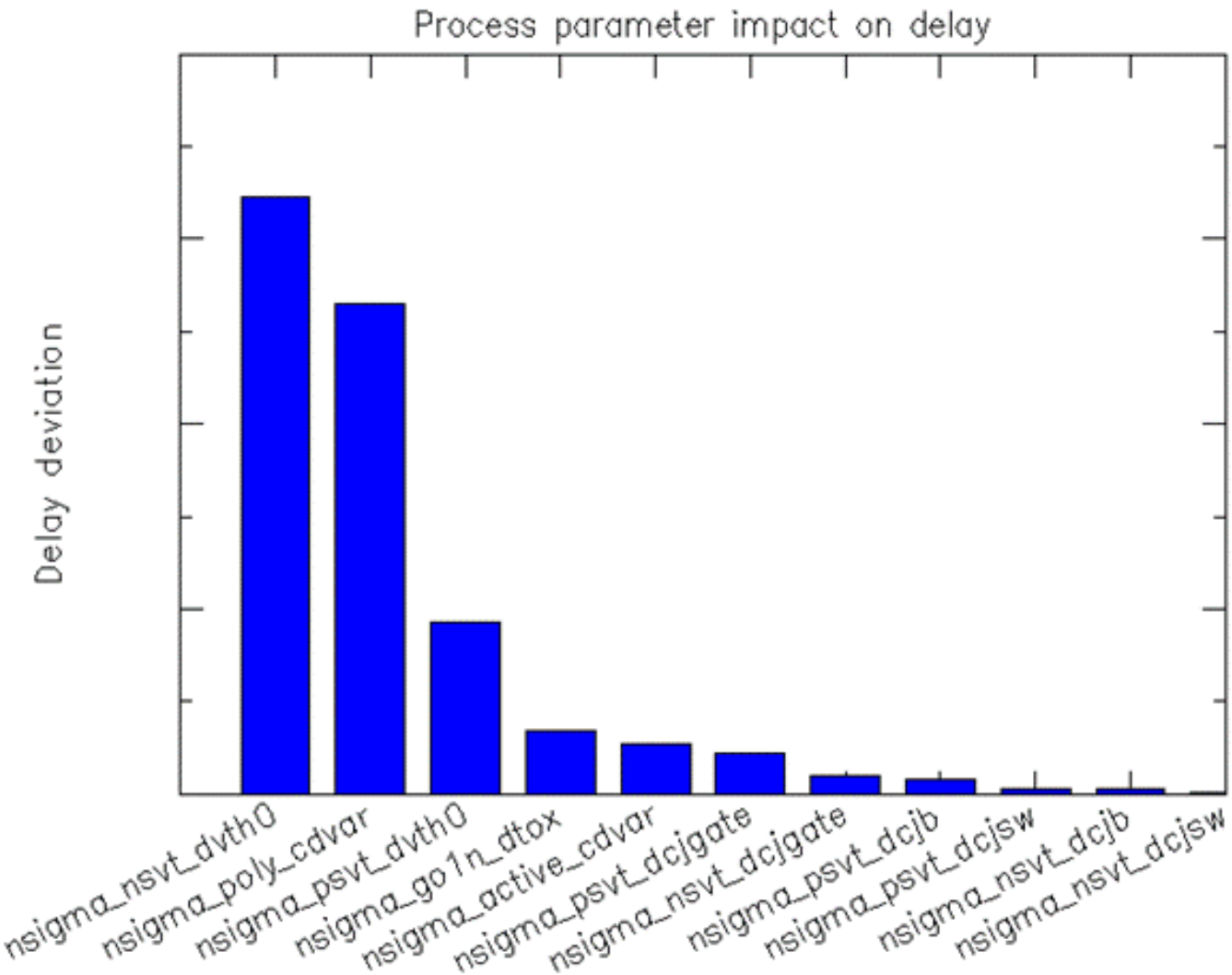
Statistical Library Characterization Flow



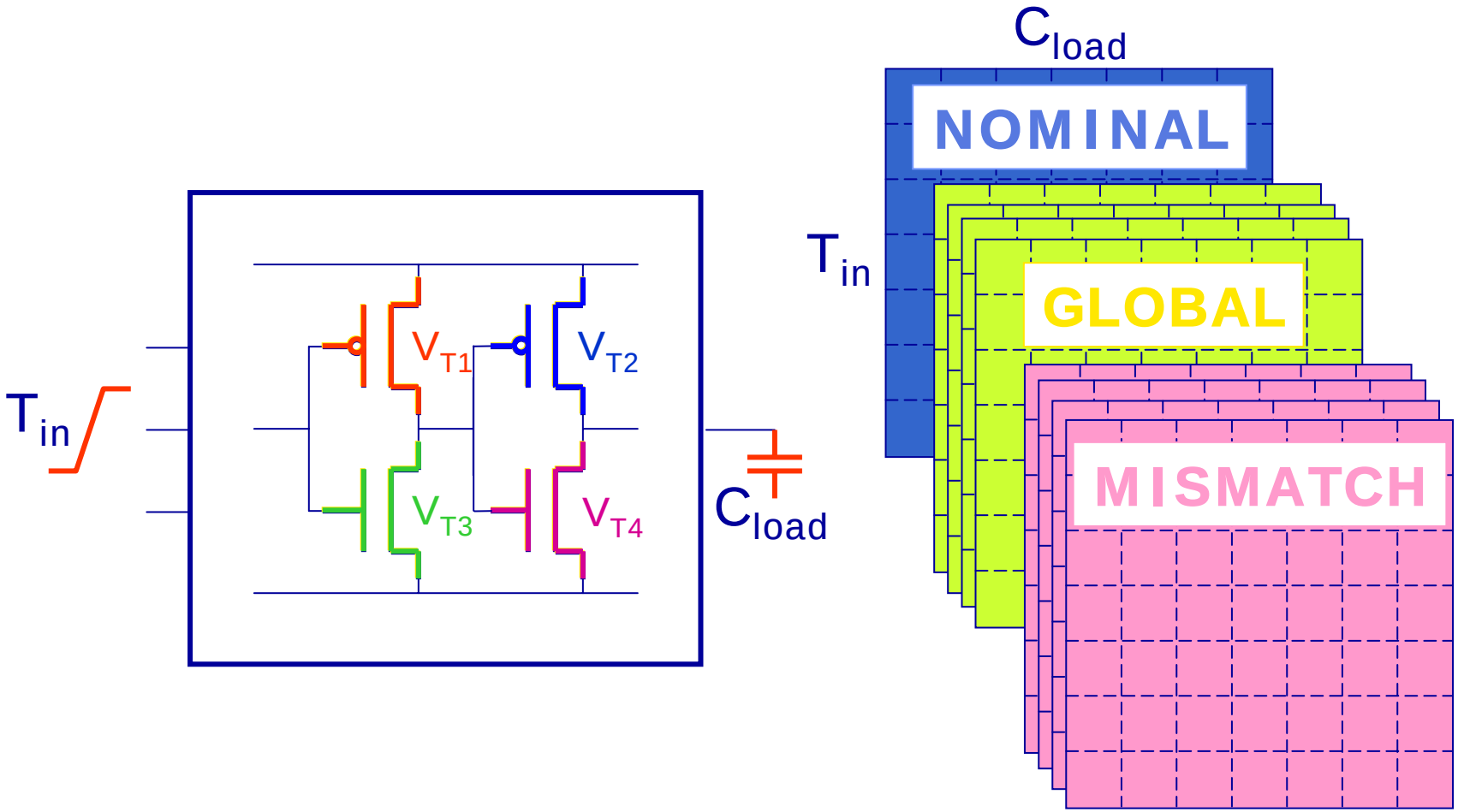
Process Parameters Selection

- Ideally all the process parameters should be considered
- In our SPICE models several parameters are defined
- Among them 16 are impacting the std. cell libraries quantities (timing, capacitance)
- Ideally all the 16 parameters should be used during characterization but with unaffordable runtime and library file size
- The best solution would be to perform a PCA over the process parameters and to derive a subset of uncorrelated components
- Alternatively, only a subset of these parameters can be considered during characterization, still using the corner value for the others

Most Critical Parameter Selection



Global and Mismatch Characterization



Mismatch Characterization Issues

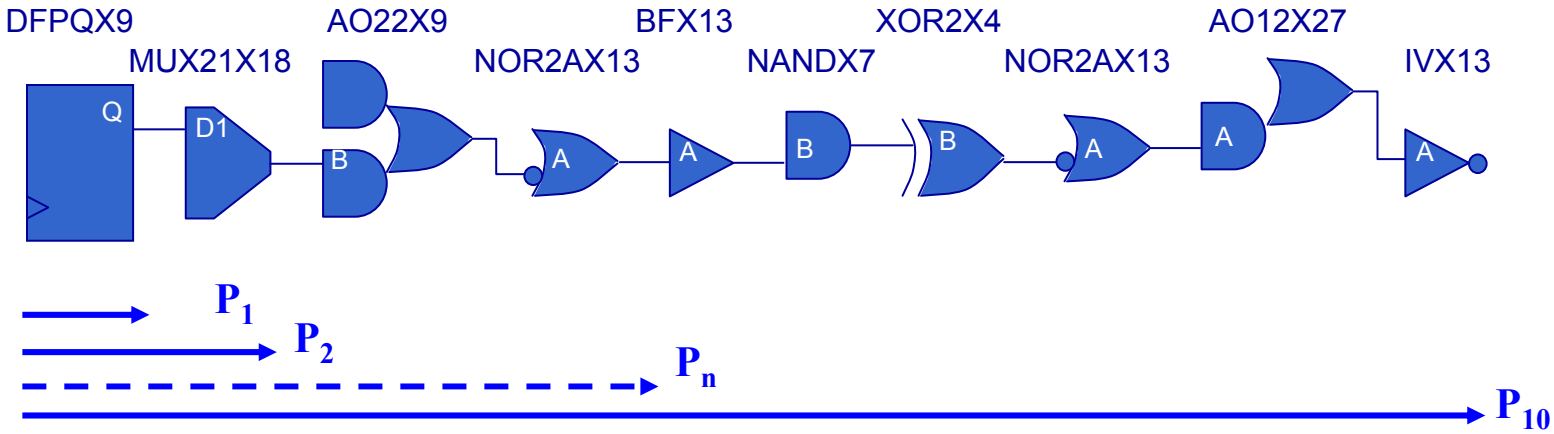
- Ideally every single transistor should be characterized
- Large characterization time
  - Speed-up techniques are needed to reduce the runtime
    - Identification of the transistors impacting the performances
    - Reduction of the characterization grid size
    - …
- Huge library file size
  - Techniques to reduce the number of stored data
    - Mismatch parameters handled at cell level
- A trade-off between accuracy improvement and characterization effort must be identified

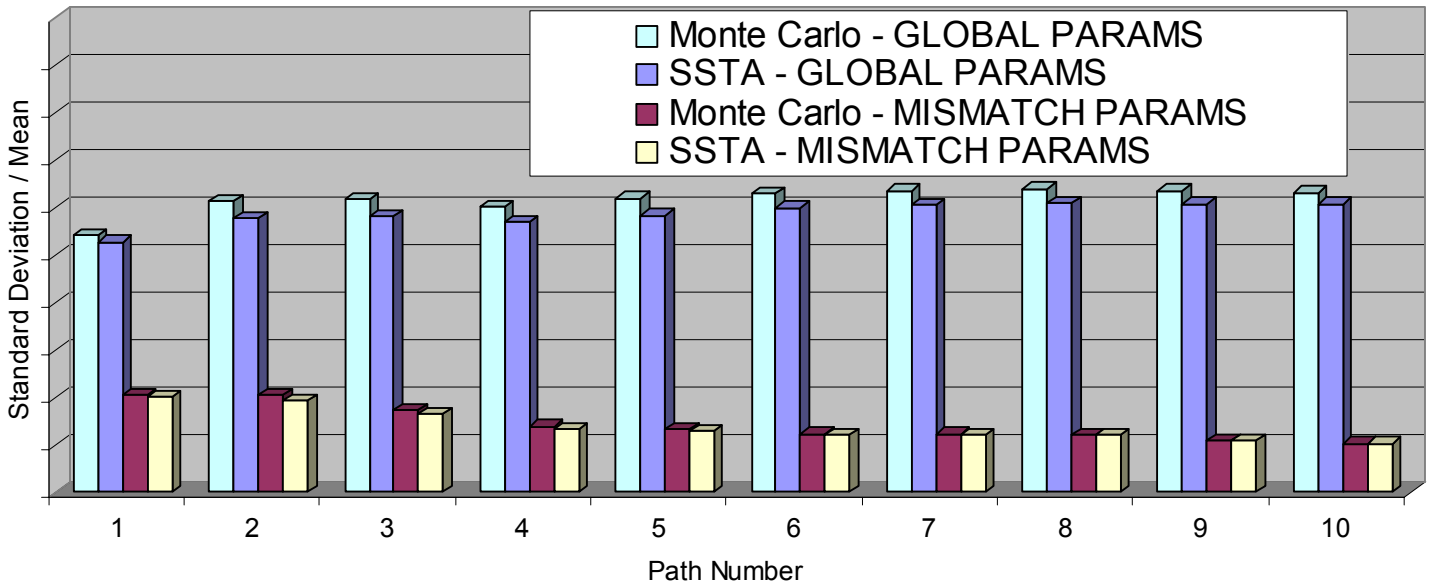
Mismatch Characterization Runtime

- We considered 2 mismatch parameters for both N and P transistors
  - VT_P, $mobility_P$, VT_N, $mobility_N$
- A representative subset of the CORE065LP library cells was characterized wrt mismatch

Library	Total Cells	Seq Cells	Characterization Runtime	
			Combinational arcs	Sequential arcs
CORE065LP	**461**	**37**	**24 hours** (20 hosts)	**3 days** (10 hosts)
CORX065LP	**26**	**17**	**2 hours** (20 hosts)	**2 days** (10 hosts)
fbx65Arm09	**51**	**0**	**5 hours** (20 hosts)	**-**
ST65Flops	**4**	**4**	**8 minutes** (20 hosts)	**5 hours** (20 hours)

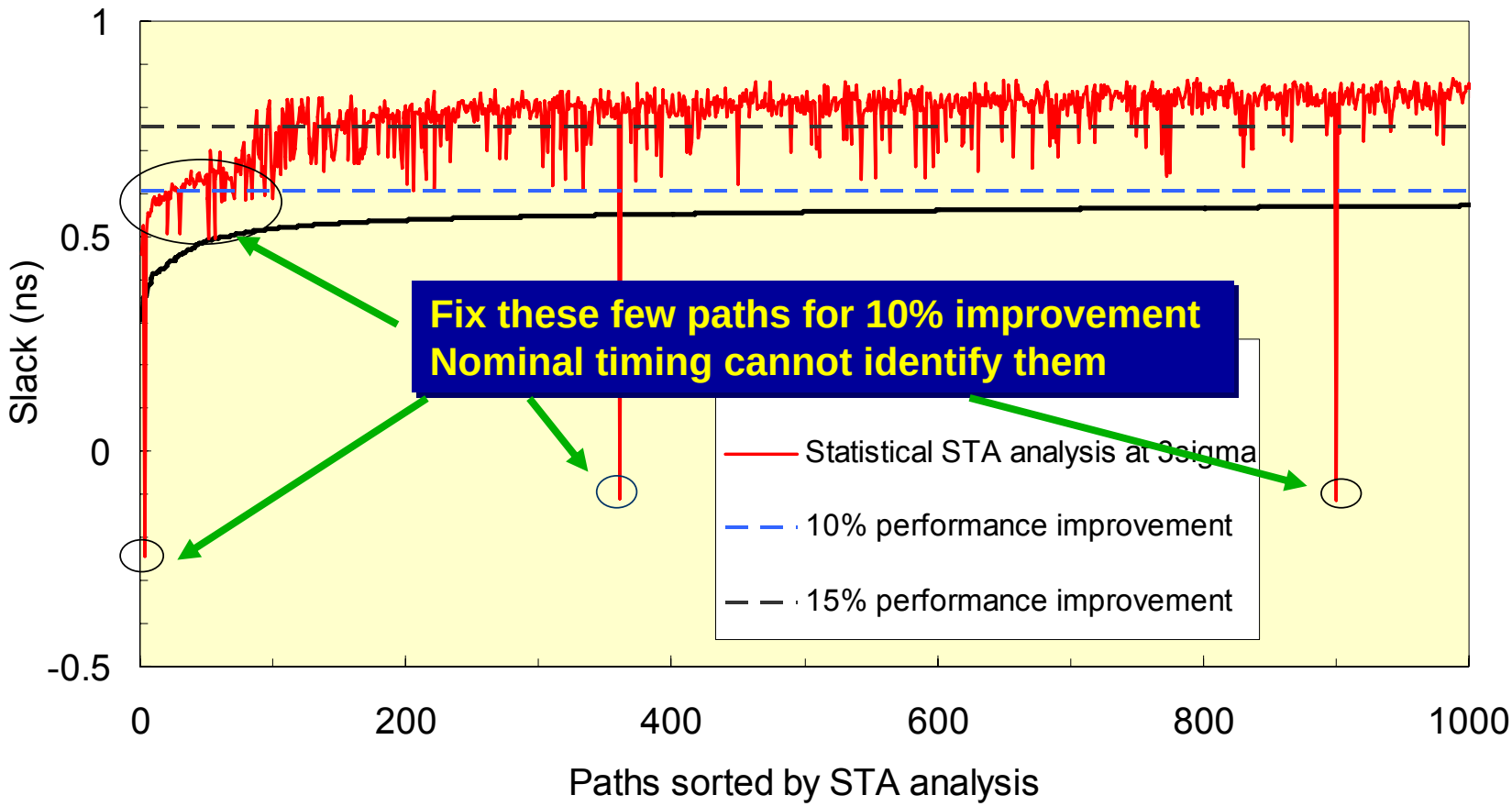
Global vs. Mismatch Variation Impact on Path Delay





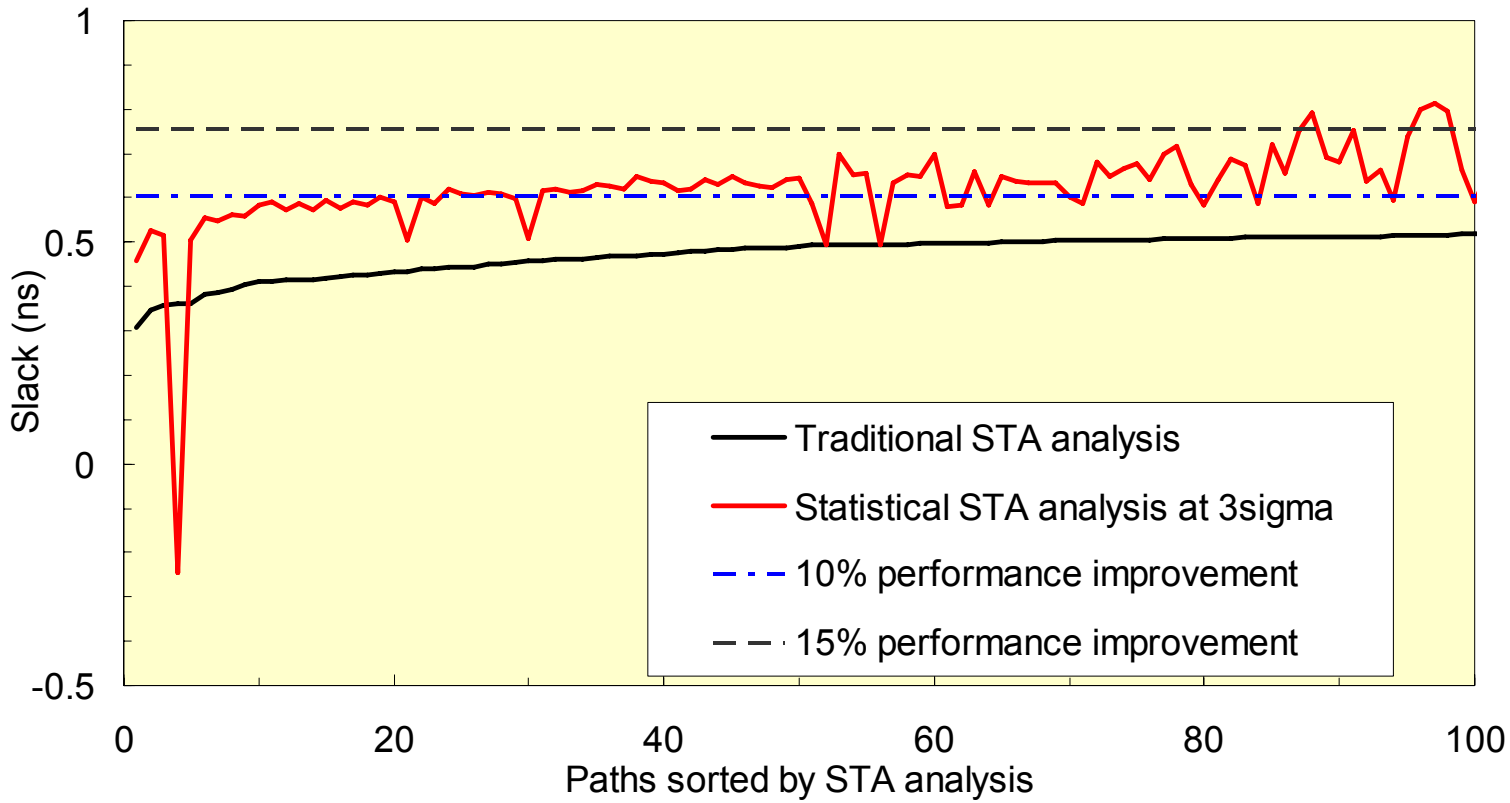
Why SSTA? Increased Performance

- First ST internal pilot project on SSTA

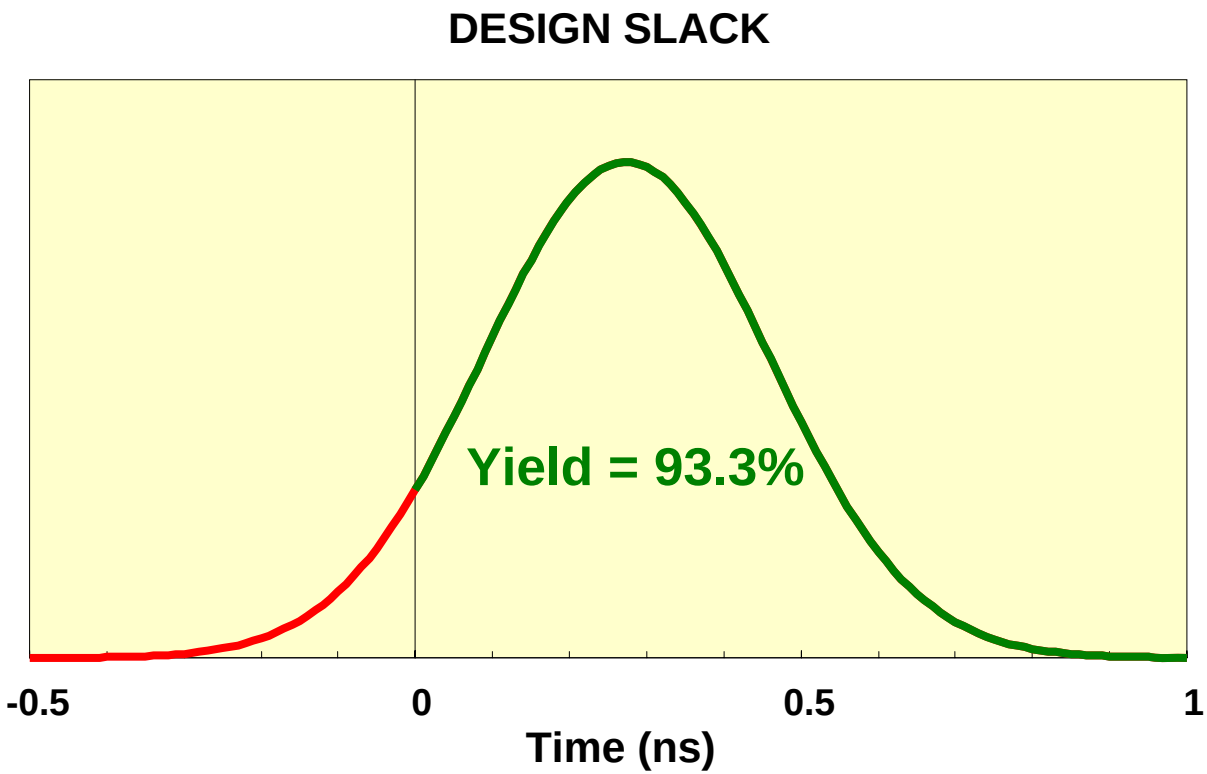


Increased Performance – Top 100 critical paths

- In this design 10% performance improvement seems achievable while 15% may be too hard

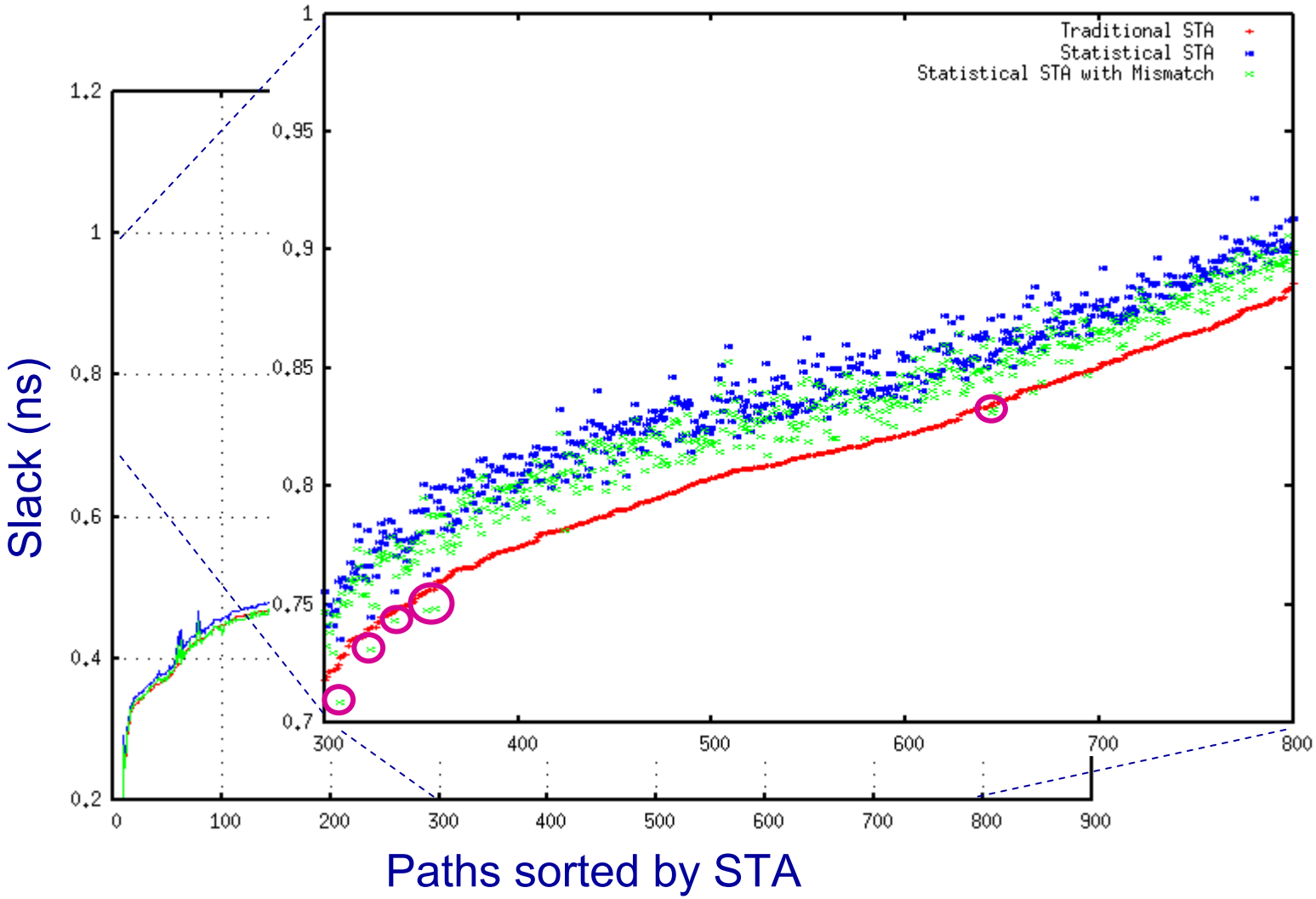


Yield Prediction and Optimization

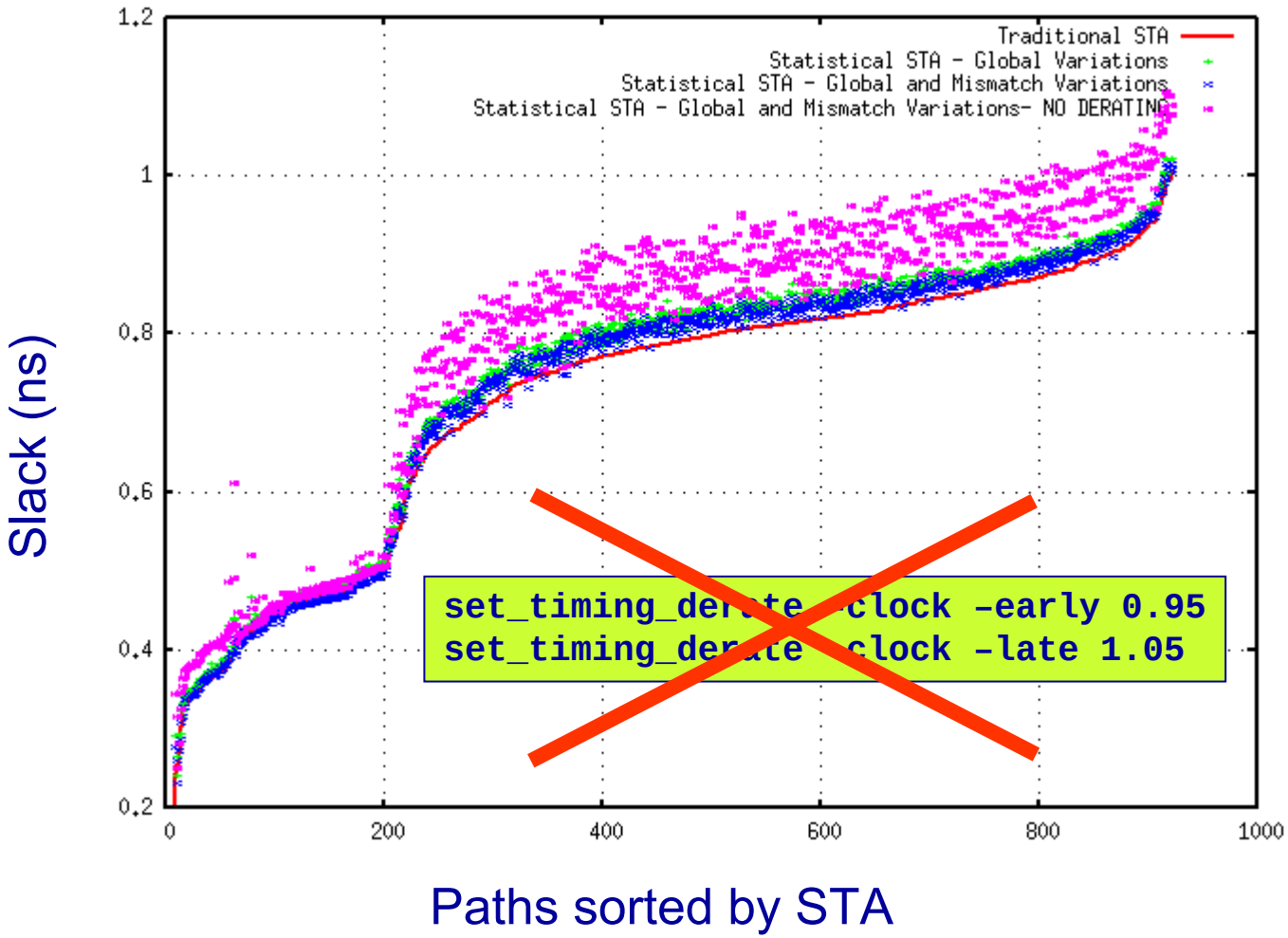


- Before statistical optimization — Yield = 93.3%
- After statistical optimization (~3 hours) — Yield = 95.8%

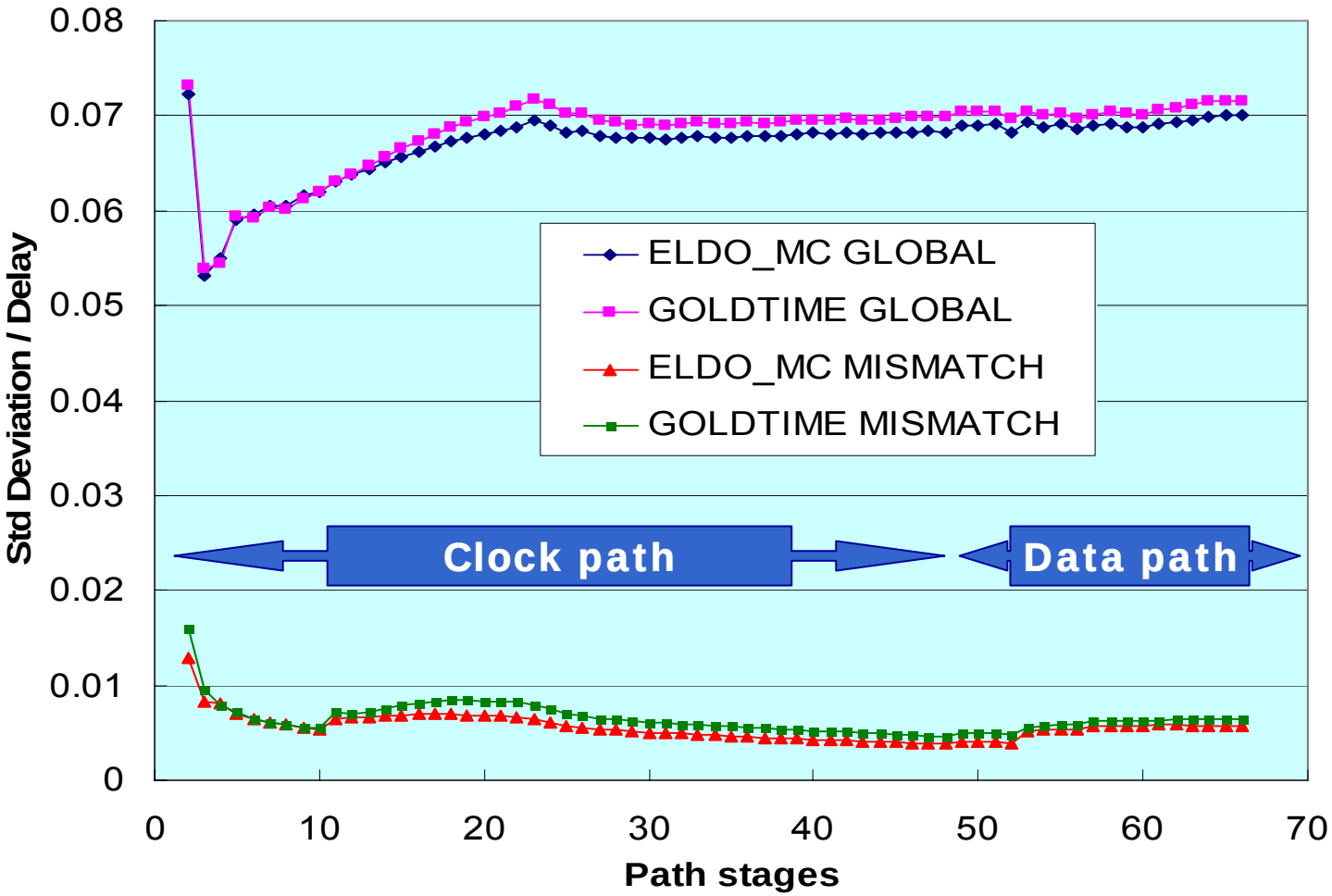
SSTA vs. STA – Results on ILP Design



SSTA vs. STA – Results on ILP Design



ILP Design – Selected Path

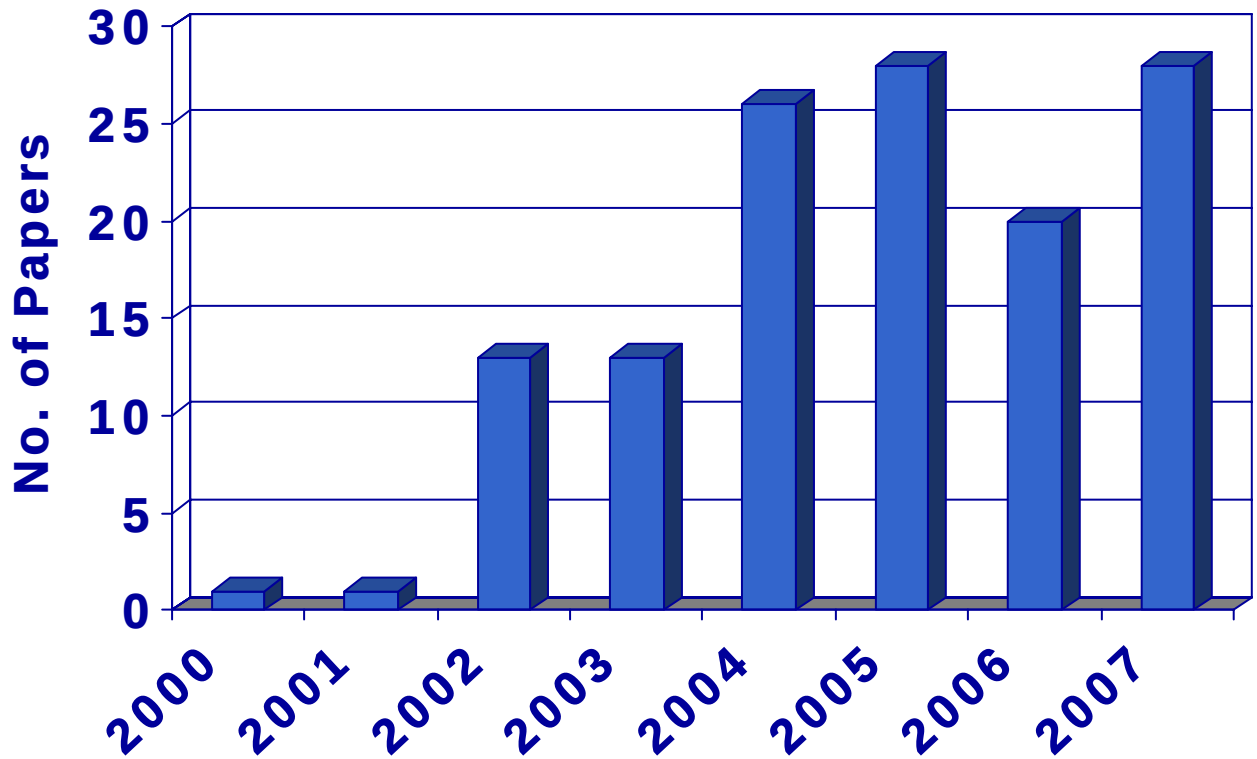


Statistical Analysis of ILP: STA vs. SSTA

- SSTA vs. STA shows different criticality on a few paths
  - The oscillating behavior of SSTA curve means that paths are sorted differently when their slacks are computed with SSTA instead of STA
- SSTA 3σ slacks are generally larger than STA slacks (even when considering the mismatch impact)
  - From these results the gain does not seem impressive but we used the same distributions provided in the CMOS065LP SPICE models
- The results are based on Lot-to-Lot variations
  - Since mismatch is accurately taken into account in SSTA we may not consider the derating typically used for bounding the OCVs (timing constraints should be modified accordingly)

Research Efforts on SSTA

- Papers on SSTA presented at international conferences and workshops on EDA



Research Efforts on SSTA

- Most of the papers presented between 2005 and 2007 propose new approaches to consider the non-linearity of gate and wire delay and the non-linearity of the MAX operation among Gaussian distributions in block-based analysis
- That's fine, but…
- Are we solving the most important problem?
- Is this the real blocking point for the adoption of SSTA in our sign-off flows?
- We believe we should also focus our efforts on developing effective techniques to drastically speed-up the characterization cycle for statistical libraries
- The other critical issue is getting process and silicon characterization data from the fabs

Some "Deterministic" Conclusive Remarks

- The truly atomistic nature of transistors in advanced nanometer technologies will introduce an increasing amount of random variability that cannot simply be controlled at the manufacturing level
- Corner-based design (as intelligent as it gets) will continue to be both pessimistic and risky
  - Excessive guardbanding may not justify the huge investments to scale down to the next technology nodes
- SSTA can help removing pessimism and increasing design robustness
  - The benefits of statistical optimization can be significant
- But without process data, silicon characterization, and correlations we cannot sign off with SSTA
  - Signal integrity should also be considered
- And it is true that we should expect some reluctance from designers (they do not want to change the sign-off methodology)
- But the ever-increasing random variability will continue rising interest and driving efforts in statistical timing analysis and optimization