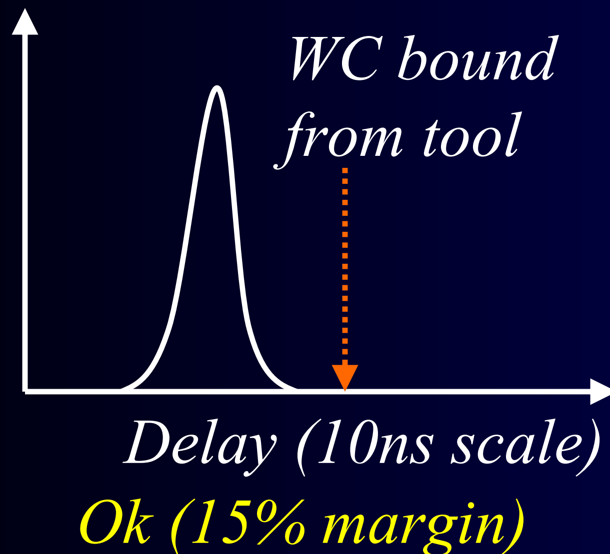
Dealing with timing issues for sub-100nm designs - from modeling to mass production

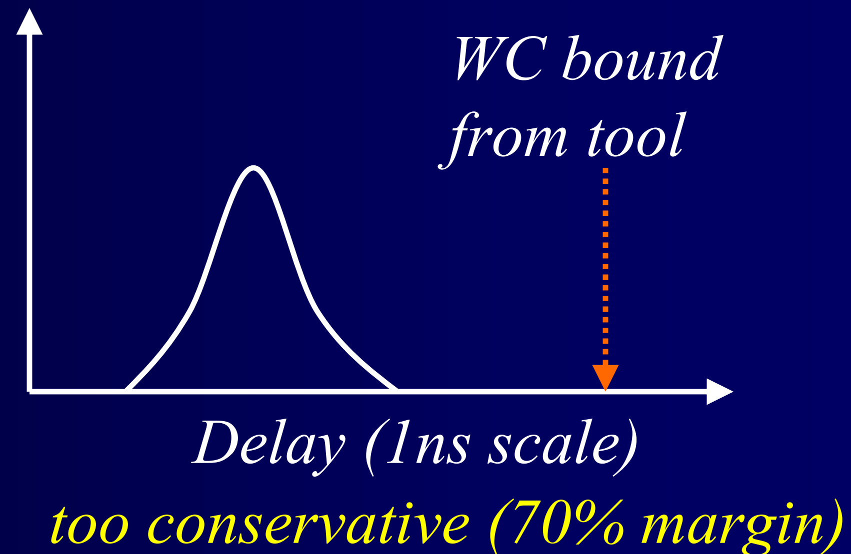
Li-C. Wang and Magdy S. Abadir
University of CA-Santa Barbara and Freescale Semiconductor

What changes in sub-100nm domain?

- Process variations include variability and uncertainties
  - Variability – (predictable) systematic variations
  - Uncertainties – random variations
  - Direct impact on design margin and the rate of yield learning
  - Their relative percentage has increased to the point that *traditional static worst-case (WC) timing analysis becomes too conservative (margin is too big, hard to sign-off)*



Ok (15% margin)



too conservative (70% margin)

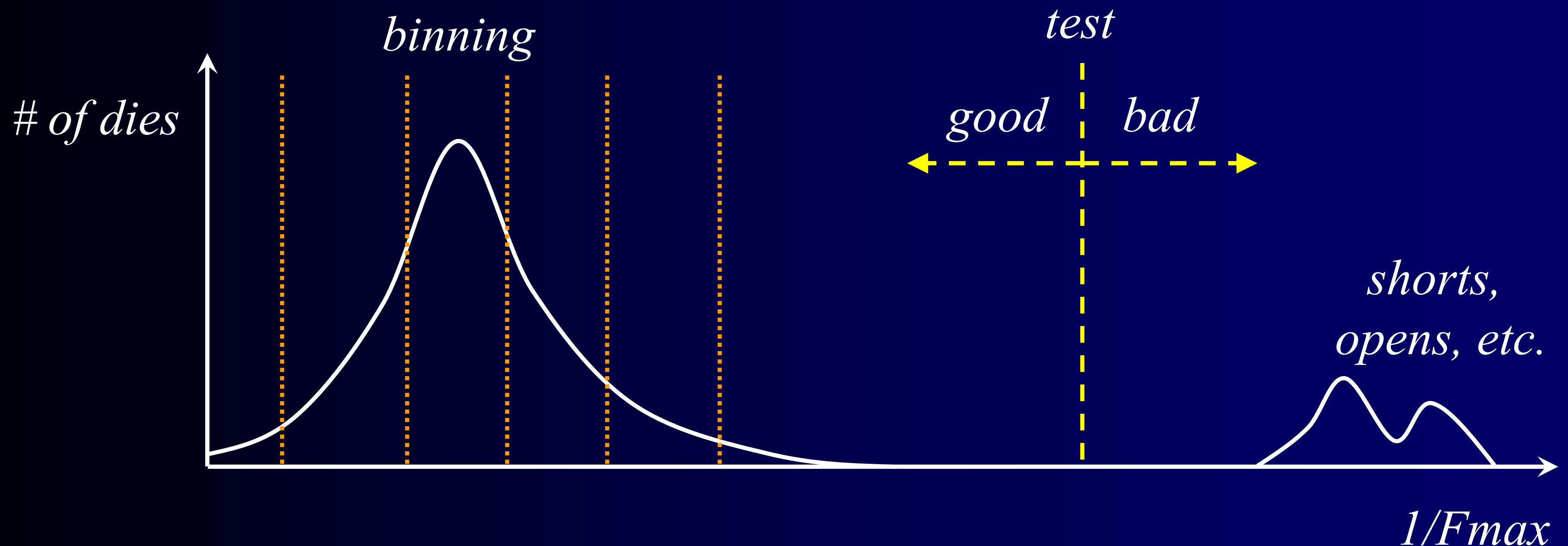
Affect manufacturability

- Increase number of design rules
  - Including *hard rules* and *recommended rules*
  - Rules are to ensure manufacturability
  - Not necessarily for design optimization, validation, debug
  - Allow information flow from manufacturer to design

- Increase importance of silicon debug
  - More issues seen at 1st silicon (design or test issues)
  - Variations/uncertainties can be *design dependent*
  - Require new EDA platform to support efficient design-silicon correlation effort

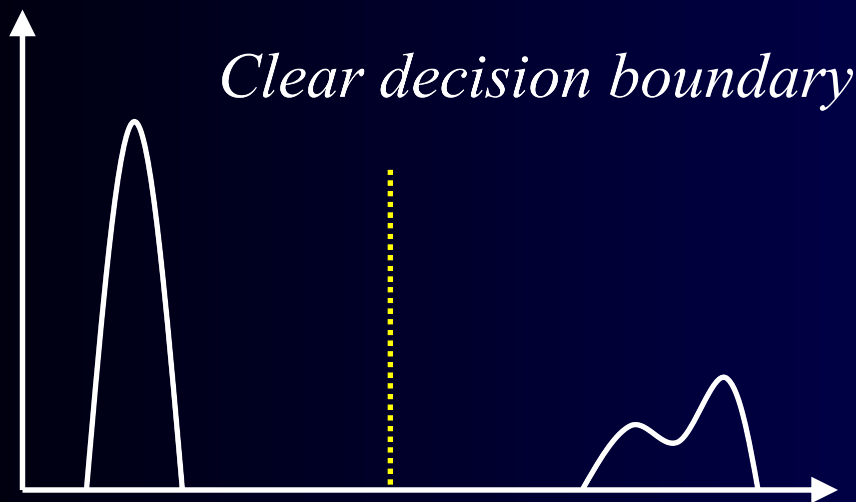
Testing and Binning

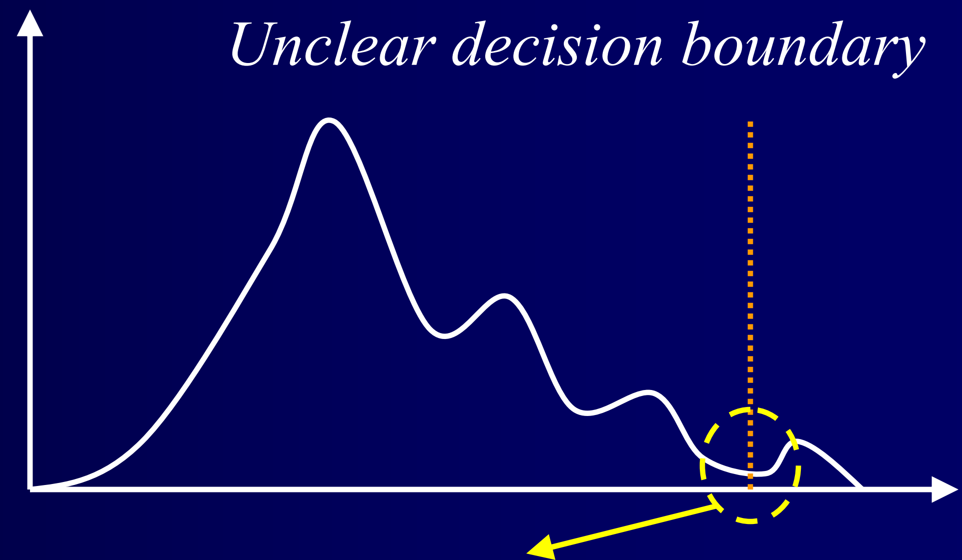
- Defects alter topology
  - Testing is to decide good or bad
- Variations change performance
  - Binning is to decide (performance) range



Deciding between good and bad

- For sub-100nm designs, the decision boundary may no longer be clear
  - ➢ Timing distribution may spread widely
  - ➢ Testing may become more like "binning"
    - ✓ Demand "higher precision" tests for better decision making





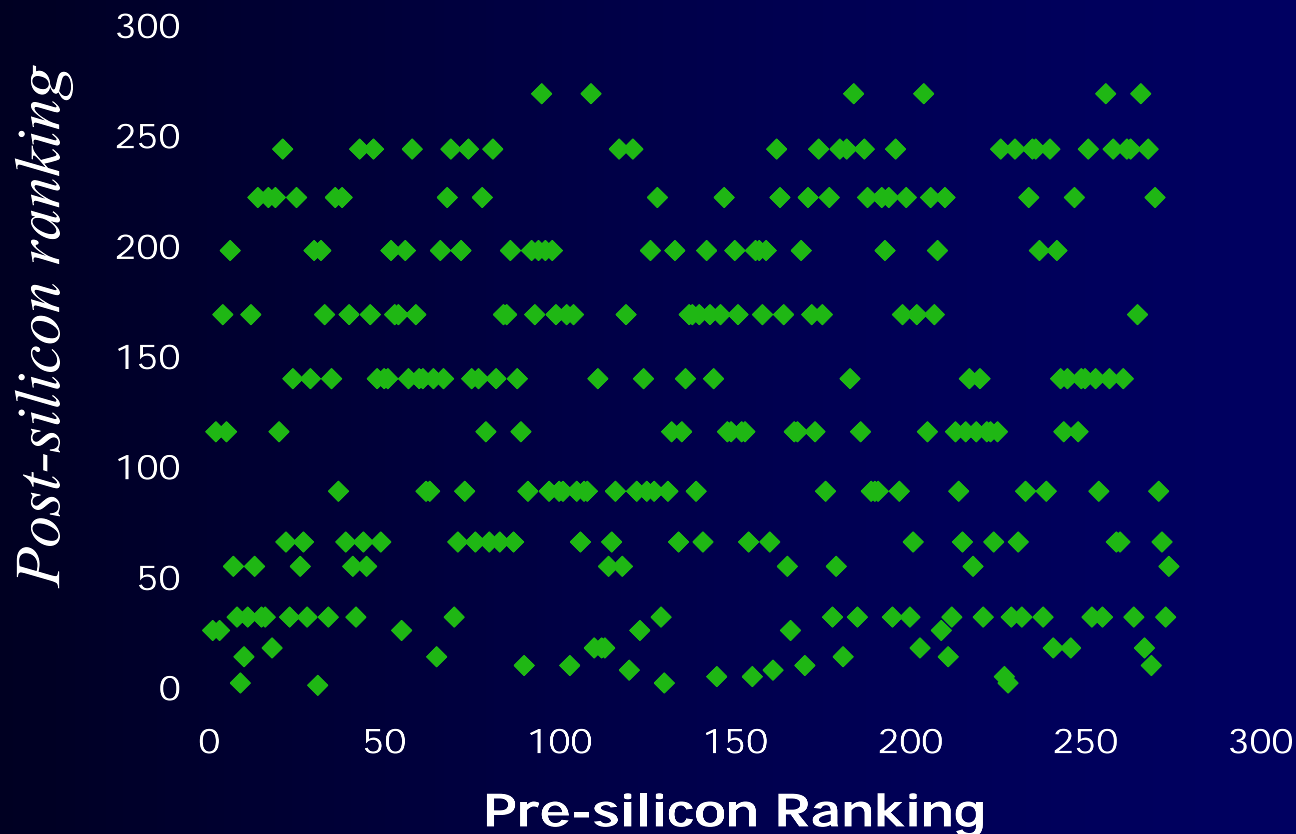
Impact on path timing predictability

- Can timing analysis tool predict speed limiting paths?

- Critical paths Vs. speed paths
  - Critical paths are *predicted* speed paths
  - Critical paths are for design optimization
  - Many second-order timing effects are not accounted for in traditional timing analysis

Impact on path timing predictability

- Timing-analysis-reported critical paths do not correlate to silicon-observed speed paths
  - Correlation = 0.05 (we wish this to be 0.99!)



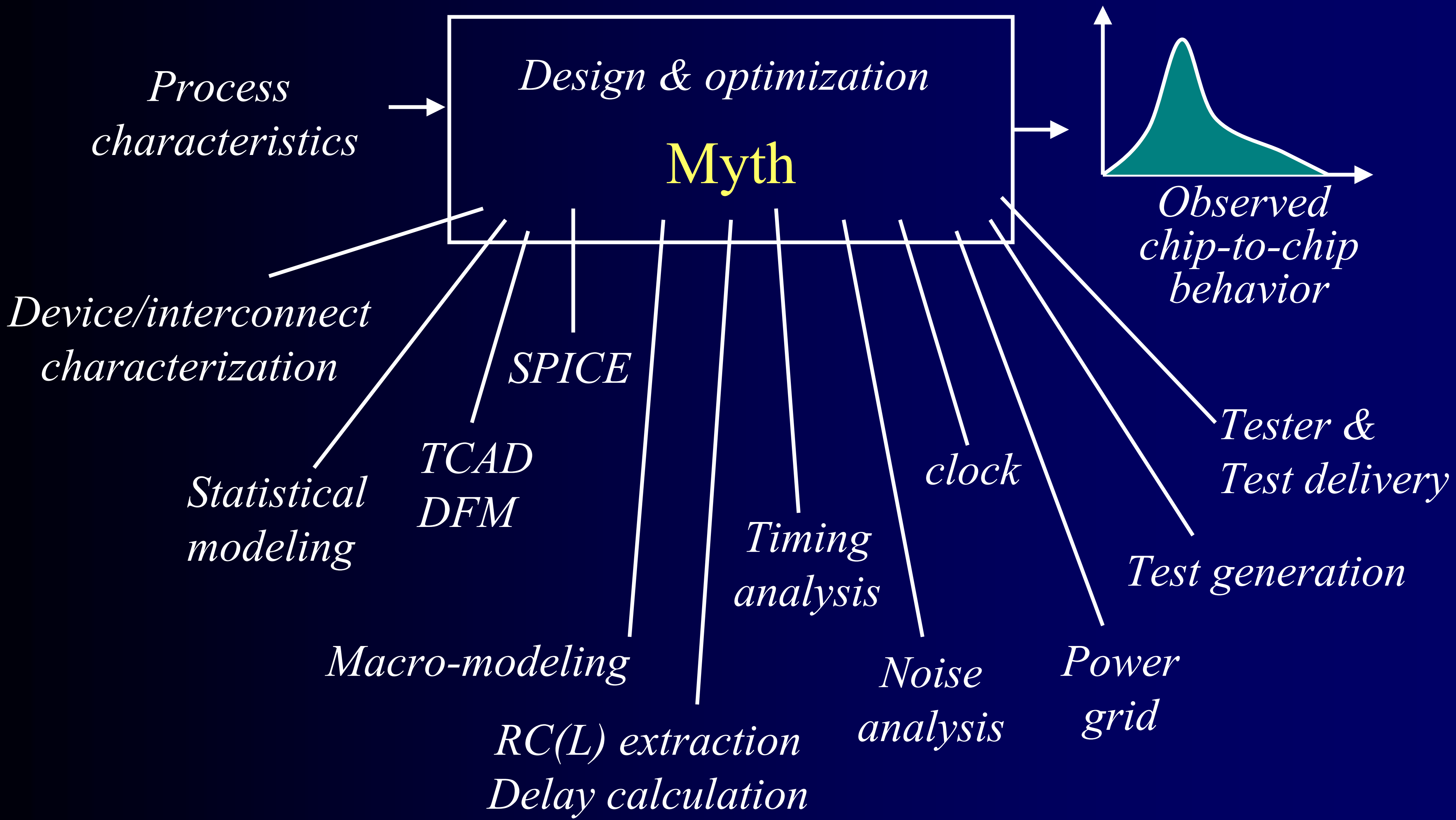
Doubts

- Can I still trust the static timer for design optimization?
- Can I use a timing analysis tool to predict speed limiting paths?
- If I have tests for critical paths, are they sufficient to expose all the worst-case timing corners?
- Is statistical timing analysis the ultimate answer I am looking for?

Commonly-asked questions

- What cause a speed path to be missed by timing analysis tools?
  - Why is a speed path not a critical path?
  - Is it sufficient to test only critical paths (not speed paths)?
- How variations should be modeled in order to support timing analysis?
  - How to build an effective statistical timing model?
- Where do the variation models come from?
  - What do we need from a foundry for building a reliable statistical timing model?
- What are the important variations to be considered in analyzing timing?
  - Which is the dominating variation factor(s)? Leff or Vth variation?

Chip variability is a result of many things



Things that affect timing

- Factors
  - Device characteristics (Vth, Ids, etc.)
  - Interconnect characteristics (RCL)
  - Coupling
  - IR drop, power noise
  - Temperature
  - Clock skew
  - Modeling errors
  - Approximation/optimization algorithms
  - And so on …

- Variability and uncertainties
  - Process variations (including measurement uncertainties)
  - Environmental variations (temperature map, power map, etc.)
  - Variation in test patterns

Dealing with chip variability

- Result of interactions among
  - Process variability and uncertainties
  - Design variability
  - Modeling uncertainties
  - Variability in assumptions employed in tools for fast approximation
  - Variability and uncertainties in test and measurements

- Divide-and-conquer (design): To cope with chip variability, we need to decompose the sources of variations and minimize their interactions
  - To analyze and control variations separately

- Statistical learning (test): To overcome chip variability, we need a way to treat the complex interactions as a black box
  - To learn the lumped timing behavior through testing of silicon samples

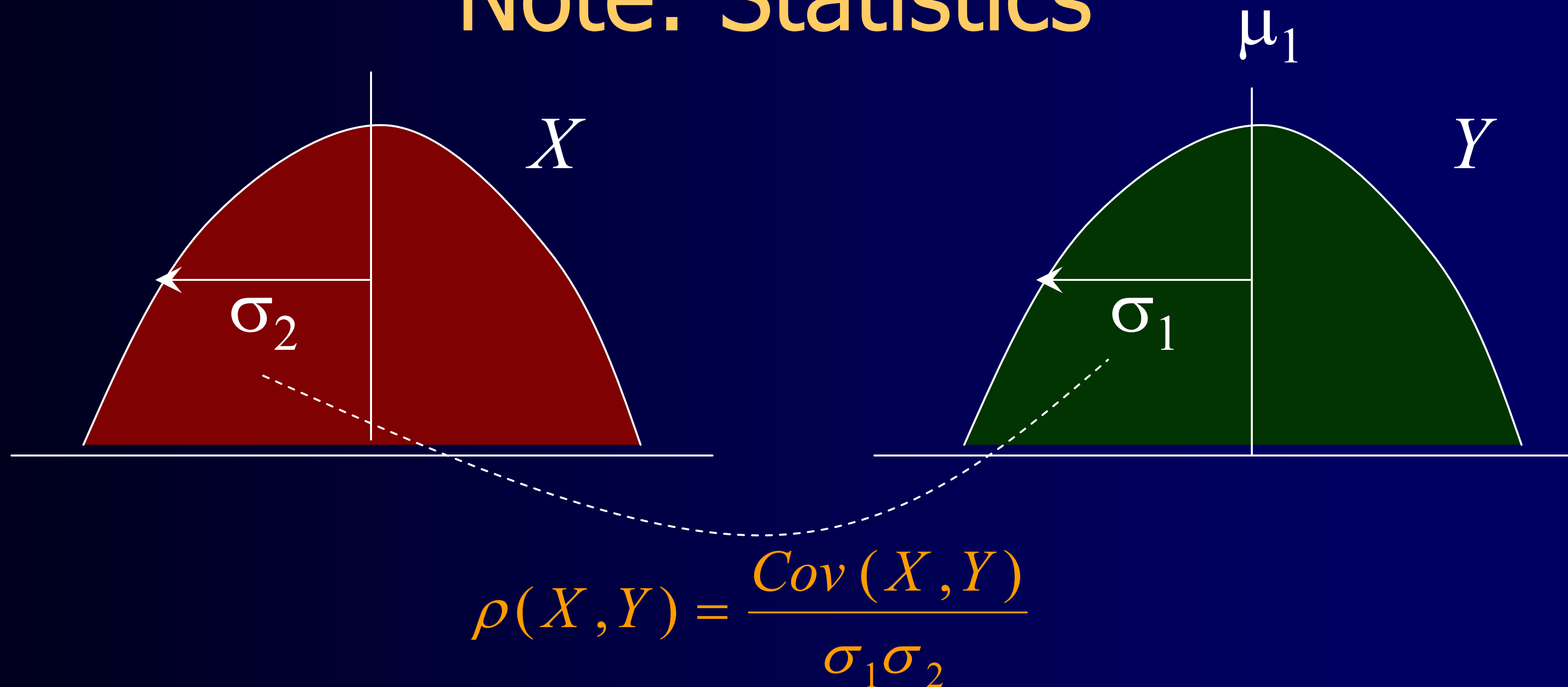
WC analysis or nominal analysis?

- Worst-case analysis
  - Assume worst cases in models and in tool inaccuracy
  - Sum up worst cases in the analysis
    - ✓ Divide and conquer by assuming budgets and margins
  - Variability and uncertainty cause large margins to be left out, which make it difficult for design closure
- Nominal analysis
  - Modeling and analyzing nominal behavior
  - Variability and uncertainty is resolved in testing stage
  - A sophisticated binning methodology may be required
    - ✓ May involve tremendous silicon debug effort
  - High-performance designs take this route already

This tutorial

- Studies some of the myth mentioned above
  - Process guys, TCAD people, circuit designers, EDA engineers, and test people often have different perspectives to the "variation problem"
  - This tutorial intends to examine all perspectives in one place
- Discuss issues from process characterization to silicon speed binning
- Investigate problems from a statistical perspective

Note: Statistics



$$\rho(X,Y) = \frac{Cov(X,Y)}{\sigma_1 \sigma_2}$$

- Usually, when we talk about "statistical analysis," we are interested in
  - Mean of a random variable
  - Standard deviation (sigma) of a random variable
  - Correlation among a set of random variables
- μ/σ can be seen as the percentage of variability
- $[\mu\text{-}k\sigma, \mu\text{+}k\sigma]$ can be seen as the bounds of variability

Topics to cover

- Basics (3.5 hours)
  - Introduction
  - Process characterization and modeling of variations
  - Macro-modeling and timing analysis
  - Statistical timing analysis
- Advances (2.5 hours)
  - Simplified SSTA and pattern-based STA
  - DSM timing effects
  - Studies of speed binning
- Optional (30-40 minutes)
  - Advanced methods based on statistical learning (only if time allows)

So, everything in design begins with SPICE, let's understand how SPICE parameters are extracted

Main question: Can I expect an accurate statistical SPICE model in the near future to support my statistical timing analysis?

Semiconductor Metrology

- Metrology is defined as the measurements of various parameters

- $Cp = (USL - LSL) / 6\ \sigma_{process}$
  - USL : upper process SPEC limit
  - LSL : lower process SPEC limit

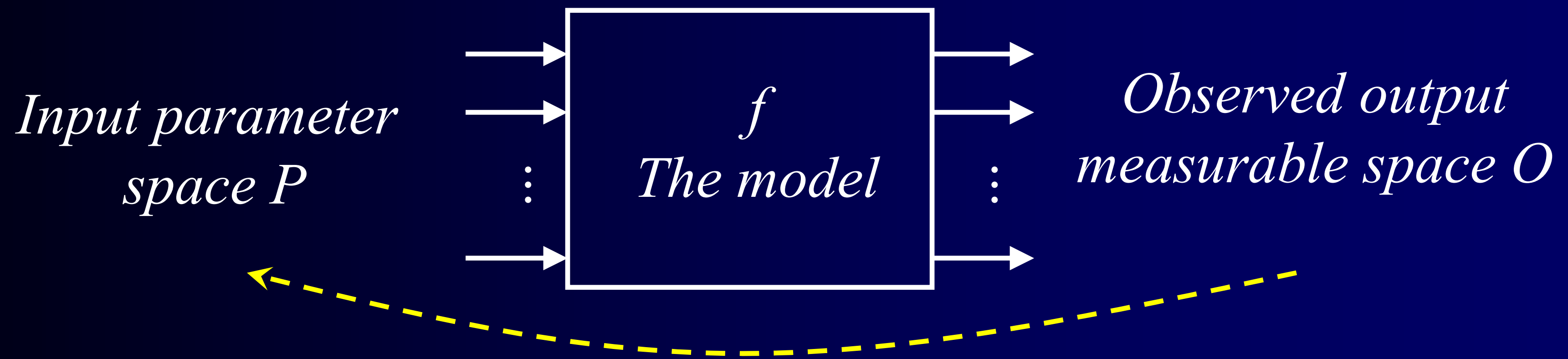
- $P/T = (6\ \sigma_{measurement}) / (USL - LSL)$
  - P : measurement precision
  - T : process tolerance
  - Used to evaluate the ability of an automated metrology tool

- Typically, P/T should be less than 10%, although 30% is usually allowed

For example

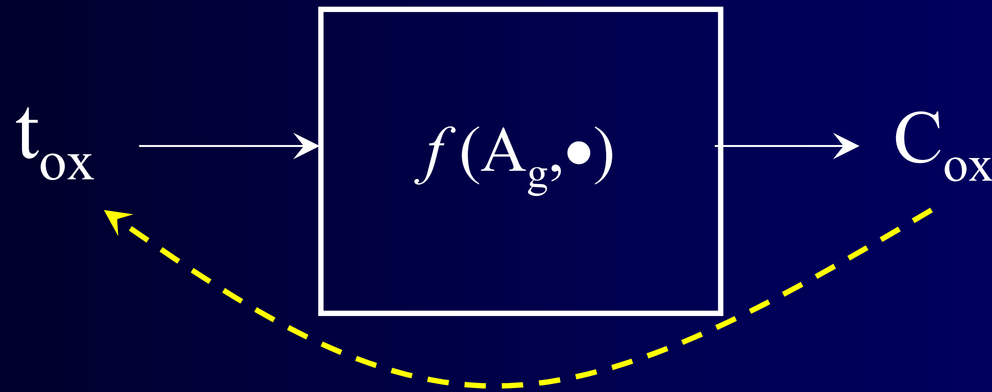
- Measure the thickness of the transistor gate dielectric at 100nm technology generation
  - Suppose the gate is 2nm thick
  - Process tolerance is $\pm$ 5% = 0.1nm
- P/T = 10% = (6 $\sigma_{measurement}$) / 0.1nm
  - $\sigma_{measurement}$ = 0.0017nm
  - An atomic step on silicon is about 0.15nm!
- Direct measurement on some process parameters can be difficult

Model-based measurement



- Each measurement method is based on a model f that relates observed signals to the values of variables being measured
- Model-based measurement alleviates the high precision requirement for measuring some process parameters directly
- Depending on the model and the algorithm used to extract values from the observed signals, various degrees of error can be introduced

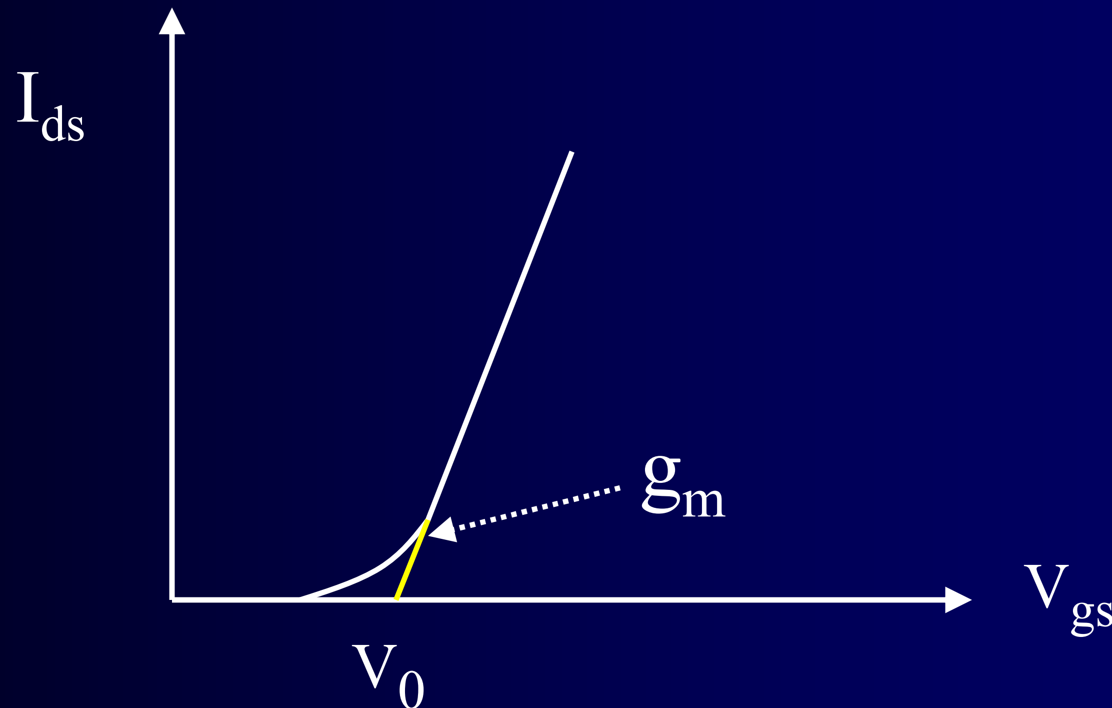
MOS parameter extraction – an example



- Consider determining the gate oxide thickness t_{ox}
- From conventional Capacitance-Voltage measurement
  - We use the simple formula $C_{ox} = (\varepsilon_{ox} / t_{ox})\, A_g$
  - C_{ox} : measured capacitance
  - ε_{ox} : usually 3.9 ε_0
  - ε_0 : permittivity of free space $8.854\text{X}10^{-14}$
  - A_g : device gate area (build a large gate)

Extraction of threshold voltage V_{th}

- Use the formula in linear region
  - $I_{ds} = (\mu\, C_{ox}\, W / L)\, (V_{gs} - V_{th} - 0.5\, V_{ds})\, V_{ds}$
  - g_m (transconductance) $= \partial I_{ds}/\partial V_{gs} = (\mu\, C_{ox}\, W / L)\, V_{ds}$
  - Set $I_{ds} = 0$, obtain $V_0 = V_{th} - 0.5\, V_{ds}$
  - So, $V_{th} = V_0 + 0.5\, V_{ds}$



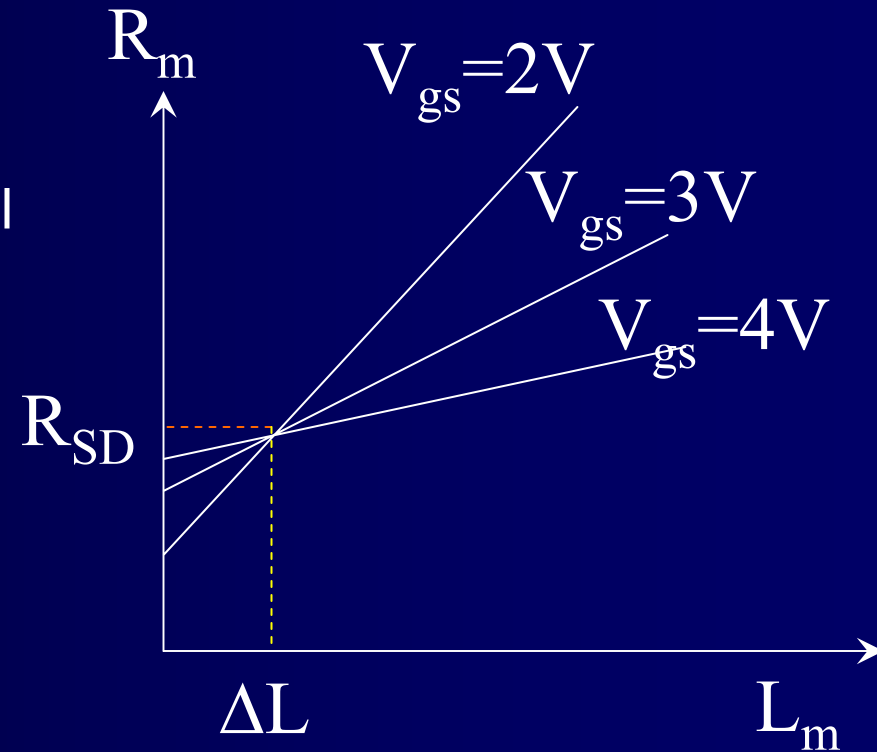
Effective mobility μ

- I_{ds} is measured in the linear region $V_{gs} > V_{th}$
  - At low V_{ds} (< 0.1V)
- Use the formula
  - $g_m = \partial I_{ds}/\partial V_{gs} = (\mu\ C_{ox}\ W / L)\ V_{ds}$ = the slope
  - $\mu = (g_m\ L) / (C_{ox}\ W\ V_{ds})$
- Or another method is to calculate
  - $g_{ds} = \Delta I_{ds} / \Delta V_{ds}$ at each V_{gs}
  - $\mu_{eff} = (g_{ds}\ L) / (C_{ox}\ W\ (V_{ds} - Vth))$
  - μ_{eff} significantly drops near $V_{gs} = V_{th}$

Channel length

- $L = L_m - \Delta L$
  - L_m : drawn channel length
  - ΔL : difference between drawn and actual
  - The objective is to measure ΔL

- Measuring ΔL is more complicated
  - Use channel resistance method (R_m), by
  - $R_m = A (L_m - \Delta L) + R_{SD}$
  - Calculating $A = 1 / (\mu C_{ox} W (V_{gs} - V_{th}))$
  - At various V_{gs} values
  - Intersect different lines in R_m Vs. L_m plot
  - Use intersected point to obtain ΔL

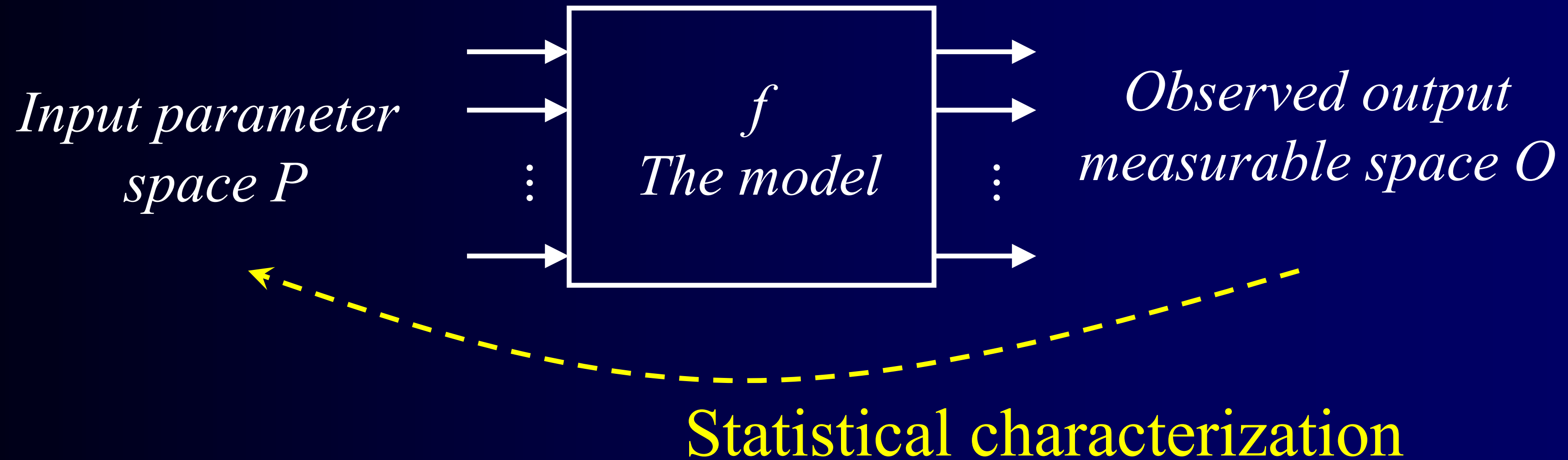


- See *Handbook of Silicon Semiconductor Metrology*

To summarize …

- MOSFET device model
  - $I_{ds} = 0$ for $V_{gs} - V_{th} < 0$
  - $I_{ds} = (\mu\, C_{ox}\, W / (L - \Delta L)\,)\,(V_{gs} - V_{th} - 0.5\, V_{ds})\, V_{ds}$
  - $I_{ds} = (\mu\, C_{ox}\, W / 2(L - \Delta L)\,)\,(V_{gs} - V_{th})^2$ (saturation region)
- Parameter space $P = \{W, \Delta L, V_{th}, \mu, C_{ox}\}$
  - They may not be directly measurable
  - They are to be inferred from measurements of "electrical properties" I_{ds}, V_{gs}, and V_{ds}

Statistical characterization of *P*



- If we treat each variable in P as a random variable, we need to estimate their means and sigmas
- These random variables in P can be correlated!
- This increases the difficulty of measurement

Extracting statistics by ignoring correlations

- We can measure many devices individually for extract the statistics of a single parameter
- Because the measurement error ε_p is unknown, the statistics of P can become questionable
- Moreover, a complex model such as BSIM-3 have hundreds of parameters, many of which are hard to extract by measuring capacitance, current, voltage
- These increase the difficulty of variation extraction

Parameters are correlated (Boning & Nassif 99)

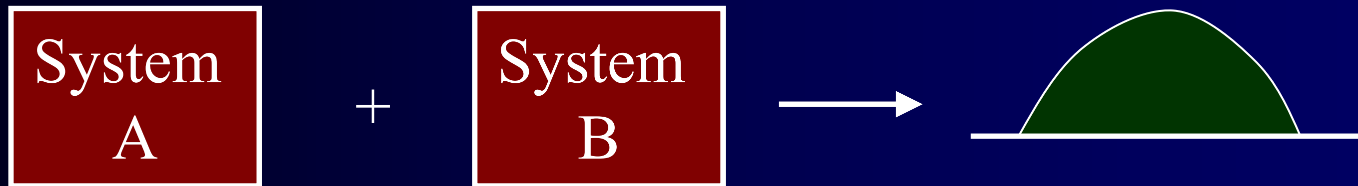
- Consider $P = \{ V_{th}, \beta, \theta \}$
  - $\beta = \mu C_{ox} W / 2(L - \Delta L)$
  - θ : is a new parameter to model mobility roll-off with vertical field
- Use the formula
  - $I_{ds} = \beta (V_{gs} - V_{th}) V_{ds} / (1 + \theta (V_{gs} - V_{th}))$
  - Measure on 476 MOSFETs
  - Obtain correlation structure as the following

	β	θ	ε_p
V_{th}	-0.897	-0.780	-0.207
β		0.914	0.328
θ			0.329

Systematic error

- The error ε_p is not independent from the parameters
  - The parameters are eventually used to characterize the performance of a device
  - The error ε_p will be propagated into error in this performance characterization
- The fact that error ε_p is not independent from the parameter increase the error in performance characterization

Intuition: systematic variability increases variance

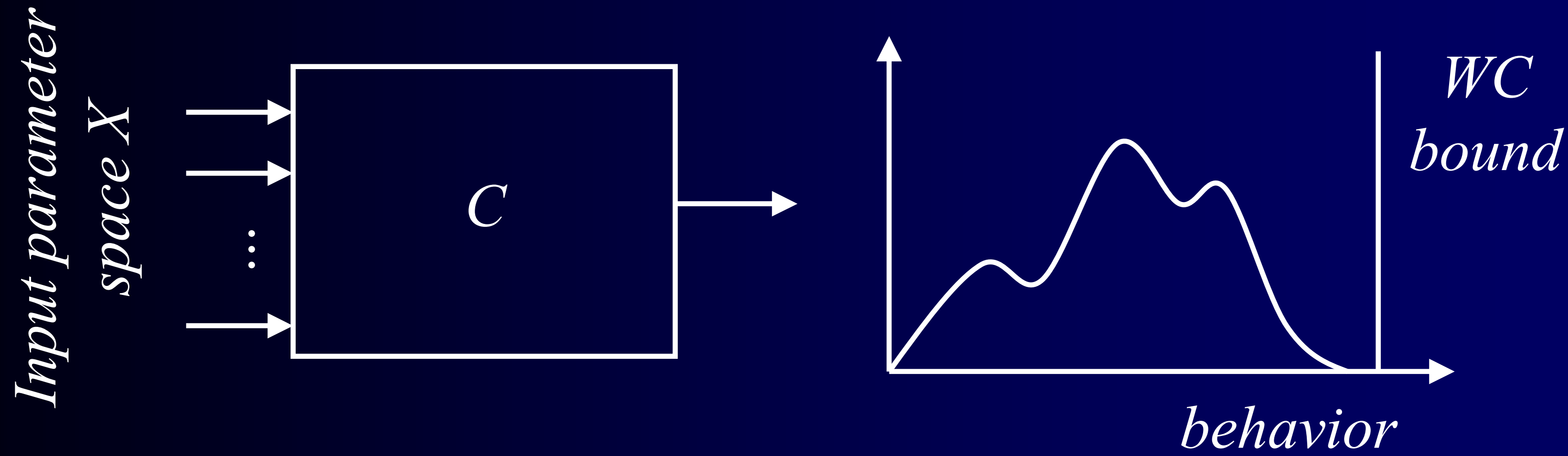


- Assume two Gaussian distributions
  - $A = N(\mu_1,\sigma_1)$, $B = N(\mu_2,\sigma_2)$
- Assume $f = A + B$
- When A, B are totally independent
  - $\sigma(f) = (\sigma_1^2 + \sigma_2^2)^{1/2}$
- When A, B are 100% correlated
  - $\sigma(f) = \sigma_1 + \sigma_2$
- See that correlation increases the sigma of the output behavior f

In summary

- It is usually difficult to estimate the correlation structure among parameters
- As a result, we may have
  - The parameter statistics are not updated often to reflect the maturity of the Fab process
  - There is a strong demand to develop characterization methods that are less sensitive to the correlations among parameters
  - That is why we rely on worst-case analysis

Worst-case corner analysis



- Given statistical variations in input parameter space X, compute *a bound* for the worst-case behavior of interest

Worst-case characterization

- Worst-case characterization is not easy
  - It still require to sample the distribution
  - But the result is less sensitive to the distribution change once it is fully characterized

- Each type of performance metric may result in a set of worst-case parameter values
  - eg. delay, power, noise immunity, etc.
  - Result in worst-case corner analysis
  - A simple model for an ASIC cell may use one unique set of values for all types of performance

- Characterization is done for each type of devices or structures

Summary

- Statistical characterization can be expensive
- Correlations among parameters are crucial but may not be easily obtainable
- Worst-case characterization is less sensitive to process shift

- Can I expect an accurate statistical SPICE model in the near future to support my statistical timing analysis?
  - You can draw your own conclusion ...

Process variations and modeling of process variations

Break for question

Variations

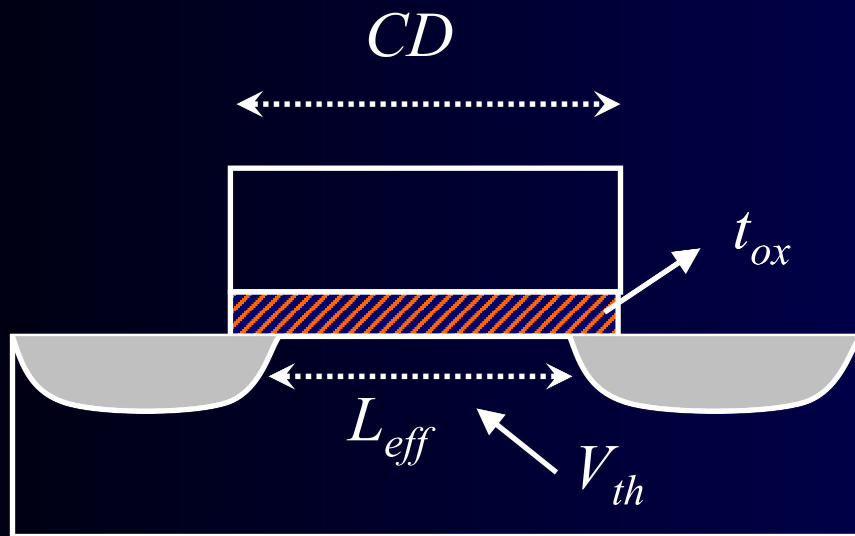
- Temporal Vs. Spatial
  - Temporal : concern equipment drift over time
  - Spatial : non-uniformity across wafer or die
- Inter-die (die-to-die) Vs. intra-die (within-die)
  - Inter-die : same location across dies (wafer level)
    - Lumped statistics of fab-to-fab, lot-to-lot, wafer-to-wafer, and die-to-die variations
  - Intra-die : different locations on a die (die level)
- Systematic Vs. random
  - Systematic : exist correlation structures among random variables; certain trends exist
  - Random : no correlation among random variables

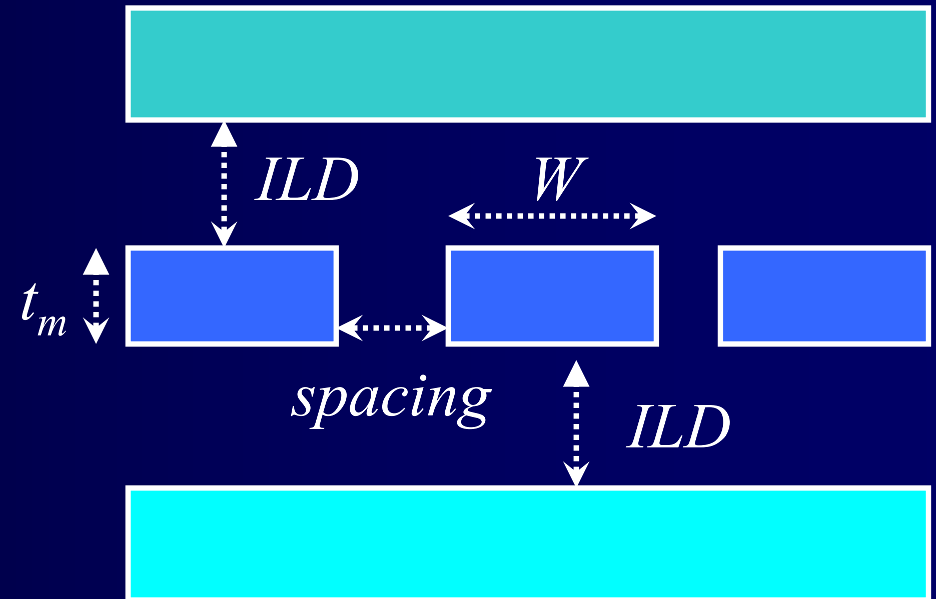
Process variations (Boning & Nassif 99)

- Process variations can be classified as
  - Variation in geometry
  - Variation in material
  - Variation in electrical property

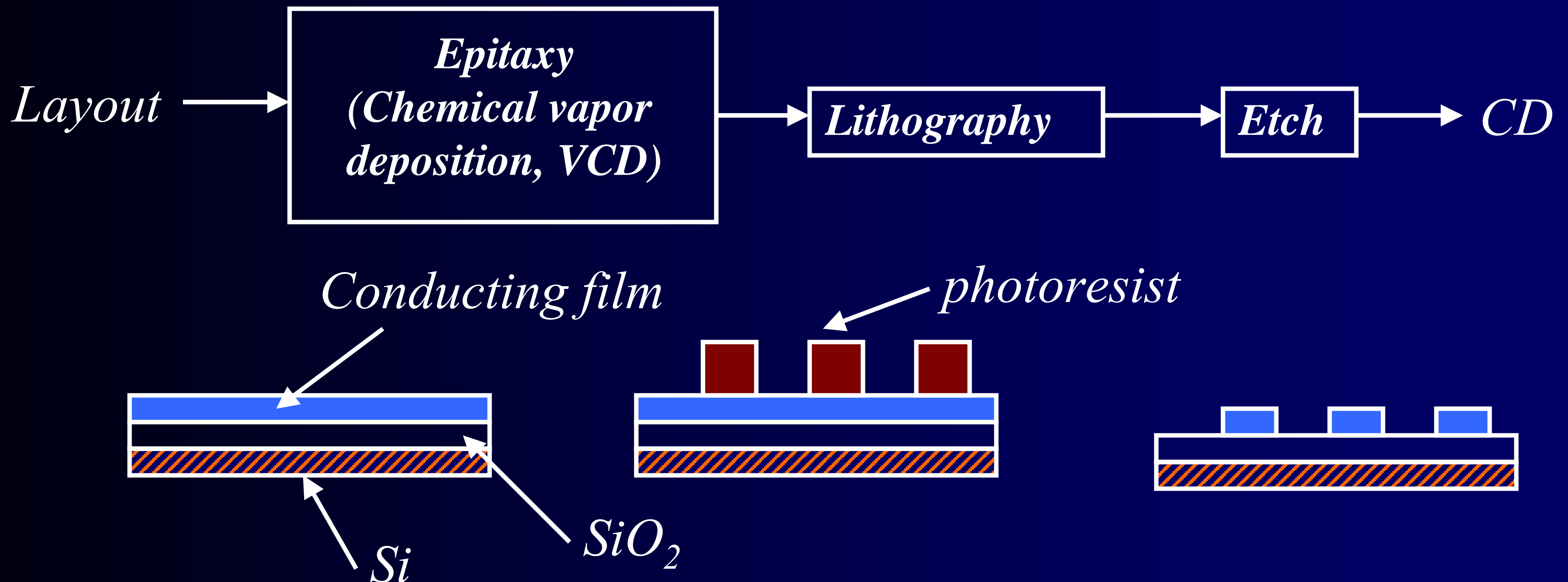
- It can also be classified as
  - Device variation
  - Interconnect variation

Device and interconnect





Variations happen at various stages



- Process may cause pattern-independent or pattern-dependent variations

Device/geometry (Boning & Nassif 99)

- Film thickness variation
  - Gate oxide thickness is critical
  - Usually well-controlled

- Lateral dimension (length, width)
  - Typically due to photolithography proximity effects
    - ✓ Systematic pattern dependent
  - to Mask, len, or photo system deviations
    - ✓ Not layout dependent
  - to plasma etch dependencies
    - ✓ Can have wafer scale dependency, or depend on layout density and aspect ratio (L/W)

- MOSFETs are sensitive to
  - channel length L, t_{ox}, and some W
  - L variation has received attention due to its impact directly on output current characteristics

Device/material (Boning & Nassif 99)

- Doping variation
  - Due to does, energy, angle, or other ion implant dependencies
  - Affect junction depth and dopant profiles
  - Hence, affect effective channel length L_{eff}
  - Also affect V_{th}
- Variation in deposition and anneal processes
  - Suffer substantial wafer-to-wafer and with-in wafer variations
  - May result in large device-to-device random variation
  - Impact contact and line resistance

Device/Electrical (Boning & Nassif 99)

- Vth variation
  - Often due to oxide thickness, geometry variations, and other sources
  - It is characterized separately because of its importance
- Discrete dopant variation
  - Random placement and concentration fluctuation due to discrete location of dopant atoms in the channel and S/D
  - Study shows that it is not a severe problem for logic but may affect SRAM containing large number of devices that should be well matched
  - Also cause Vth variation
- Leakage current
  - Sub-threshold leakage currents can vary significantly

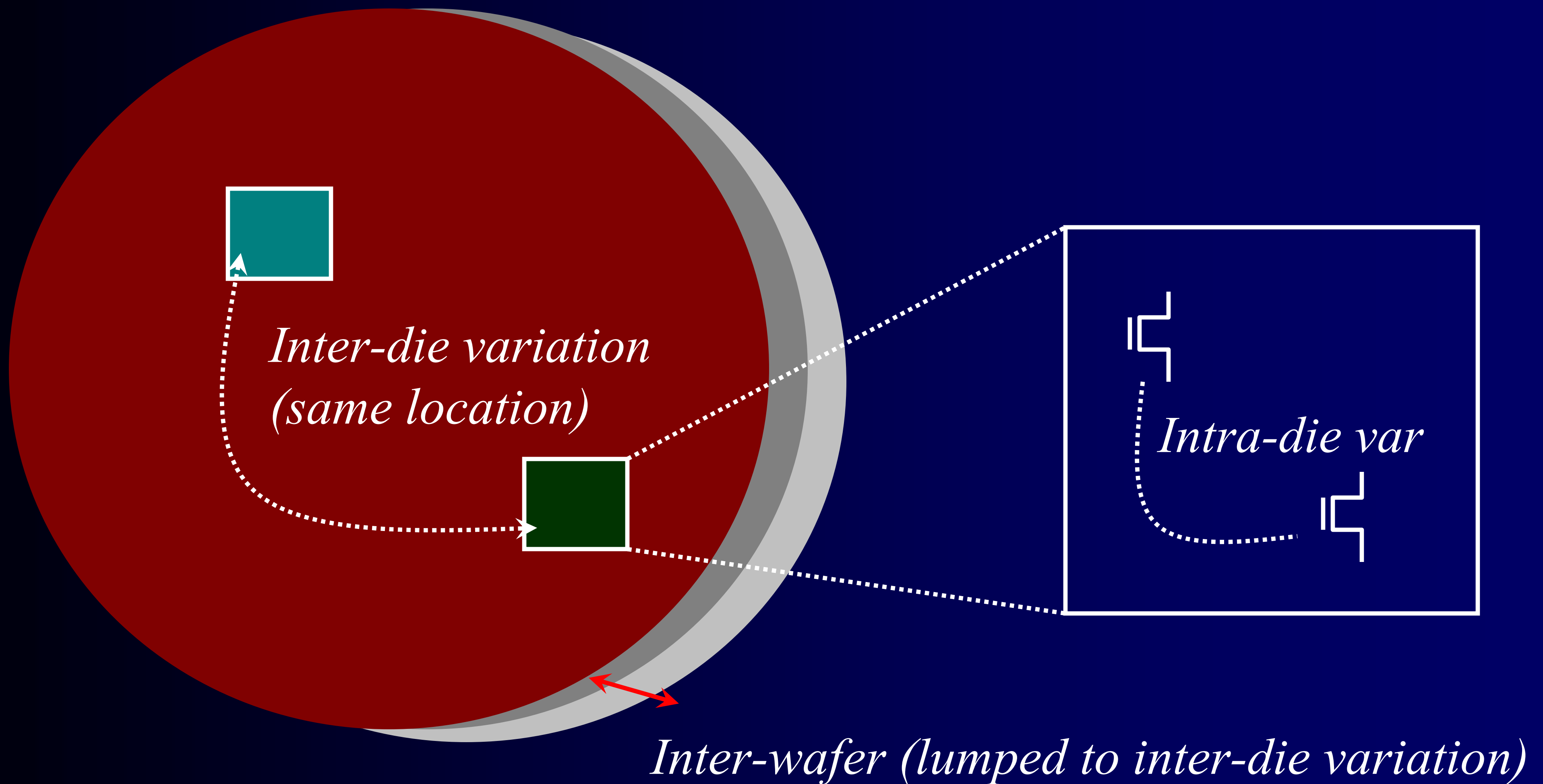
Interconnect/geometry (Boning & Nassif 99)

- Line width and space
  - Mainly photolithography and etch dependencies
  - Directly induce line resistance variation
  - Also cause capacitance variation within layer and across layers
  - Affect signal integrity analysis
- Metal thickness
  - Is usually well controlled in conventional process
  - Can have wafer-to-wafer and within-wafer variations
  - Copper polishing process can result in thickness loss of 10-20% depending on the patterns
- Dielectric thickness
  - Can have substantial variations
  - At wafer level, typically on the order of 5%
  - Within-die can have pattern dependent variation due to such as CMP
- Contact and via size
  - Affected by etch process and systematic layer thickness variation
  - Directly impact contact and Via resistance

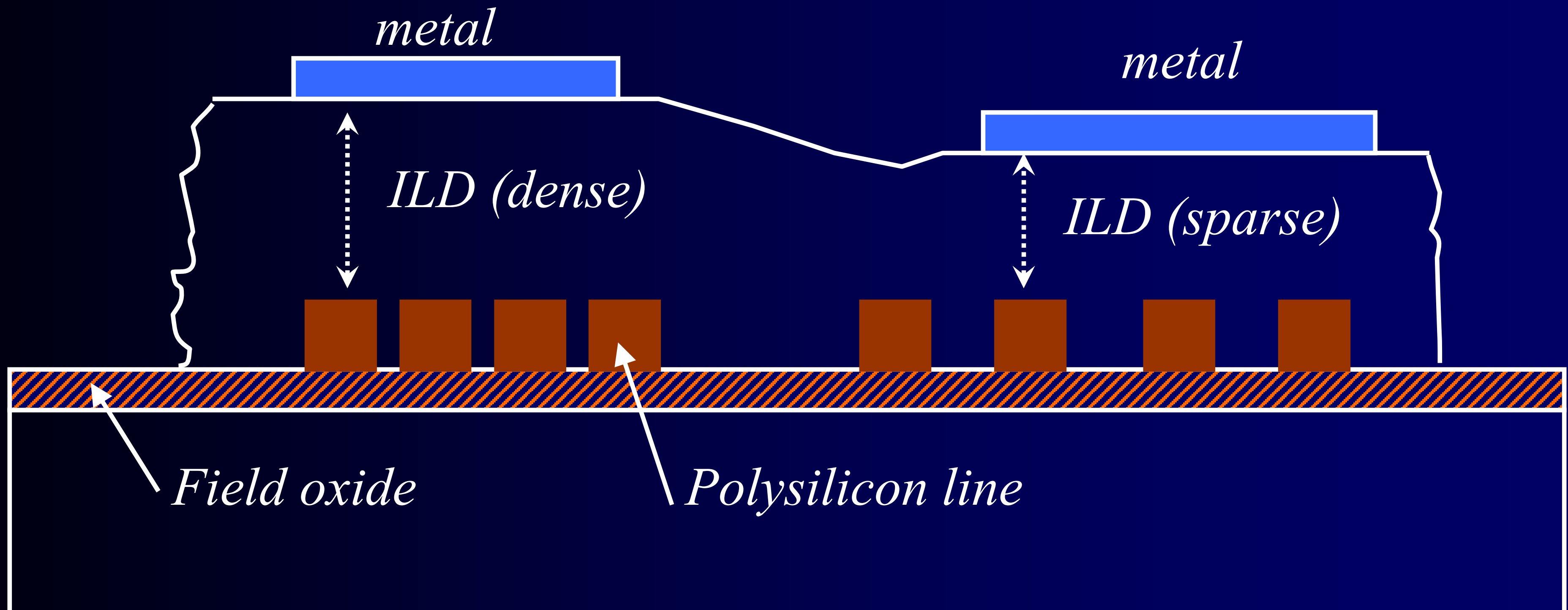
Interconnect/material (Boning & Nassif 99)

- Contact and via resistance
  - Sensitive to etch and clean processes
  - Substantial wafer-to-wafer variation
- Metal resistivity
  - Usually well controlled and vary wafer to wafer
- Dielectric constant
  - Depend on the deposition process
  - Is usually well controlled
  - Pattern dependent variation may be important for low-K dielectrics in interconnect

Inter-die Vs. Intra-die variations

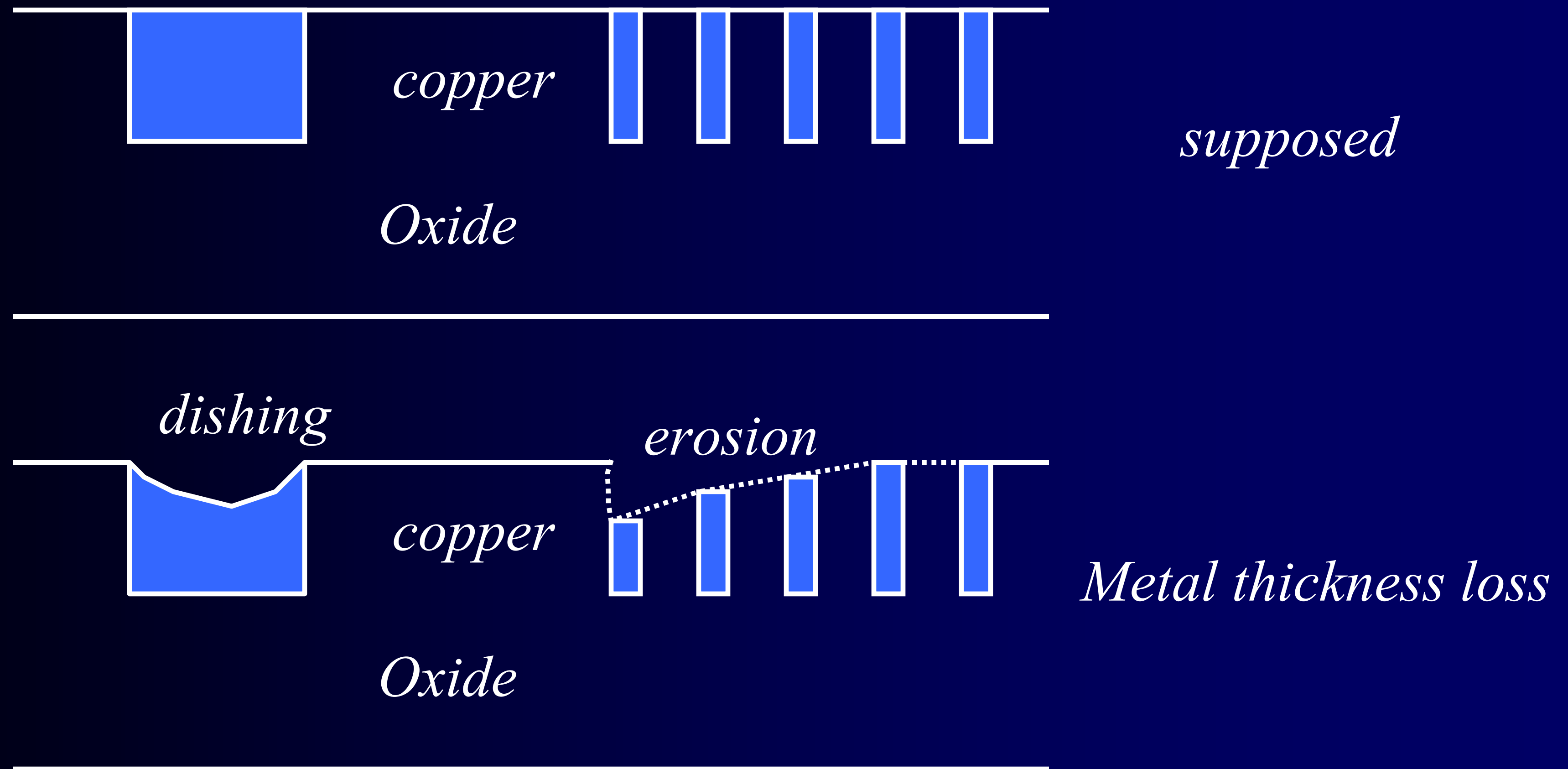


Pattern-dependent variation (intra-die)



Orientation, spacing, or other neighboring conditions of a location on a die can cause layout-dependent variations

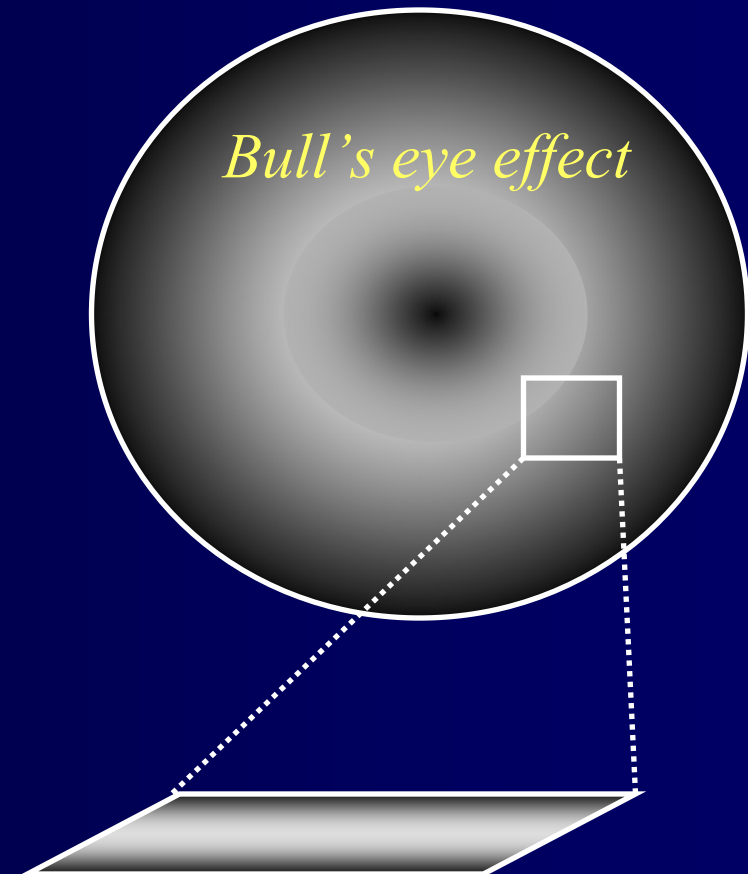
Pattern-dependent variation (Intra-die)



Bull eye effect

- Wafer level variations
  - Generally caused by equipment non-uniformity or other physical effects such as thermal gradients, etc.
  - Usually give smooth surfaces across wafer; 5-10% across
  - Usually exhibit symmetrical properties such as a "bull's eye"

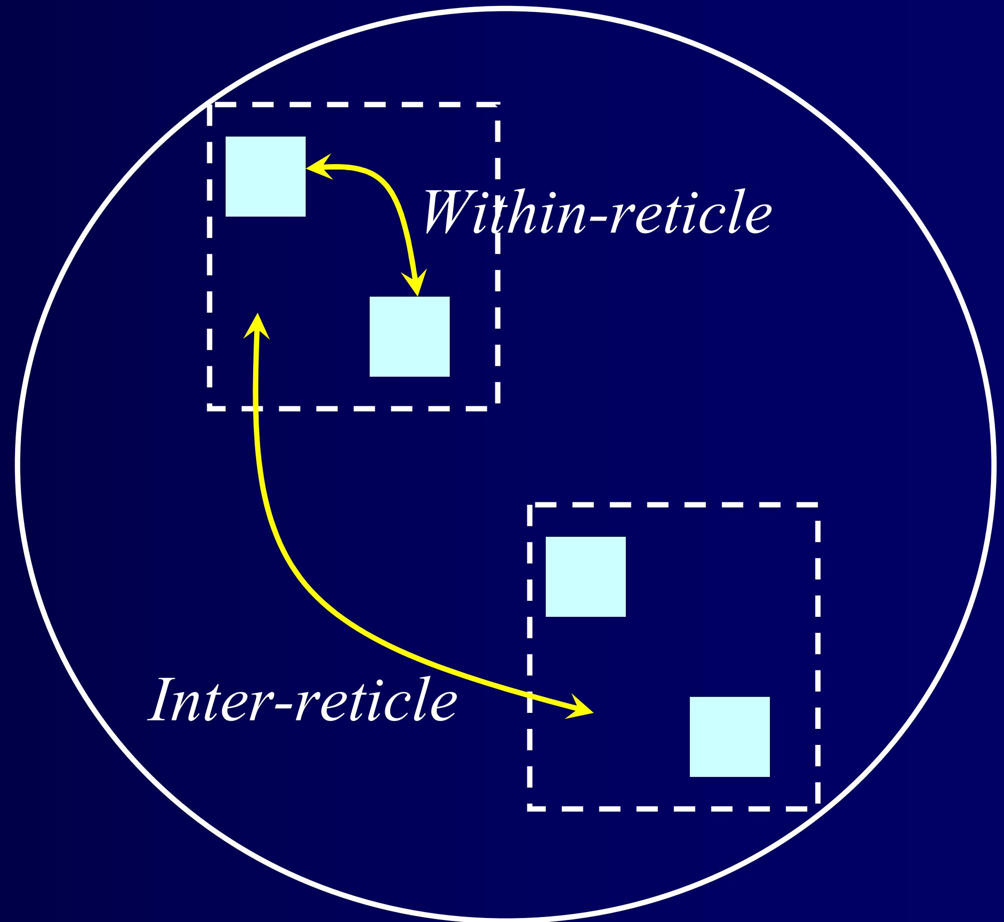
- Die level variations
  - Generally caused by layout-based and topography-based interactions with the process
  - Can be systematic or random



Wafer level variation on a die can be modeled as a smooth surface

Reticle level variation

- Systematic correlations in delay for dies within the same reticle
- Large random variations in delay for dies in different reticles

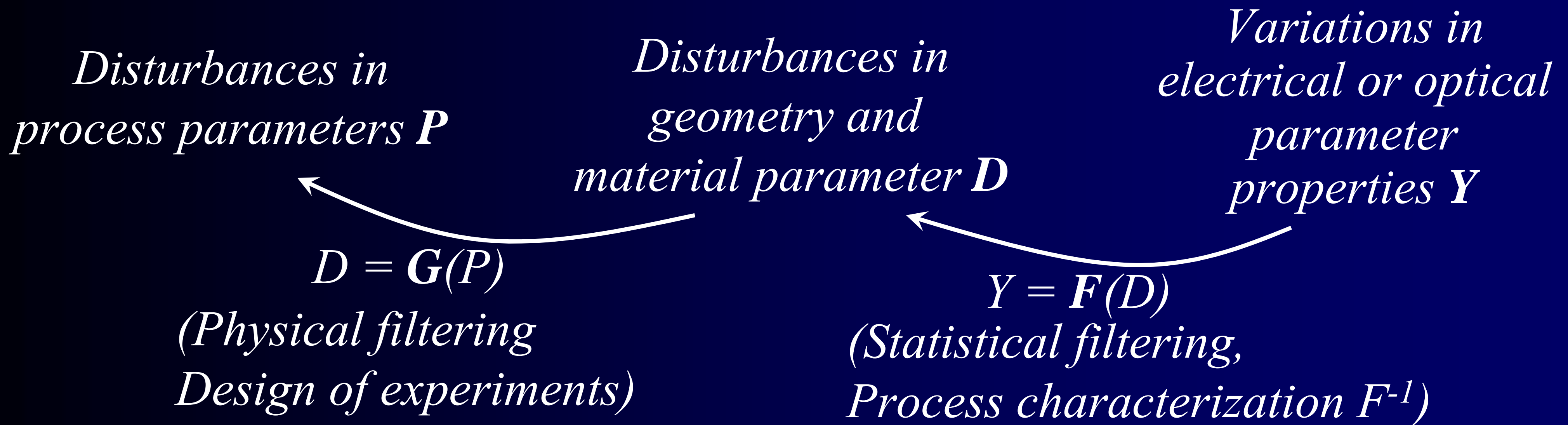


Studying variations

- Variations have been there for a long time
  - Historically, analog designs are much more sensitive to process variations than logic
    - Eg. Mismatch issue in two devices
    - See *Statistical modeling of device mismatch*, Michael, C.; Ismail, M.; Solid-State Circuits, IEEE Journal of, Volume: 27 , Issue: 2 , Feb. 1992

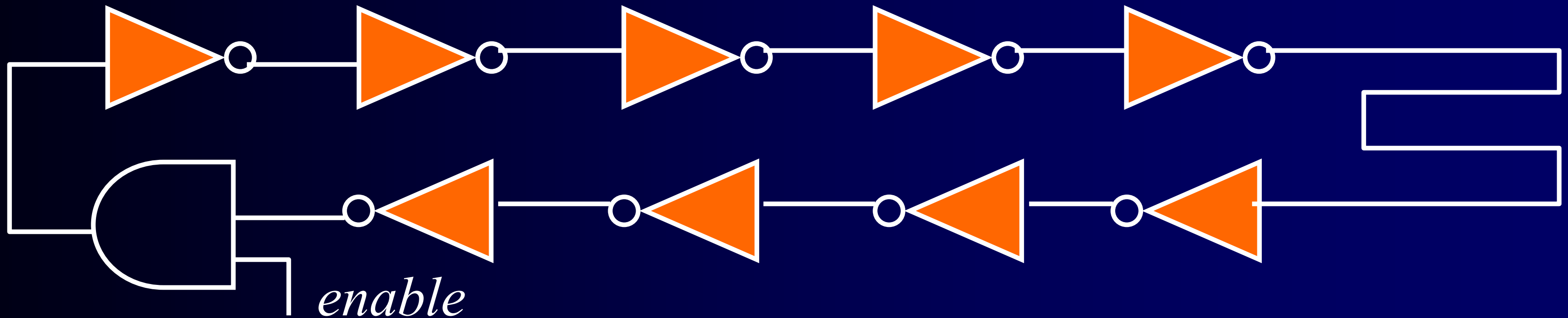
- Historically, the studies of process variations
  - *Are primarily for the control of process quality*
  - Diagnose unusual equipment disturbances
  - Diagnose unusual environmental fluctuations

Studying variations



- *P* are the (true) independent sources of variations
- *G* can be studied through design of experiments
- Parameters in *D* can be correlated
- Usually easier to observe *Y* (optical or electrical)
- *F* is studied through (statistical) process characterization
  - Here "filtering" corresponds to the diagnosis process to relate causes of variations

Vehicle – test structures



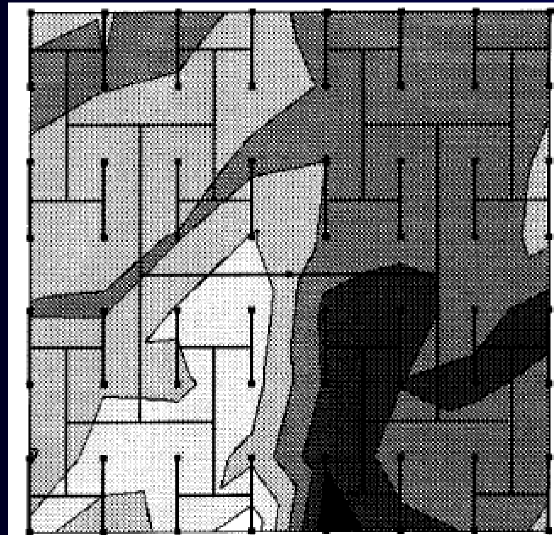
- A typical structure is a ring oscillator
- Typically, consider
  - What and how to measure
  - Poly spacing (study proximity effects)
  - Orientation
  - Poly density (study etch loading)
  - Metal fringing
  - Metal coupling
  - And so on …
- You can find a good tutorial on the topic at
  - http://www.tauworkshop.com/TauSlides/7.1.pdf (by Boning, et. al. 2002)

Example study : Gate CD variability on delay

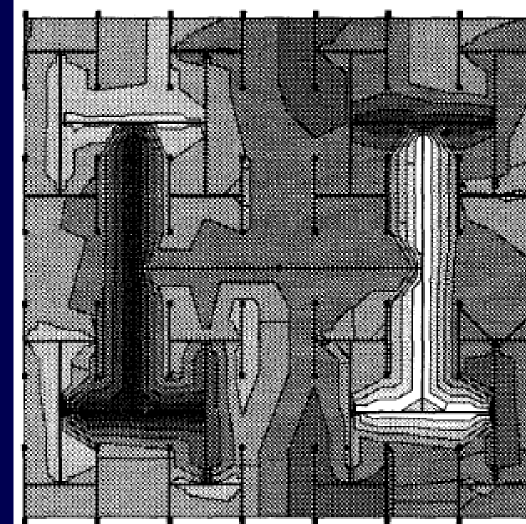
- See M. Orshansky et. al. 2002 TCAD, 2004 TSM
- Highlights
  - Study Lgate variability in 0.18μm technology
  - Development of test chips
    - Consider density and orientation
  - Consider impact on clock tree, cell delay, path delay, and circuit delay
  - Consider sampling resolution, sampling location, as well as optical proximity correction
- Conclude
  - CD variability is pattern dependent (density and orientation)
  - Intra-die CD variation is largely systematic
  - Cell delays vary as much as 17% among different locations
  - Clock skew vary as much as 8% of clock cycle (74ps)
  - Circuit delay degrades as much as 20%
  - Mask level spatial gate OPC should be employed
  - OPC that takes spatial gate information into account performs better than traditional OPC approach

Example study : variability on clock skew

- Source: [IEDM'98] S.R.Nassif. *Within-Chip Variability Analysis*
- Highlights
  - Based on 0.25μm technology
  - Study intra-die variability
  - Channel length variability ±0.035 μm
  - Wire width variability ±0.25 μm
  - Wire widths for worst-case skew – 48.9 ps
  - Channel lengths for worst-case skew – 171.5 ps



Channel lengths



Wire widths

Example study : Pattern-dependent variation on delay

- Source : V. Mehrotra et. al. DAC 2000, 172-175
- Highlights
  - Study delay variation in both Aluminum and copper (0.60 μm metal and ILD thickness)
  - Study clock skew in 0.25 μm technology
  - Study pattern dependent effects such as density to ILD thickness, dishing and erosion in CMP
- Conclude
  - Models for systematic variations are required for accurate simulation of circuit performance
  - Interconnect CMP variation can increase bus delay by more than 30% even in copper technology
  - Clock skew is not strongly impacted by interconnect CMP variation
  - Variation in device gate length can significantly alter path delays with an increase in maximum skew of about 50ps

Other studies

- Variation in Vth
  - M. Niewczas, IEEE ICMTS, 1997
    - Focus on test structures to study Vth
  - T. Tanaka et. al. IEDM 2000
    - Focus on variation in dopant profile
- Variation in gate line edge roughness
  - S. Xiong, et. al. IEEE Tran. Semi Manu. 2004
  - A. Asenov, et. al. IEEE Tran. Elec. Device, 2003
  - Roughness is not an issue today
  - May affect leakage current due to short channel effect as technology scales
- Circuit sensitivity to interconnect variation
  - Z. Lin et. al. IEEE Tran. On Semi Manu. 1998
  - Interconnect is hard to characterize and model
  - Develop a model for interconnect variation
- Sub-wavelength lithography
  - A. Kahng and YC Pati, DAC 1999
  - Conclude the importance of OPC and need for more effective OPC algorithms
- And many others …

Notes

- When you read a paper, first understand what is its primary objective
  - 1. Is it for diagnose process disturbance and process control/yield improvement?
  - 2. Or is it for modeling support for design tools?
- There are many more studies in the first category than in the second category
- For timing impact, 1st order effects have been
  - Device: Leff, Vth
  - Wire: Via

Modeling of variations - linear model

- On a given die, variations are modeled as a linear combination of (lumped) independent random variables
  - $P = P_0 + P_{interdie} + P_{intradie}(x,y) + P_{intradie_random} + \varepsilon$
  - $P_{intradie}(x,y)$ describes the correlation structures
  - When layout information is not available,
    - ✓ $P_{intradie}(x,y)$ can be modeled as random
    - ✓ We can assume worst cases
    - ✓ Or we can assume a proximity function
- This model may be what we want for timing analysis, but may not be easy to obtain

Remarks (Nassif, Boning, Hakim, ICCAD04)

- The availability of intra-die variation models directly link to the availability of test structures being characterized
- Tracking the drift of variations over time can be expensive and prohibited in practice
- Predicting other variations such as power noise or intra-die temperature variation needs to wait until late stage of design when global placement is available

Note – level of modeling

- When we talk about "a statistical model," we need to be clear on the abstraction level
- Each statistical model consists of a set of random variables
  - And more importantly, their correlations
- We need to keep in mind that the methods moving from one model to another are just approximation
  - Therefore, accuracy is lost along the modeling process

Variation trends

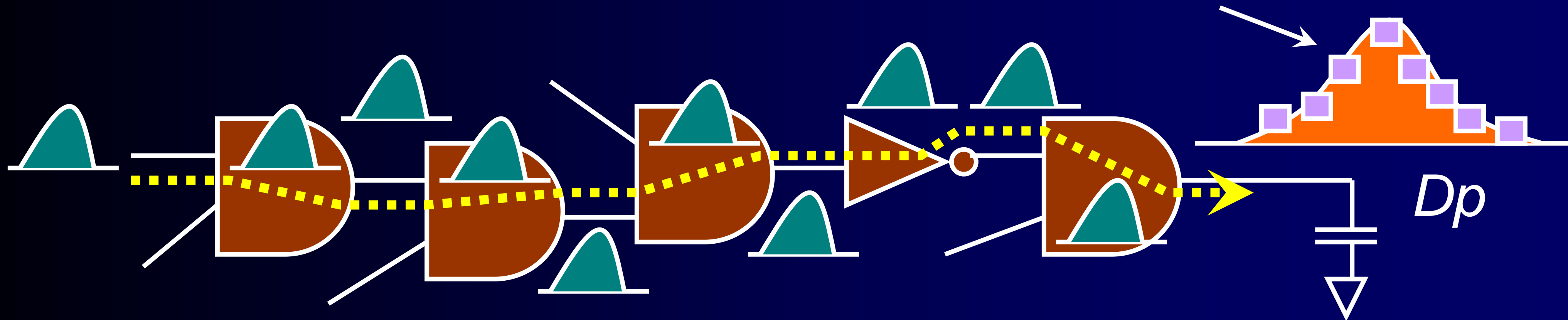
	Impact on delay	Impact on power	Trend
L_{eff}	Large	Large	Flat
W	Small	Small	Decreasing
V_{th}	Small	Medium	Increasing
Interconnect	Small	Low	Increasing
Other	Variable	Variable	Flat

N Hakim, ICCAD04, N Menezes, VTS05

Break 5 minutes for questions

Next, we will focus on timing impacts

Timing variability of a path

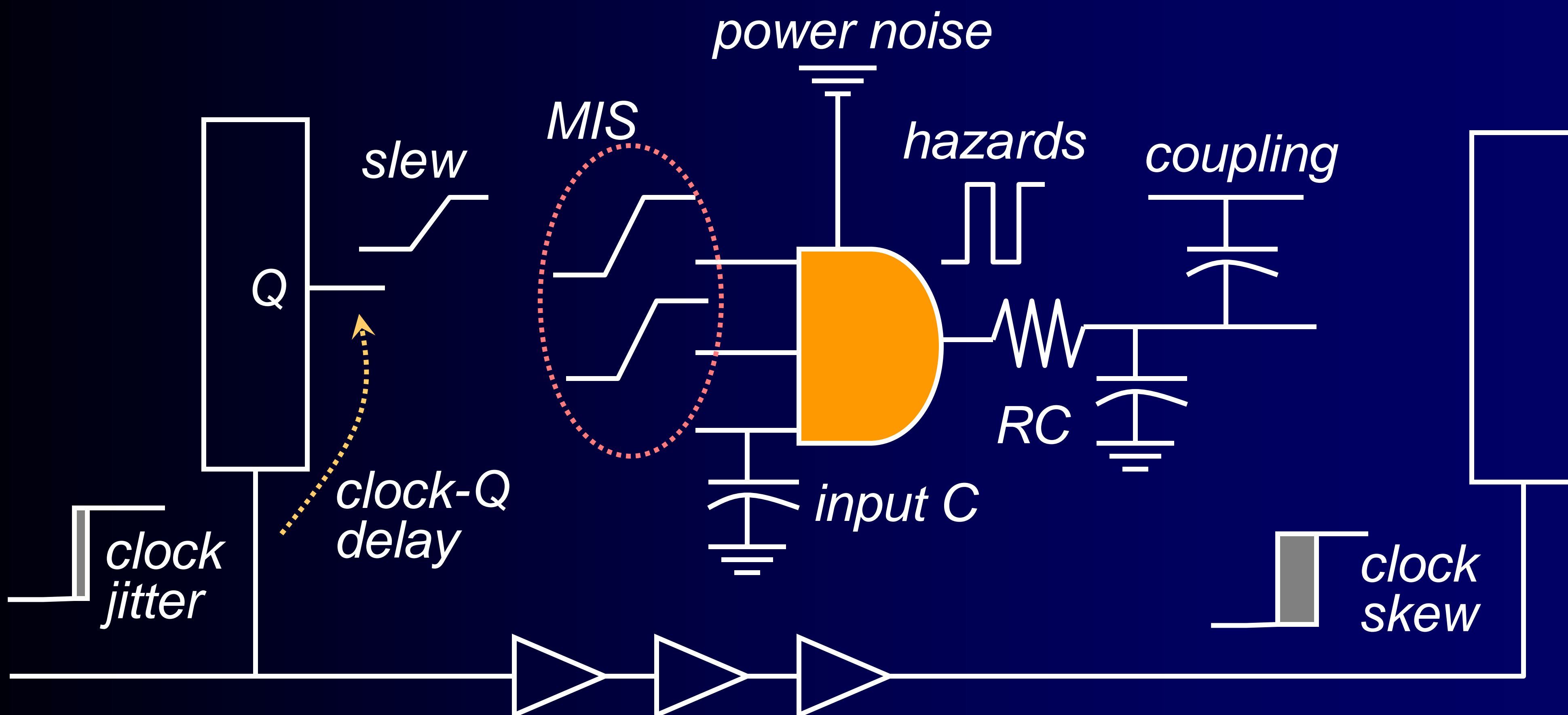


Dp = Dl + (G1+G2+G3+G4+G5) + (W1+W2+W3+W4+W5)

Input delay — Gate delays — Wire delays

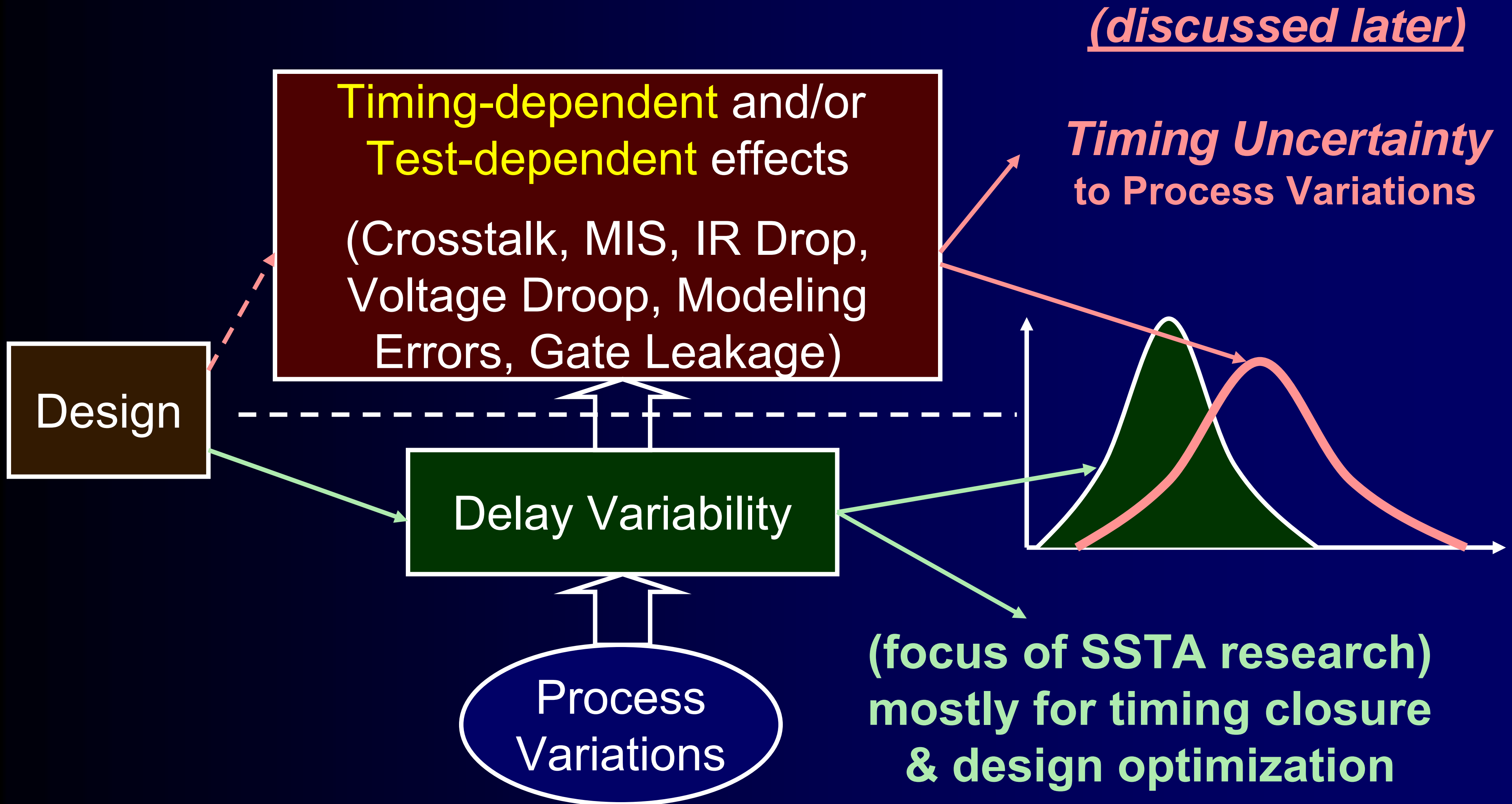
- There is a way to test against this simple view in order to estimate the delay variability
  - Test against intro-die variation
  - Test against die-to-die variation
- Also, popular SSTA approaches deal with timing variability

Timing uncertainty of a path



- Many effects are "sensitive to delay variation," complicating the simple view
- Many effects also depend on test patterns
- These effects create *timing uncertainty* in path delay

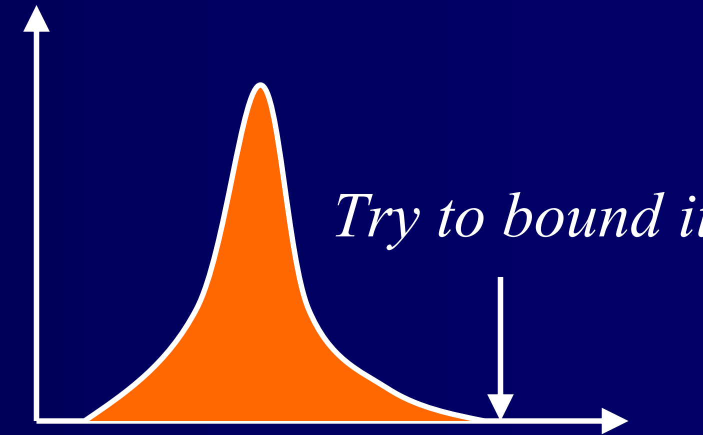
Timing variability vs. timing uncertainty



"Smooth" vs. "noisy" effects

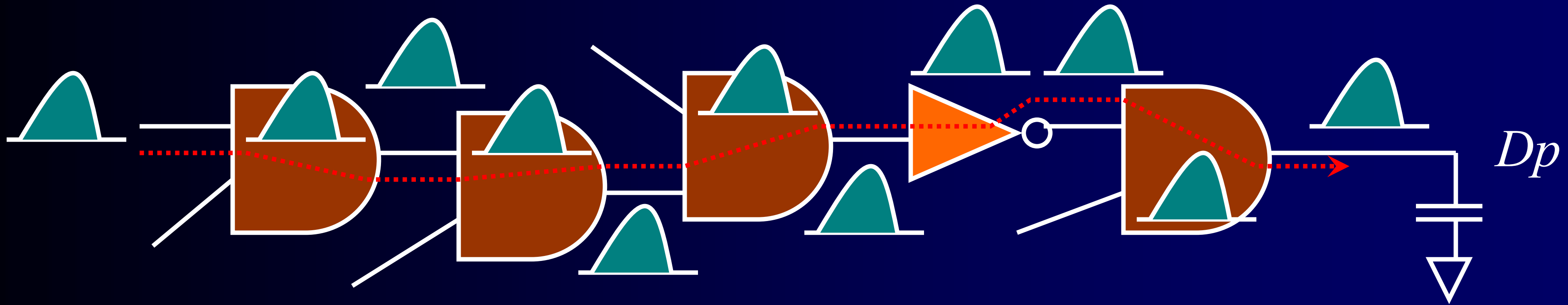
A statistical system
F$(x_1, x_2, \ldots, x_n)$ → *behavior*

- Each x_i can be modeled as $x_0 + P_{random} + P_{systematic}(x,y) + \varepsilon$
- Smooth effect
  - **Variations result in one (Gaussian-like) timing distribution**
  - **F** is a continuous function
  - Easy to capture and test



- Noisy effect
  - **Variations result in Non-Gaussian and discontinuous timing distributions**
  - **F** has discrete components
  - Hard to analyze and test

Consider the simple path timing view first



$$Dp = DI + (\underbrace{G1+G2+G3+G4+G5}_{\text{Gate delays}}) + (\underbrace{W1+W2+W3+W4+W5}_{\text{Wire delays}})$$

Input delay (arrow pointing to DI)

- **Assumption**: Nominal timing analysis captures the mean delay very well
- In test, our goal is to quantify the "sigma"
  - Path's delay sigma depends on individual cell delay sigmas
  - And how they are *correlated*

Simple probability calculations

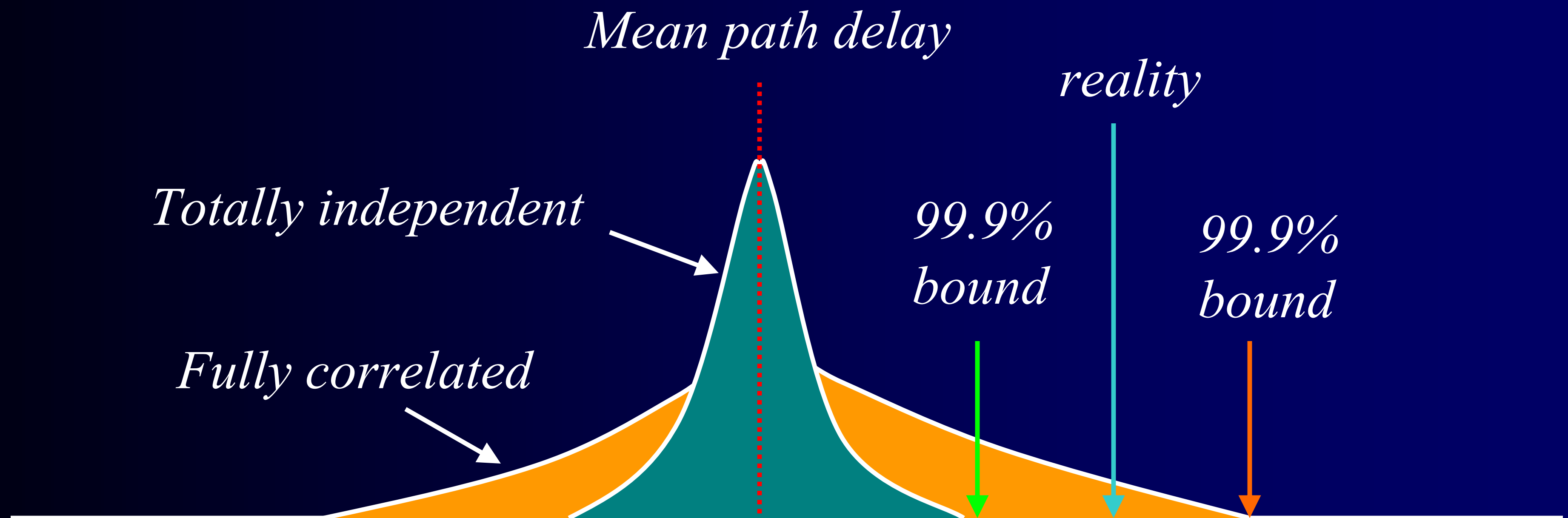
- Assume $Y = x_1+x_2+x_3+x_4+x_5$
  - Each $x_i \sim N(100, a\,\sigma_i + b\,\sigma)$
  - Where $\sigma \sim N(0,1)$, $\sigma_i \sim N(0,1)$,
  - Enforce the constraint that $a+b=5$
    - So, the sigma of each random variable is 5, regardless of what the a and b values are
  - For each random variable x_i
    - Sigma / Mean = 5%
  - σ_i represents the independent source of variation
  - σ represent the correlated source of variation

Simple probability calculation

- If a=0, b=5, $Y \sim N(500, 5\times b) = N(500,25)$ *Fully correlated case*
  - Sigma / Mean = 5%
- If a=5, b=0, $Y \sim N(500, (5^2+5^2+5^2+5^2+5^2)^{1/2}) = N(500, 11.18)$ *Fully independent case*
  - Sigma / Mean = 2.236%
- If a=1, b=4, $Y \sim N(500, 5\times b+(a^2+a^2+a^2+a^2+a^2)^{1/2} = N(500, 22.236)$
  - Sigma / Mean = 4.447%
- If a=2, b=3, $Y \sim N(500, 19.472)$
  - Sigma / Mean = 3.894%
- If a=3, b=2, $Y \sim N(500, 16.71)$
  - Sigma / Mean = 3.342%
- If a=4, b=1, $Y \sim N(500, 13.94)$
  - Sigma / Mean = 2.788%

Cases in between

Path delay under the simple view



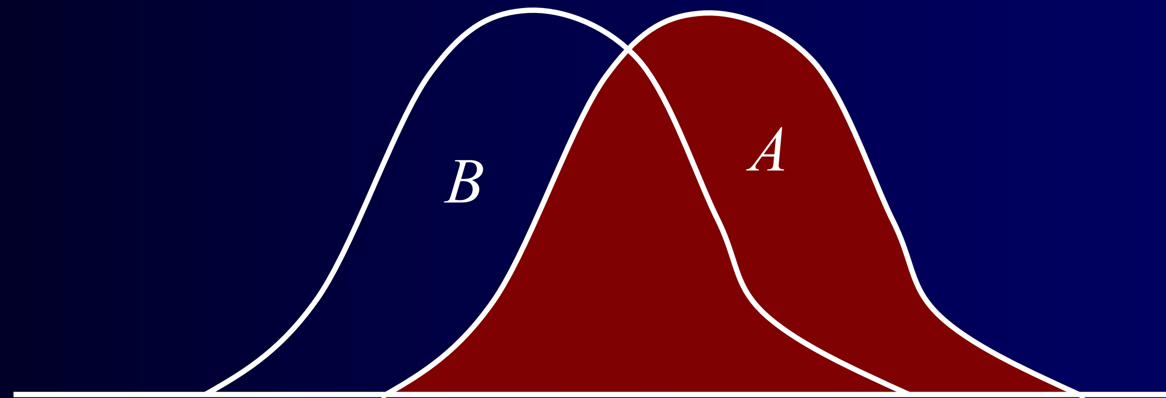
- If a path has n components, each with identical ±5% variation,
  - If all components are totally independent, the path delay is with $\pm 5/(n)^{1/2}$% variation (which decreases as path length increases)
  - If all components are fully correlated, the path delay is with ±5% variation
- in reality, a path delay variation amount is in between

Extend to a simple view for Fmax

- Given a delay t, to find T = prob (Fmax > t), we need to know how D_1 … D_N are correlated

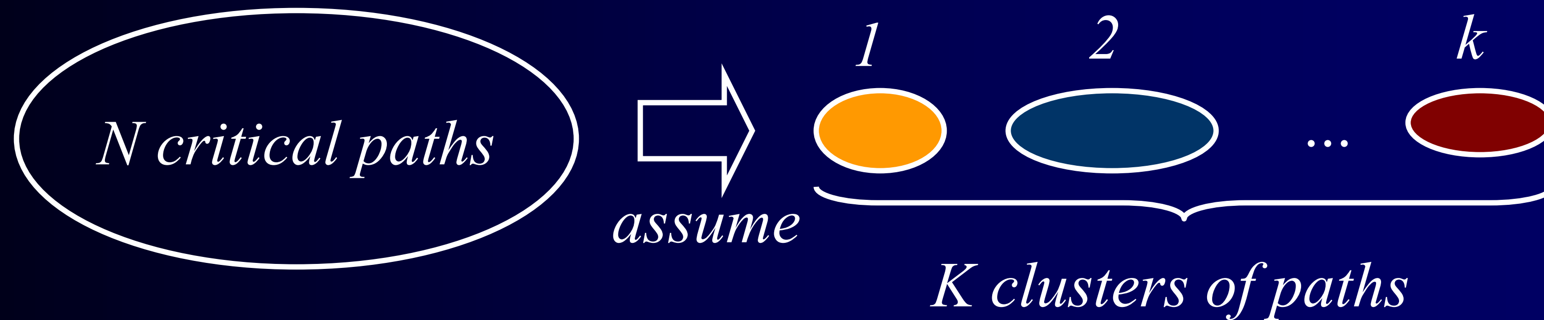
- Our goal is to test a few paths and be able to utilize the results to estimate Fmax
  - How many paths (what paths) to test?

Note on the max of two path delays



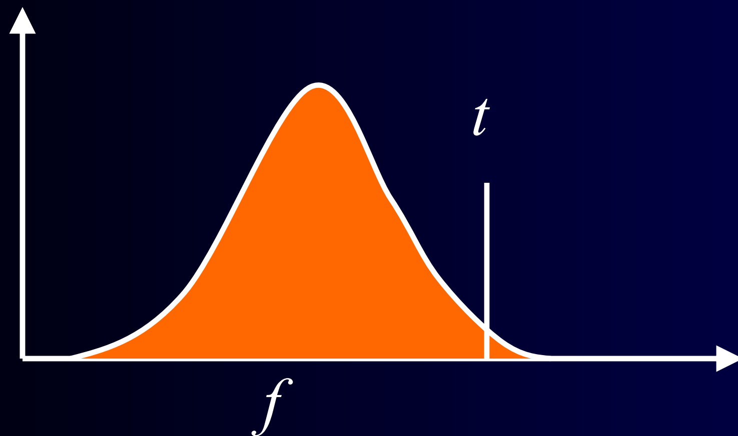
- If A and B are highly correlated, max(A,B)=A
  - This implies that if path delays are highly correlated
    - ✓ Their 3σ delays are good for ranking those paths
    - ✓ If we establish a bound for A, it is also a bound for B

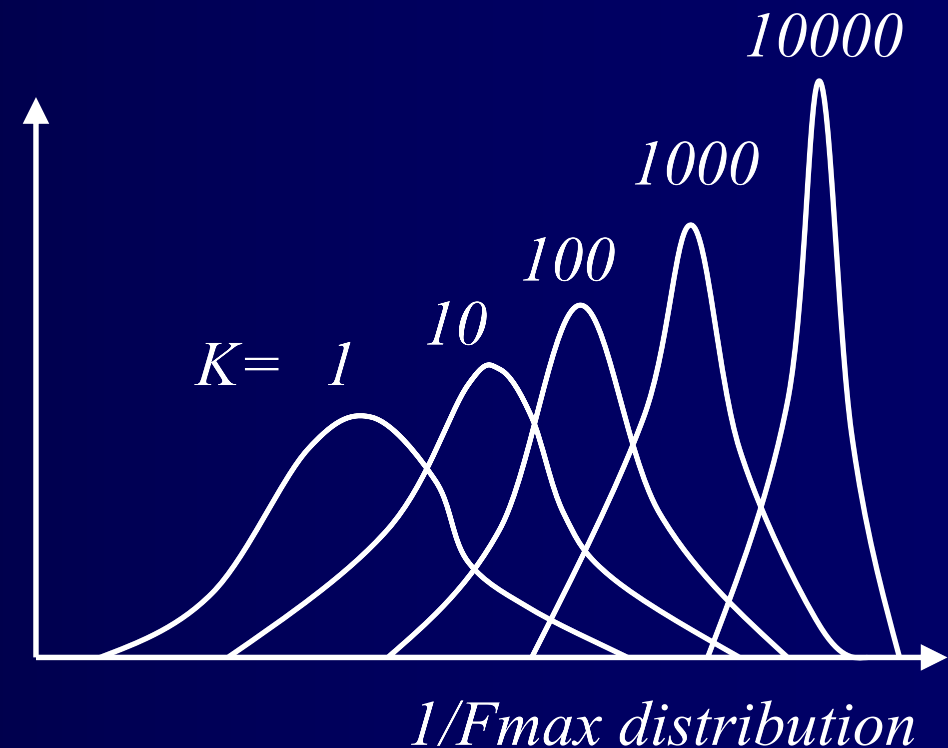
Fmax – simple view



- Little correlation between any pair of clusters
  - And all paths within a cluster are highly correlated
- Fmax ≈ max(D_1 … D_K) where
  - D_i is the delay random variable of a path from cluster i
- Let's assume all Di are with identical probability density function (PDF) f

Fmax – simple view (Bowman, et. al. 2002, and 2004)





- CDF $F(x<t) = \int_{-\infty}^{t} f(x)\,dx$
- CDF $F_{max}(x < t) = [\ F(x<t)\ ]^K$
- PDF $f_{max}(t) = \partial F_{max}/\partial x = \partial F(x<t)^K/\partial x = K\, F(x<t)^{(K-1)}\, f(t)$
- As K becomes bigger,
  - The distribution of 1/Fmax (delay) becomes narrower (smaller variation)
  - However, the mean of the delay distribution becomes larger as well

A simple Fmax estimation methodology

- Recall our model for variation $P = P_0 + P_{interdie} + P_{intradie}(x,y) + P_{intradie_random} + \varepsilon$
  - where $P_{intradie}(x,y)$ decides the correlation structure
- Suppose the correlation between two paths is entirely decided by $P_{intradie}(x,y)$
- Given an intra-die variation model, suppose that we can find a set of K independent paths as mentioned before
  - Such that any other path is highly correlated to one of these K paths
- Fmax can be determined by testing these K paths

A simple binning methodology

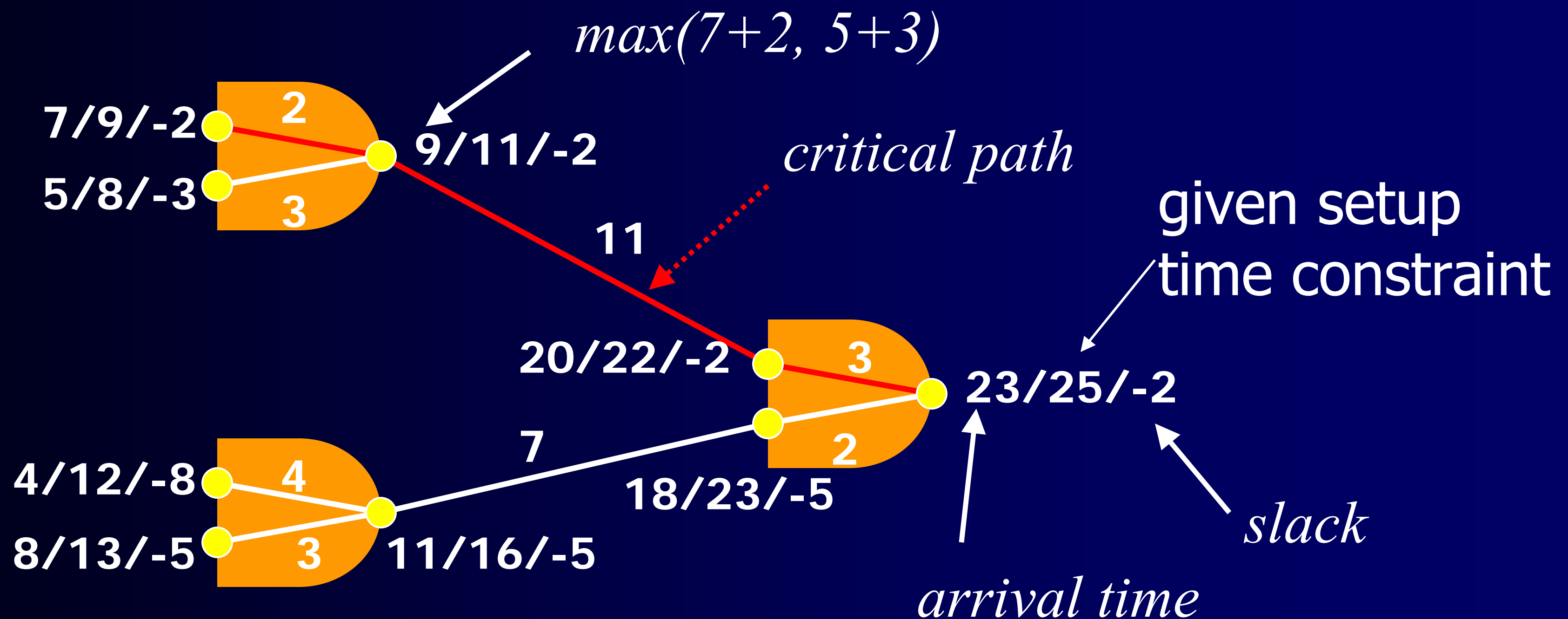
- To bin against systematic intra-die variation
  - We need to test the K independent paths
  - If intra-die variation gives strong proximity correlation across the whole die
    - ✓ We only need to use a few paths
- To bin against random inter-die variation
  - We only need to test one critical path (probably the one with the largest mean delay)
  - Because this variation affect all paths equally
- See Bowman, et. al. 2002, and 2004 for detail

Break 5 minutes for questions

Next, we will switch topic to

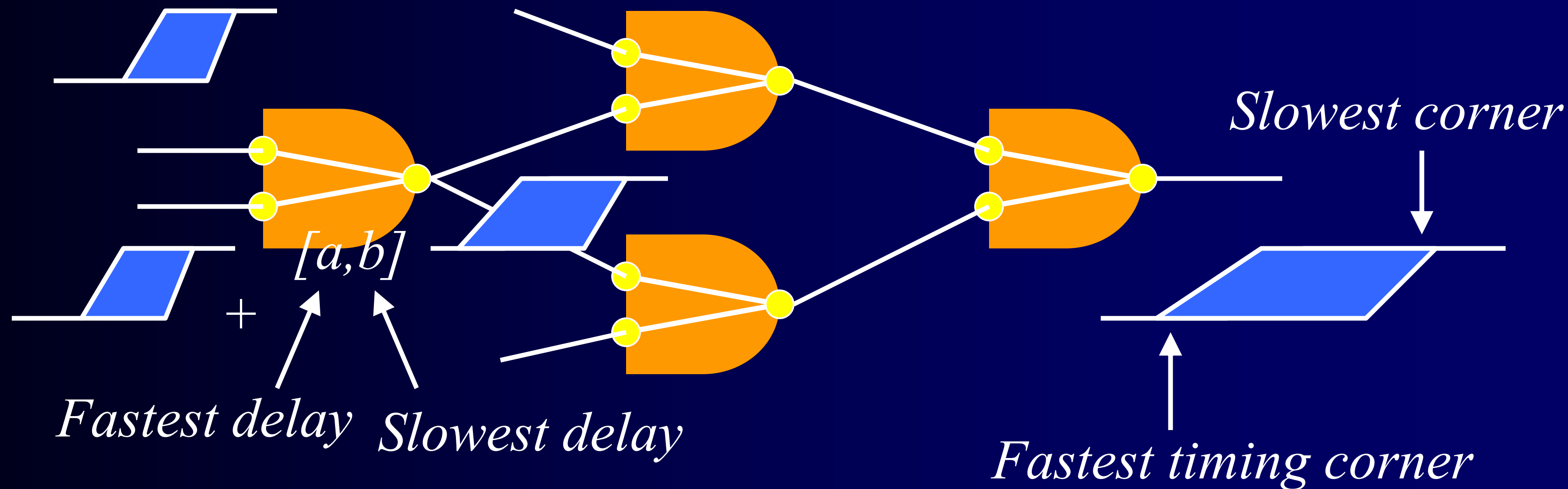
Macro-modeling and timing analysis

Static timing analysis (STA) 101



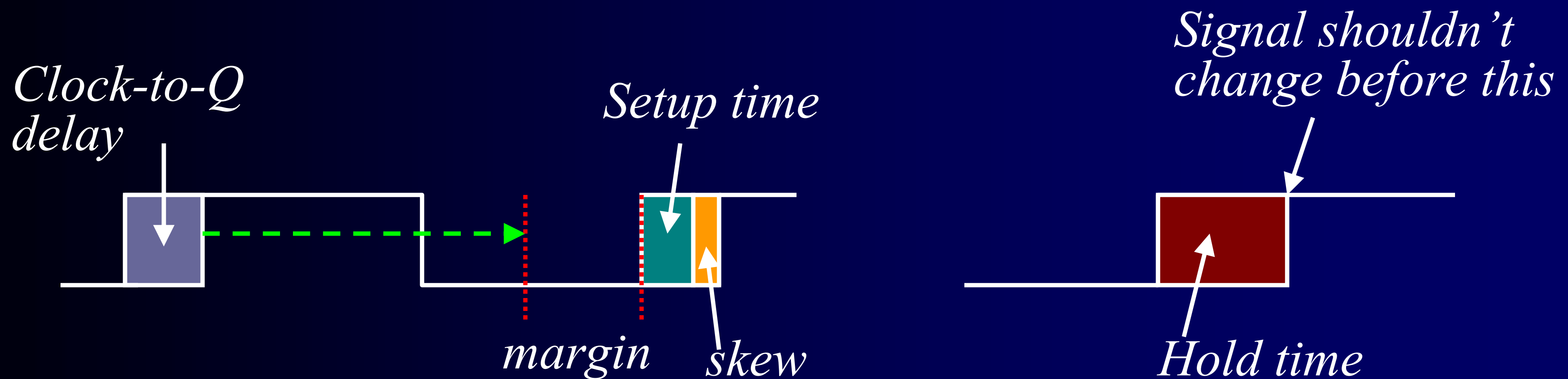
- In STA, the basic operations are "max" and "+"
- This is a fixed-delay STA
  - Each cell pin-to-pin delays are pre-characterized
  - Interconnect delays are pre-calculated before STA
  - After STA, critical paths can be identified

Propagation of timing windows



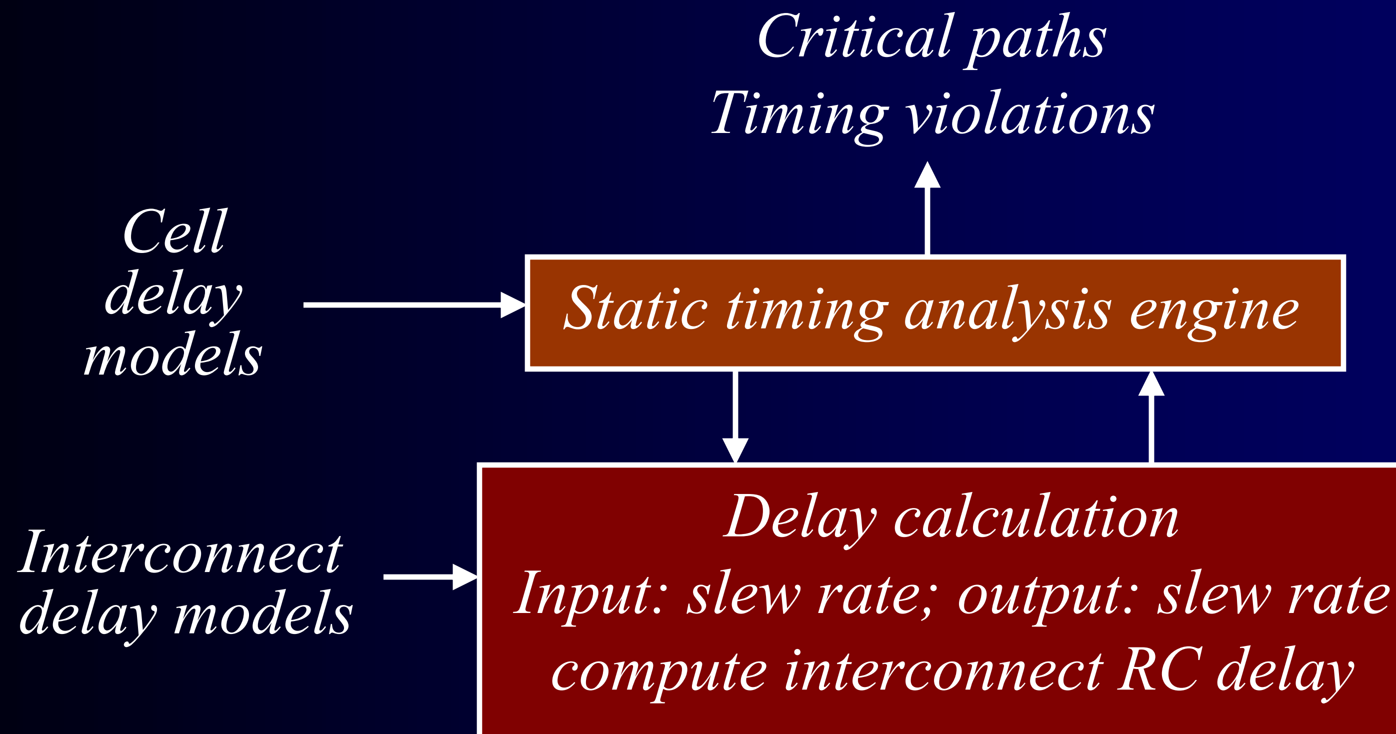
- Typically, delay is characterized as a range [fastest, slowest] due to process variations
  - Timing analysis propagate timing windows
  - Increased variations increase these windows

Timing constraints 101



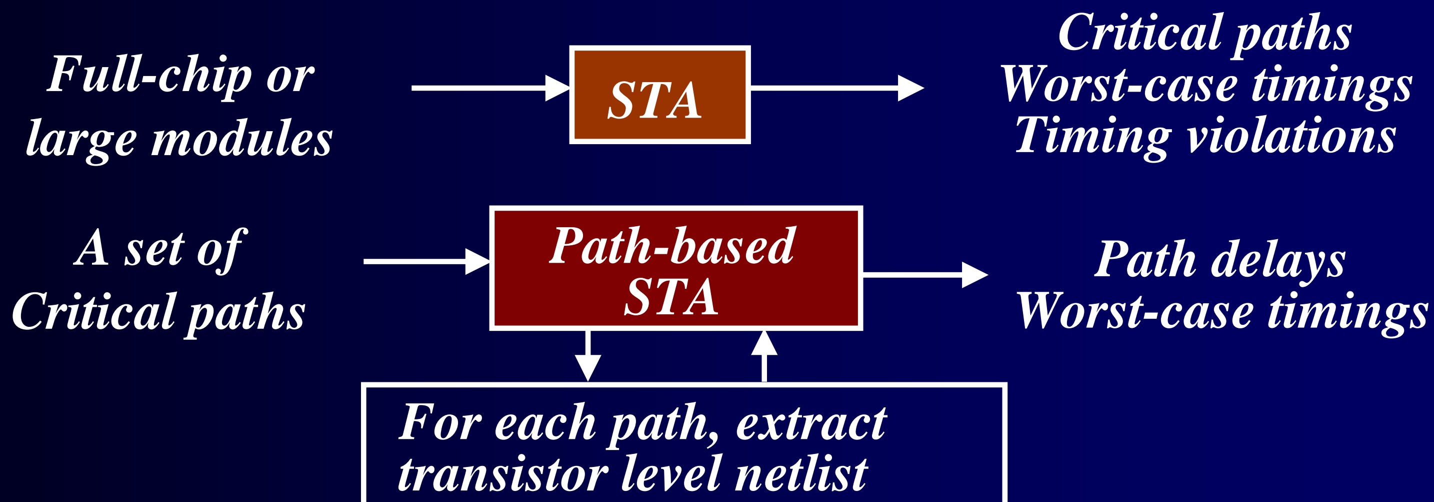
- Setup time constraint
  - Path delay cannot be too slow
  - Signal should arrive before active clock edge
- Hold time constraint
  - Path delay cannot be too fast
  - Signal should not arrive too early after active clock edge

STA 101



- STA is for design timing optimization and convergence
- Before layout, worst-case RC delays can be used

Block-based vs. path-based



- STA or block-based STA
  - Usually rely on cell models
  - The goal is to filter out critical paths for further analysis and optimization
- Path-based STA
  - Usually reply on transistor level timing analysis
  - Try to achieve SPICE accuracy
  - Do it by following a path-by-path basis
  - Then, worst timing can be simply max(path delay, path delay, …, path delay)

Timing macro-modeling

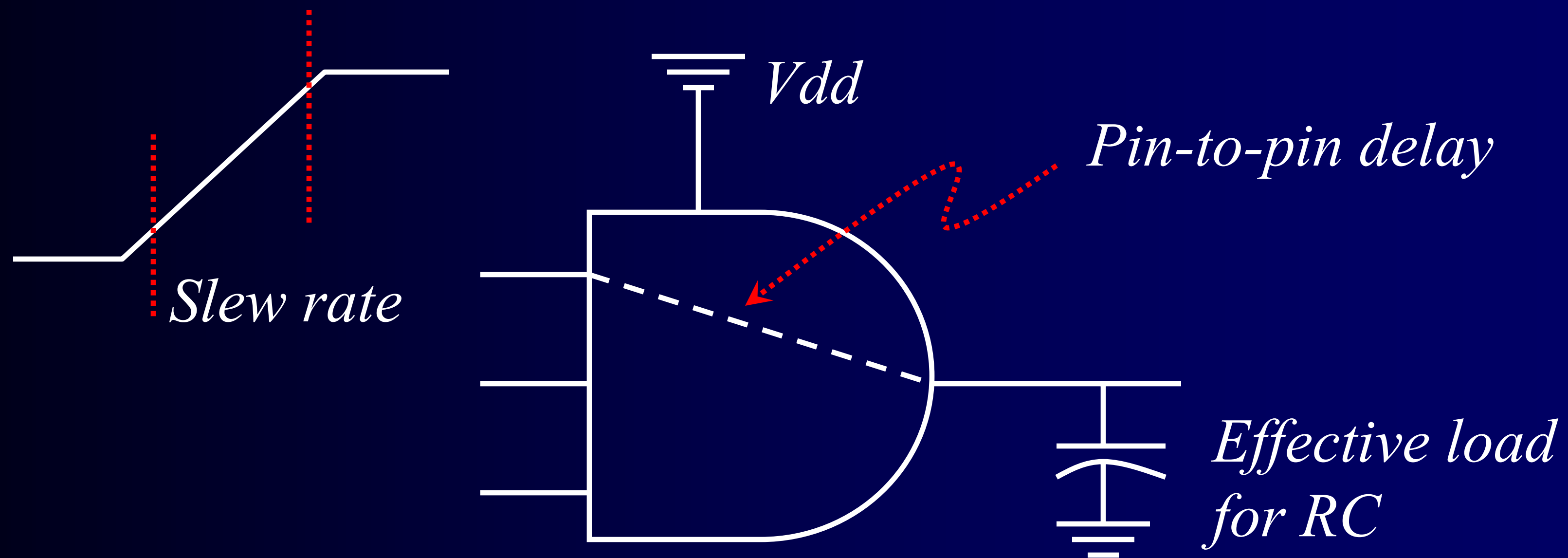
- Objective: Creating reduced models at transistor level, gate level, or cell level to support fast timing simulation
  - Treat SPICE simulation as golden
  - At transistor level, support path-based timing analysis
  - At gate/cell level, support block-based full-chip timing analysis

Timing Macro-modeling

- Gate/cell level
  - STA focused
  - Support place-and-route tools for optimization

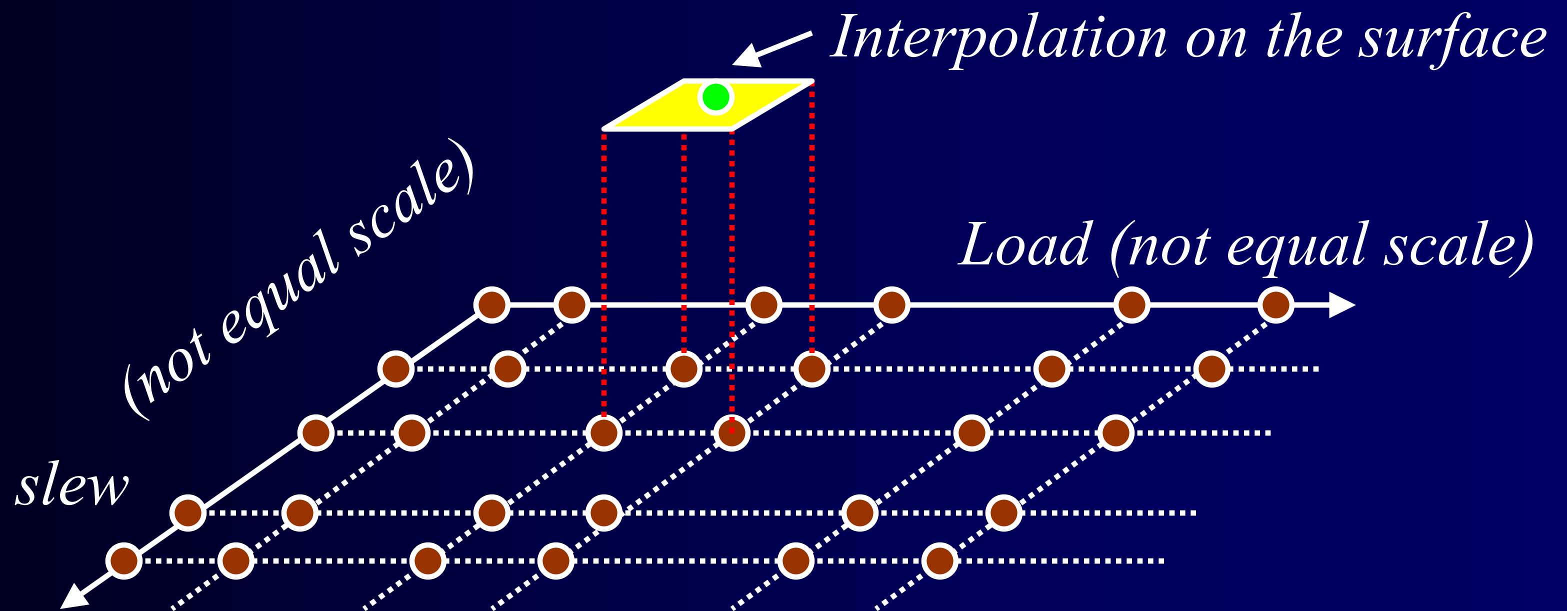
- Low-level
  - For transistor level simulation
    - Path-based timing analysis
  - Care about voltage waveforms rather than slews
    - Waveform is piece-wire modeled
      - Each piece may be modeled as a linear, quadratic, exponential function
      - Eventually, combine all pieces together
    - Achieve almost SPICE comparable accuracy
  - Focus on timing/delay characteristics
  - usually >100x faster than SPICE

Cell macro-modeling 101



- Each cell's pin-to-pin delay is characterized as a function f (S, L, V, T)
  - Slew, Load, Vdd, and Temperature
  - Each pin-to-pin is characterized separately
    - Typically at fastest process corner and slowest process corner [fast,slow]
    - Delay can be characterized as a slew rate, with respect to the 50% point of the input slew
  - Assume that 1 input transitions at a time

Cell macro-modeling 101

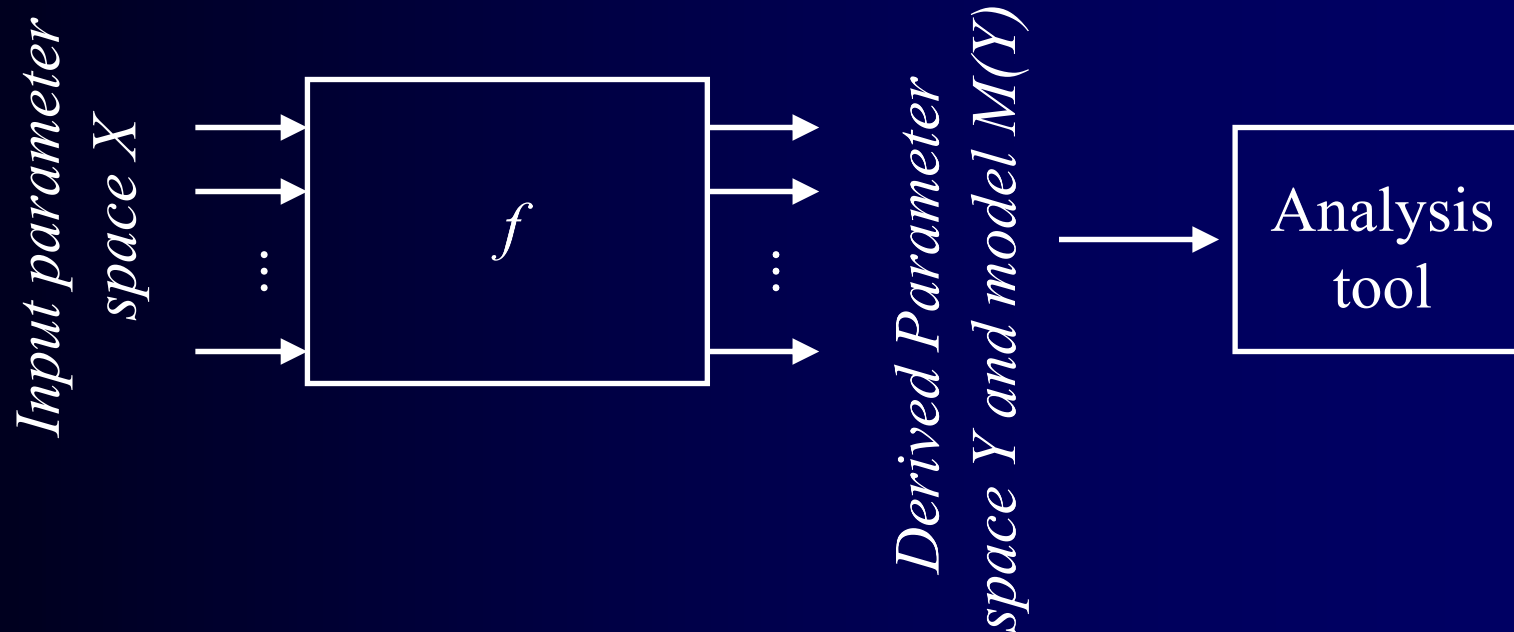


- The most common way to store cell delays is to characterize them (with SPICE, for example) at multiple slew-vs-load points
  - Store these values as a table
  - For an un-characterized slew-load point, use interpolation to find its delay
  - For changes of temperature and Vdd, apply a sensitivity factor Δ
- Alternatively, we can characterize the delay values as equations
  - For example, delay = $0.3\ S + 0.5\ L - 0.1\ S\ L^2 + 1.7\ S/L$
  - If stored as equations, table values can be used for outliers

Interconnect RC (capacitance extraction)

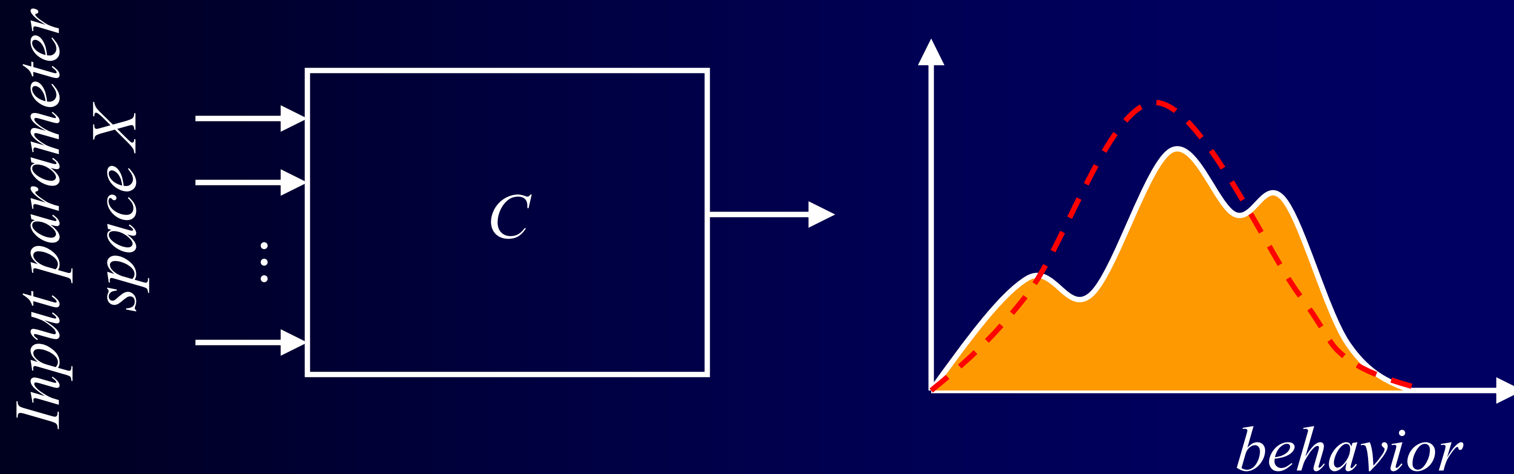
- 2D extraction
  - Consider area overlap between 2 layers (area C), side wall in the same layer (side C), and side wall to the adjacent layers (fringing C)
  - The relationships relating geometry to C are characterized by the fab
  - Commonly used approach (can be implemented as a rule based tool)
  - Practical for worst-case STA, even though it is not accurate
- 2.5D extraction
  - Consider more layers and within a layer, the distance between wires
  - Pre-characterize unit region based on possible patterns and develop library
  - Commonly used for high-performance designs
- 3D extraction
  - Most accurate but expensive
  - Boundary element method (BME), finite element method, Monte Carlo method
  - Often applied at package or in characterization of patterns in 2.5D method
- Not many people worry about RC extraction with variations today
  - Further studies are required in this area

Now, consider statistical modeling



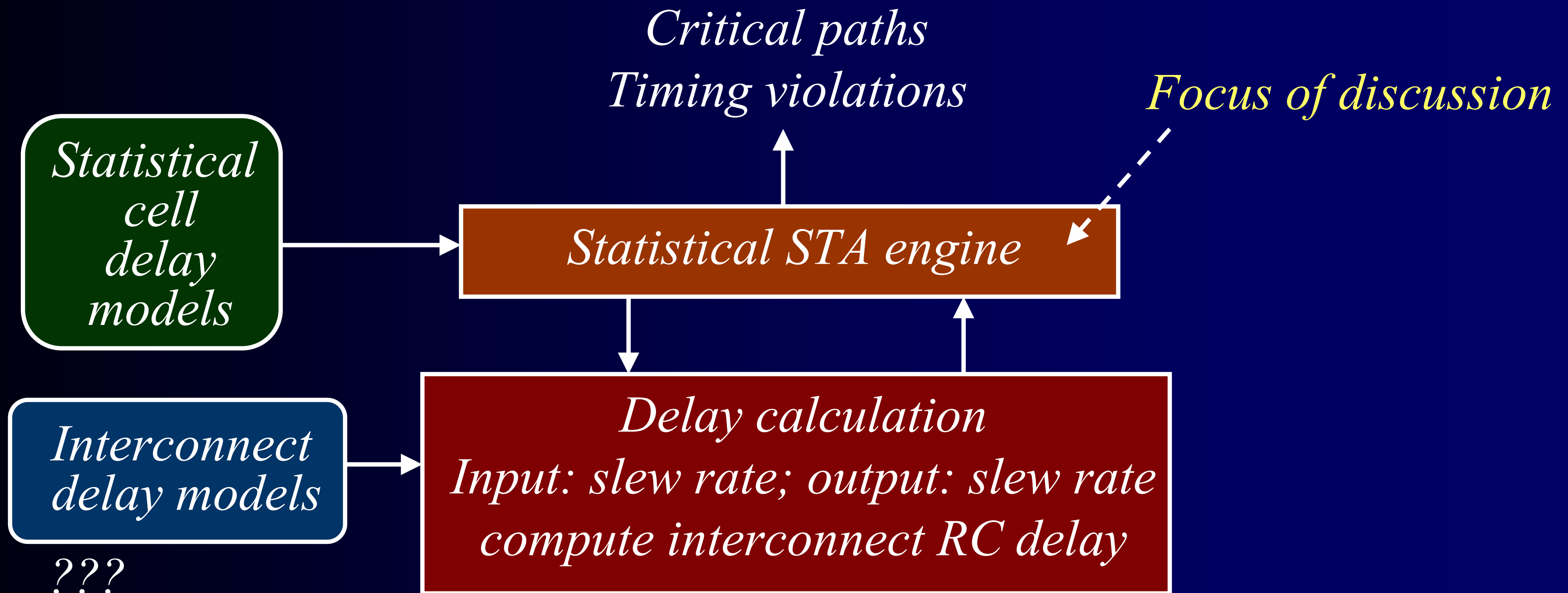
- Given statistical variations in the input space X (large dimension), derive variations in the output parameter space Y (small dimension) and the corresponding model M(Y)

Statistical analysis



- Given statistical variations in input parameter space X, approximate the statistical distribution on the output behavior of interest

Put together: Statistical STA



- Most techniques focus on SSTA engine
  - Assume a statistical cell model is available
- Modeling variations in interconnects and statistical delay calculation are under research
  - Usually, we can assume worst cases to begin with

STA vs. SSTA - motivation

Comparison on a large ISCAS benchmark

SSTA (0.25μm technology)

Mean 567.0

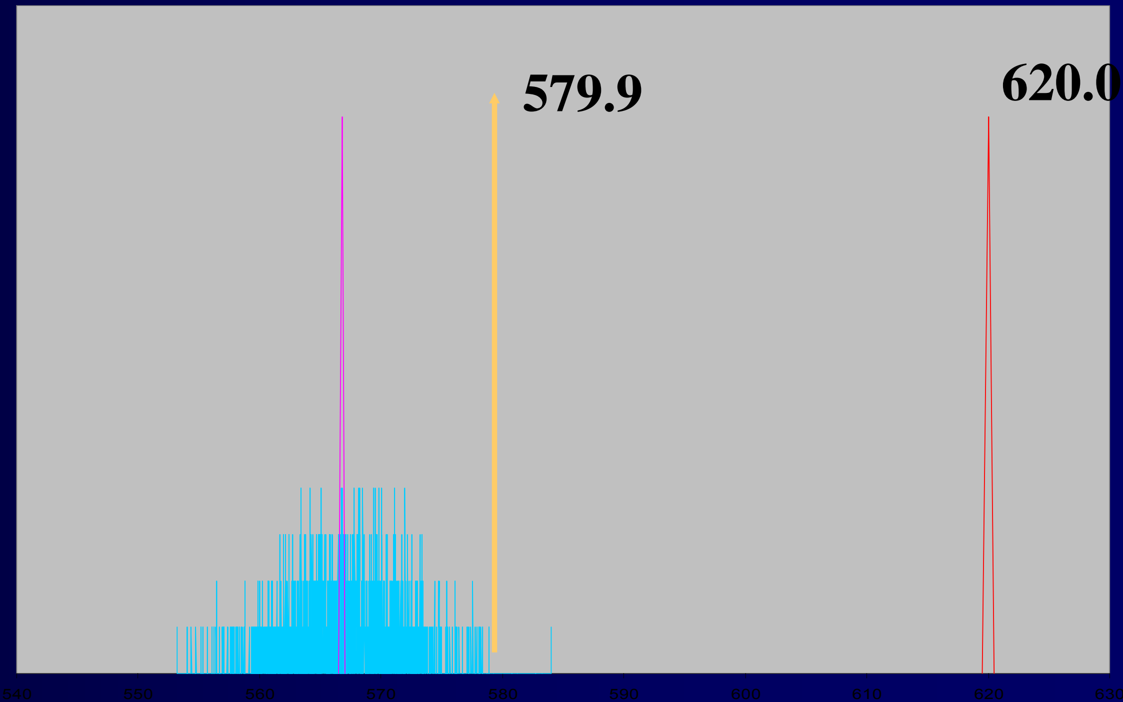
Std.dev. 4.29 (0.8%)

$\mu+3\sigma$ **579.9**

STA

mean μ_c : 566.8

worst case $\mu_c+3\sigma_c$: 620.0

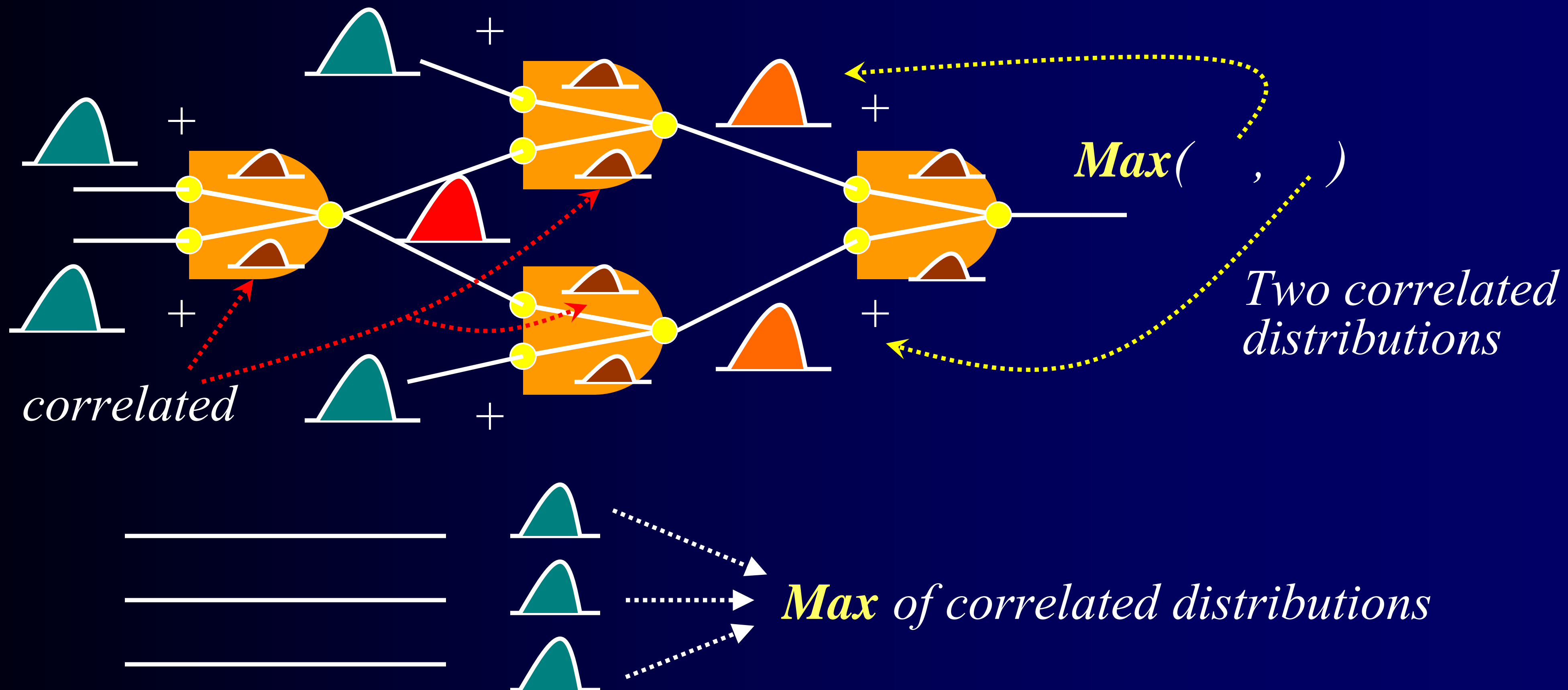


Worst case STA: $(\mu_1 + k\sigma_1) + (\mu_2 + k\sigma_2) = (\mu_1 + \mu_2) + k(\sigma_1 + \sigma_2)$

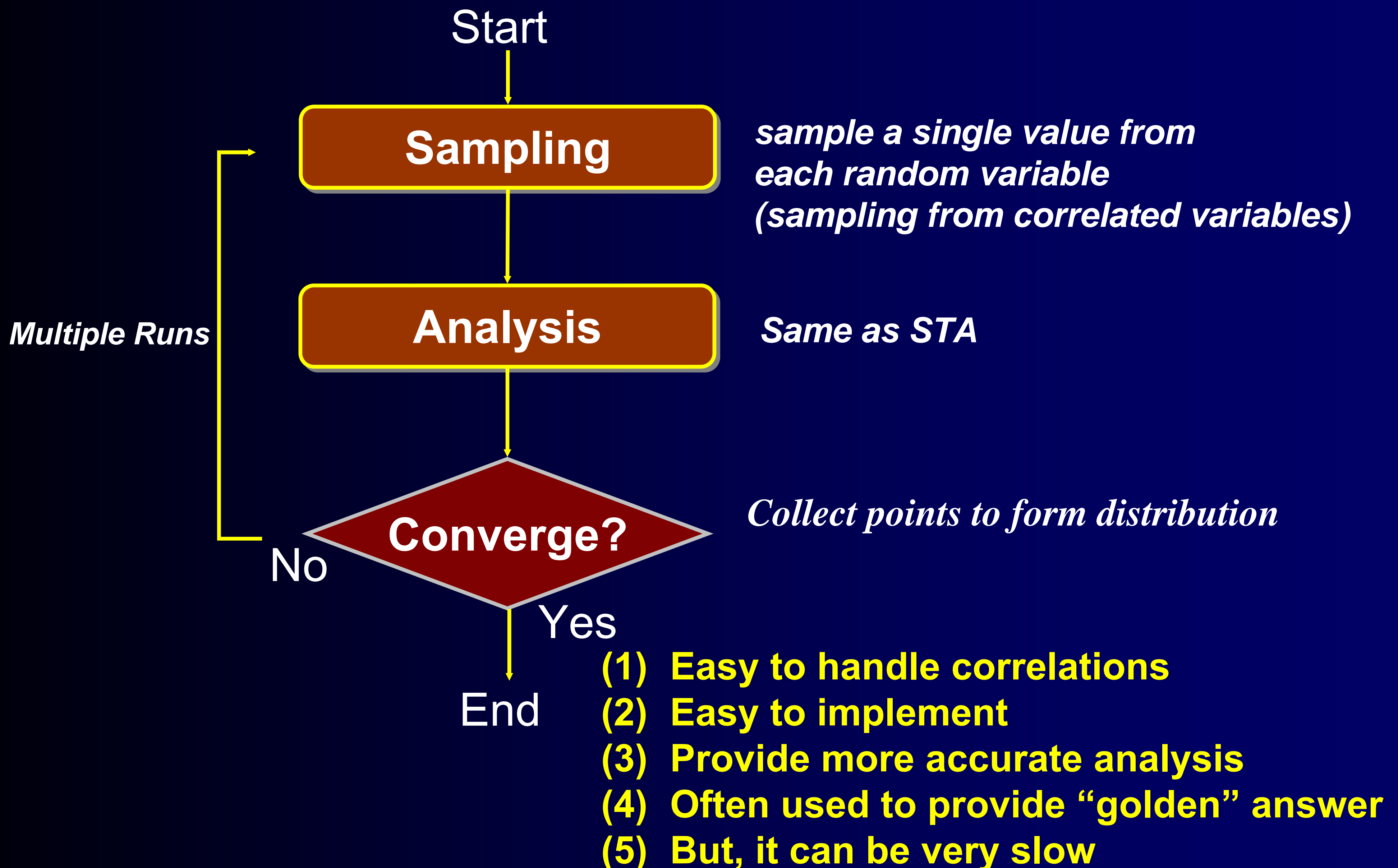
Smaller than

SSTA Convolution: $(\mu_1 , k\sigma_1) \oplus (\mu_2 , k\sigma_2) \Rightarrow (\mu_1 + \mu_2) + k\,(\sigma_1^2 + \sigma_2^2)^{1/2}$

SSTA engine – basic question



- The fundamental question is how to handle (perform +, max) correlated random variables
  - The assumption of Gaussian is no longer true after "max"
  - Same question for both block-based and path-based approaches


Simple way – Monte Carlo analysis
Start
Sampling
sample a single value from
each random variable
(sampling from correlated variables)
Multiple Runs
Analysis
Same as STA
Converge?
Collect points to form distribution
No
Yes
End
(1) Easy to handle correlations
(2) Easy to implement
(3) Provide more accurate analysis
(4) Often used to provide “golden” answer
(5) But, it can be very slow

SSTA approaches

- “Block-Based Static Timing Analysis with Uncertainty”, Devgan et al.
  - Won Best Paper Award ICCAD’03
- “Statistical Timing Analysis Considering Spatial Correlations Using a Single PERT-like Traversal”, Chang et al.
  - Presented at ICCAD’03 also
- “First-order Incremental Block-Based Statistical Timing Analysis”, Visweswariah et al.
  - Won Best Paper Award DAC ‘04
- Message at DAC05:
  - Statistical timing analysis is a hot topic!

IBM: Parameterized Block-Based SSTA (DAC04)

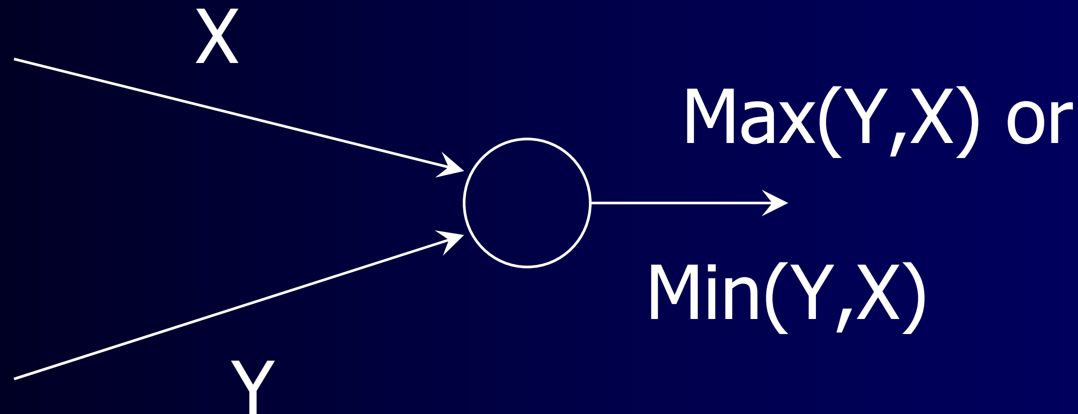
- Path-based analysis
  - Select a set of paths first and analyze those paths only (guard-band)
  - The problem is simpler (nXn correlation matrix)

- Block-based analysis
  - Like breadth-first search (level-by-level analysis)
  - Analyze the timing graph
  - They define a *canonical delay form* and propagate this form through the circuit

IBM: Parameterized Block-Based SSTA

- Delay = $a_0 + a_1X_1 + a_2X_2 + \ldots + a_nX_n + a_{(n+1)}R_a$
  - All delays are represented as the canonical form
  - All a's are constants, representing the sensitivity to variations
  - All X's are random variables, each X representing an unique independent source of variation effect
  - R_a: the random noise
- Key: given two input delays represented as the above form, how to compute the output delay represented as the above canonical form?
  - If we can do that, this approach can then handle arbitrary correlations among random variables (big plus!)
  - Doing "addition" is straightforward (why?)
  - The only issue is doing "max"

For calculating MAX(X,Y)



- Tightness probability
  - Prob(X>Y)
    - (Prob(max(X,Y)=X))
    - X dominates the delay at the output
  - Prob(Y>X) = 1 – Prob(X>Y)
  - MAX(Y,X) = X Prob(X>Y) + Y Prob(Y>X)

  - Given X,Y in canonical form, calculate the output as the max/min of X,Y and also represent the result in canonical form
  - C. E. Clark (Operations Research, 1961, pp. 145-162)
  - Jess, et. al. (IBM paper in DAC 2003 on the same topic)

SSTA in DAC 05

- Hongliang Chang, et al.
  - Canonical representation for non-linear, non-Gaussian parameters
- Yaping Zhan, et al.
  - Correlation-aware, non-Gaussian distributions
- Lizheng Zhang, et al.
  - Correlation-preserved, non-Gaussian distribution with Quadratic timing model
- Aseem Agarwal, et al.
  - Statistical gate sizing with SSTA
- Vishal Khandelwal, et al.
  - Taylor-expansion polynomial-representation based SSTA
- Conclusion: If you want to go into non-linear and/or non-Gaussian, you need to pay a large amount of computational overhead

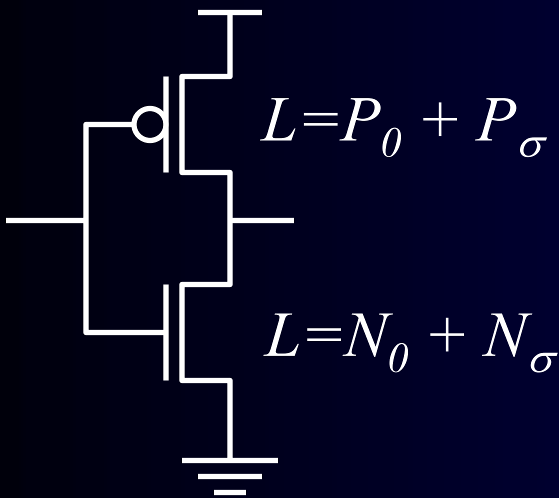
Simplified SSTA

What SSTA can buy us?

Simplified SSTA

- C. S. Amin et. al. "Statistical static timing analysis: How simple can we get?" DAC05
  - Based on Intel CAD flow
- Highlights
  - Model channel length, Vth variations
  - Decompose into random and systematic variations
  - Random variations die out on path delay
  - Systematic variations dominate
  - Max operation can be simplified
  - Clock variation and path delay variation track together because of systematic variations and hence should be analyzed together to give more margin

Variation modeling



$$Delay\ D = g(P_0, N_0) + f(P_\sigma + N_\sigma)$$

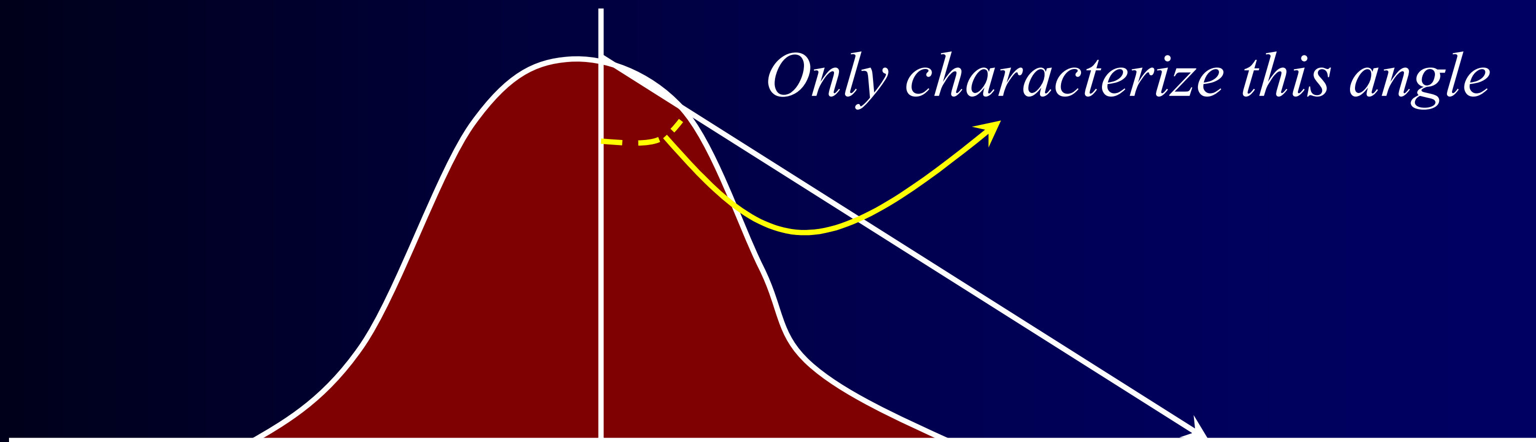
$$= D_0 + \partial D/\partial P_\sigma (\Delta P_\sigma) + \partial D/\partial N_\sigma (\Delta N_\sigma) + \ldots$$

$$\approx D_0 + A_p (\Delta P_\sigma) + A_N (\Delta N_\sigma)$$

For example, characterize $A_p = (\Delta D - D_0) / \Delta P_\sigma$

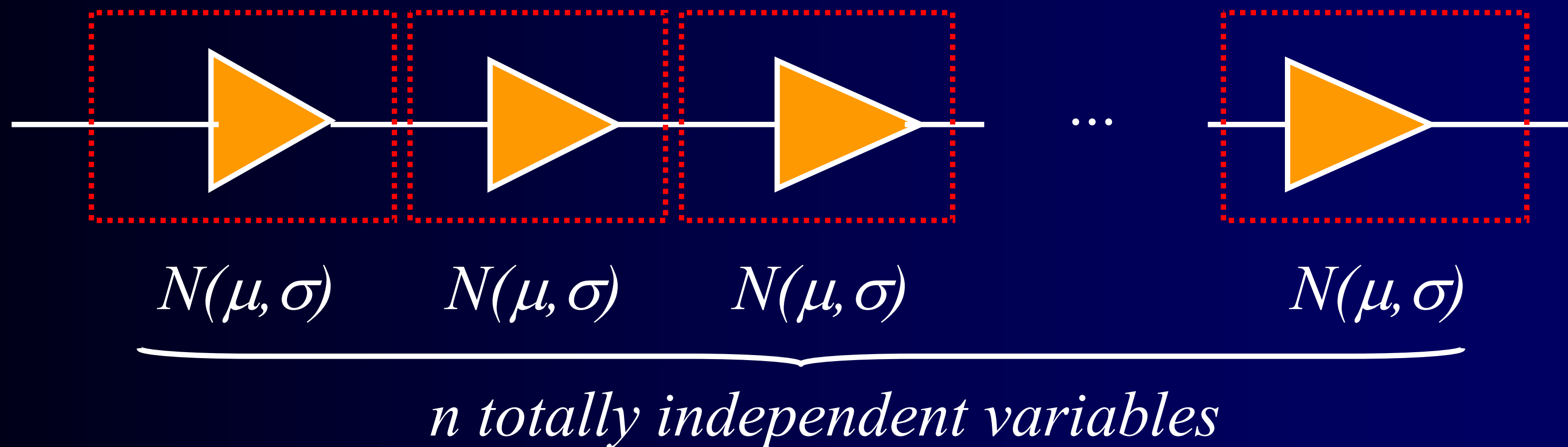
- Characterization flow
  - Compute nominal delay D_0 with nominal parameter value P_0
  - Change P's channel length L from P_0 to $P_0 + \Delta P_\sigma$ and measure the delay change ΔD
  - Compute the coefficient $A_p = (\Delta D - D_0) / \Delta P_\sigma$
  - They can call this "linear sensitivity method"

Sensitivity model can be overly pessimistic



- Linear sensitivity model usually is not good to capture the shape of the distribution
- It can be an optimistic or pessimistic model

Random variations die out on a path

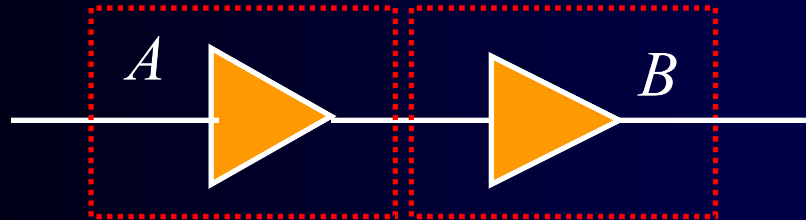


$$\% \text{ of path delay variation} = \mu_{path}/\sigma_{path} = (n\,\sigma^2)^{1/2} / (n\,\mu)$$

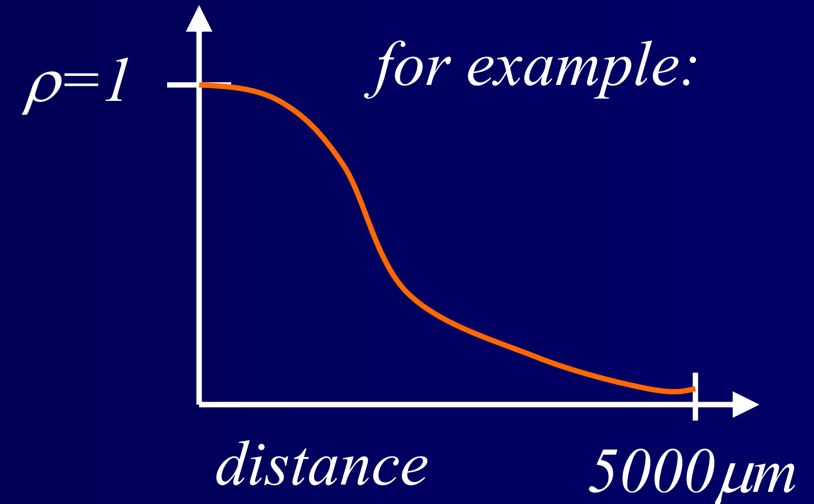
$$= (1 / n^{1/2})\,(\mu / \sigma) = \mathbf{(1 / n^{1/2})} \;*\; (\% \text{ of cell delay variation})$$

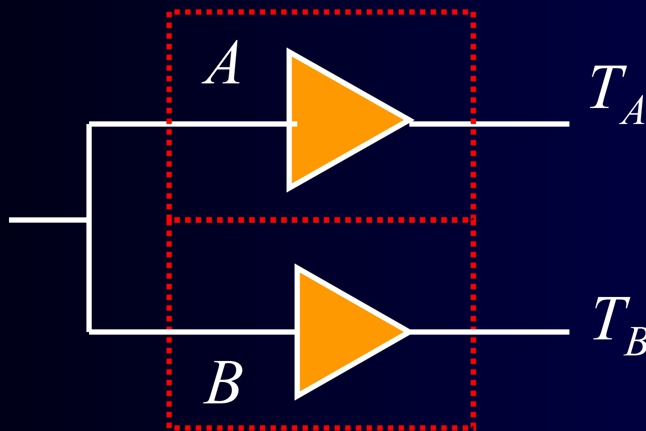
- For n = 10 (10 stages), $(1/n^{1/2}) = 0.316$
- As # of stages in a path increase, random variations in cells become less important
  - We only need to worry about systematic components

Systematic variation



$$\sigma_{path}^{2} = \sigma_{A}^{2} + \sigma_{B}^{2} + 2\rho_{AB}\,\sigma_A\,\sigma_B$$



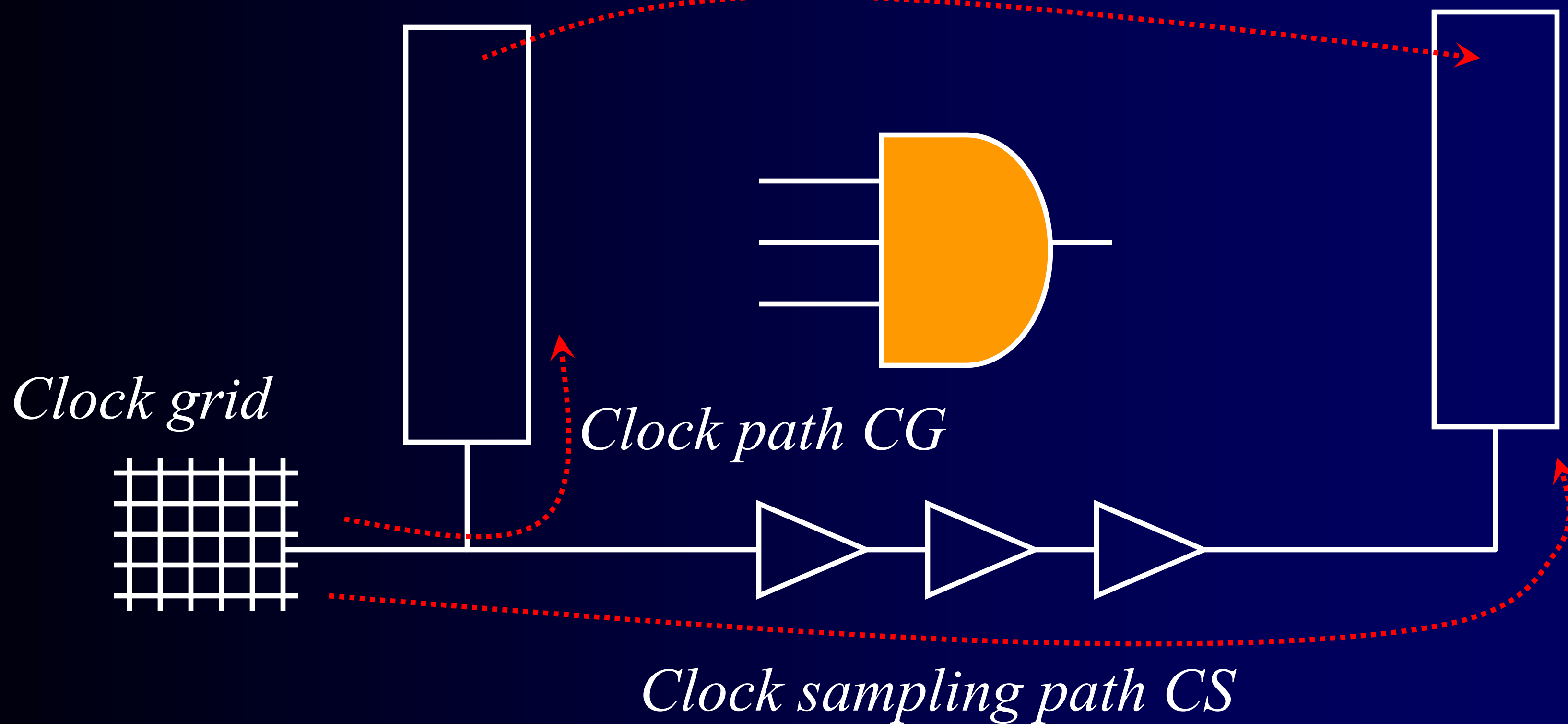


$$\sigma_{TA-TB}^{2} = \sigma_{A}^{2} + \sigma_{B}^{2} - 2\rho_{AB}\,\sigma_A\,\sigma_B$$

Variance increases as distance increases

- High correlations among cells and paths that stay closer to each other
- Clock path and delay path stay closer to each other
  - They should be analyzed together

Clock path and delay path



- $\sigma_{margin}^2 = \sigma_{CS}^2 + \sigma_{CGD}^2 - 2\ \text{covariance}\ (T_{CS}, T_{CGD})$
- Additional margin can be bought out due to systematic variations

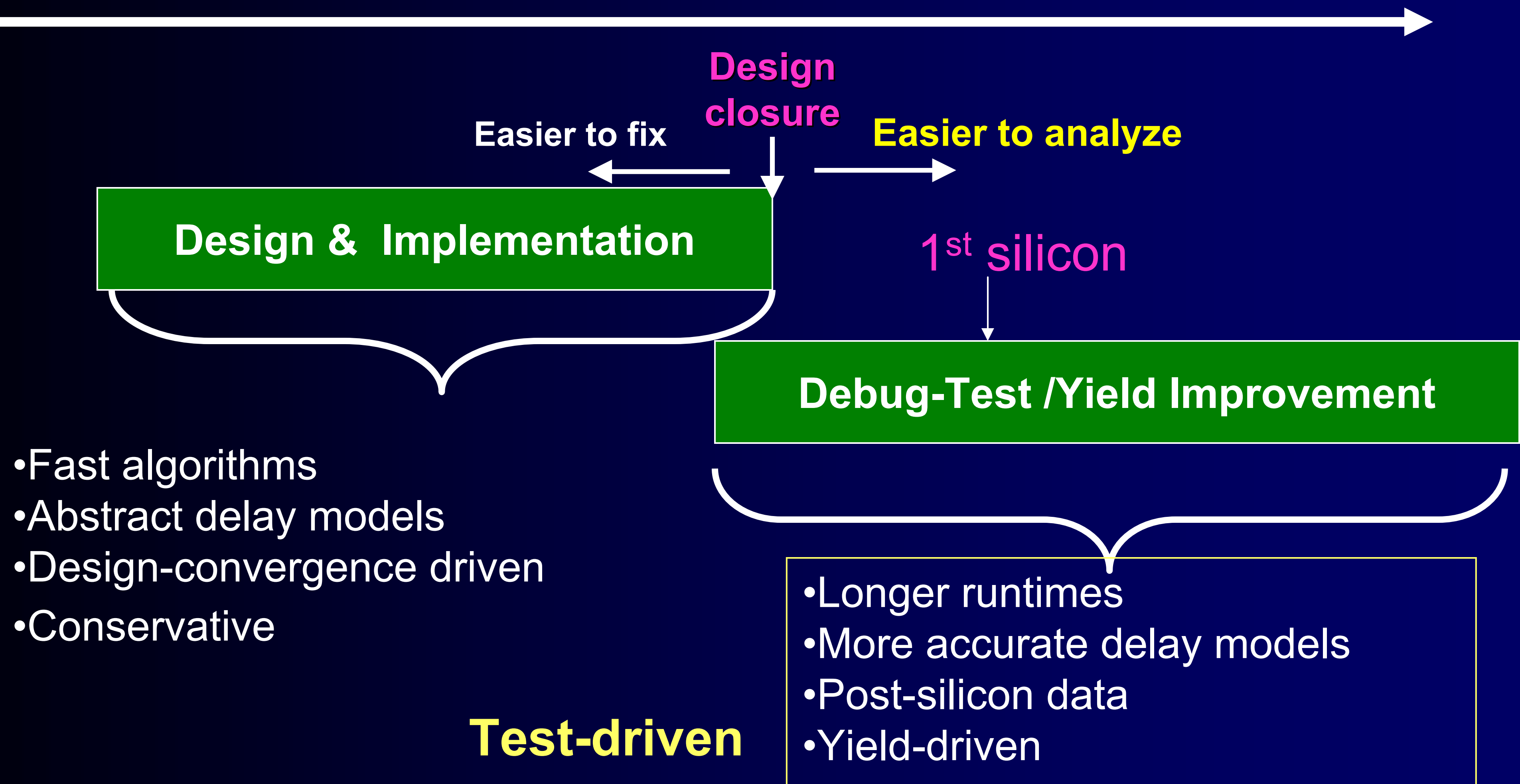
Summary

- The simplified SSTA was applied to
  - A large microprocessor block (> 100K cells)
  - Based on 90nm technology
  - Analyze 492 most critical paths
- Error in computing standard deviation of the margin is on average only 0.19% of path delay
- Only a few paths show up as the most critical paths on 600 samples
- Ordering among paths, decided by a fixed-value STA, does not alter much by either random variations or systematic variation
  - Random variations die out
  - Systematic variations make paths within a block track each other well

Pattern-based statistical timing analysis

Test-driven (Noel Menezes, VTS05)

Product development timeline

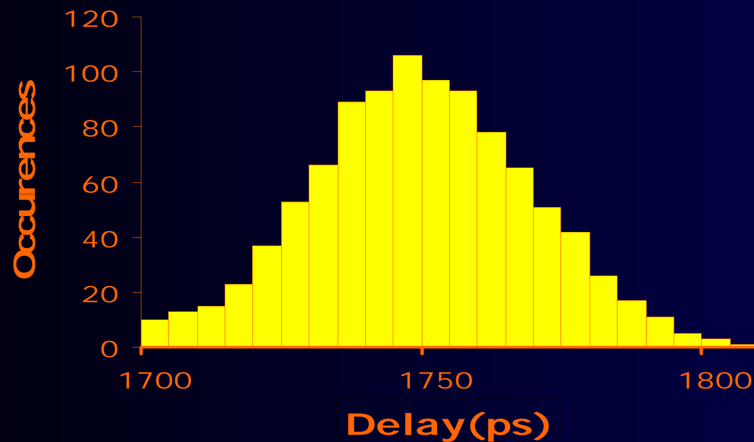


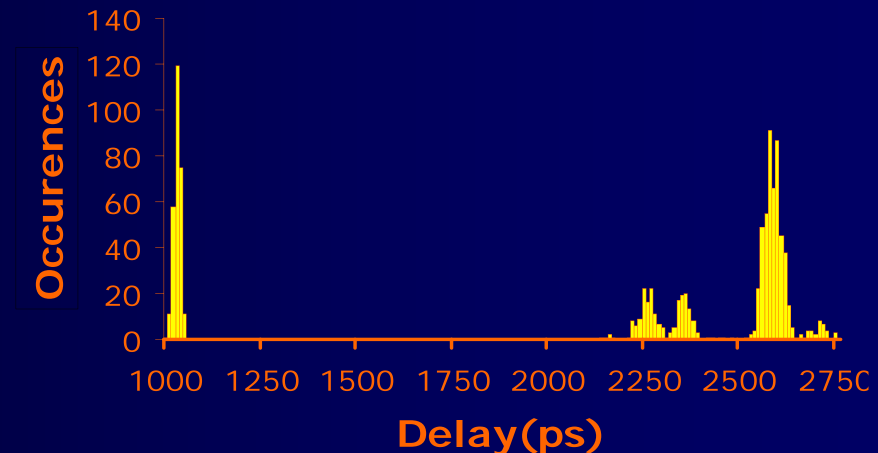
Pattern-based Statistical Timing Analysis

- What the tool does: Given a 2-timeframe pattern, estimate its delay distribution as **(mean, σ)** based on given a timing model
  - Benjamin Lee et. al. VTS05, ITC05
- Among many challenges, one difficulty lies in the fact that a pattern may sensitize different sets of paths on different dies
  - **Hazards** may be present on one die but not another
  - Overall delay distribution becomes multi-modal
- Let's look at the Monte Carlo simulation results …

Pattern delay distributions

- Delay distributions of two different patterns
  - Result from Monte Carlo simulation of 1000 samples





- Pattern 1: Near **normal** distribution
  - Same path dominates on all dies
- Pattern 2: Multi-modal – **non-normal** distribution
  - Hazards sensitize different paths on dies

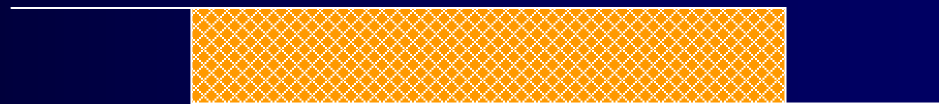
Uncertainty window analysis

Concept from ITC'04, Krusemen et. al

- Duration when signal can be 1 or 0

Full waveform

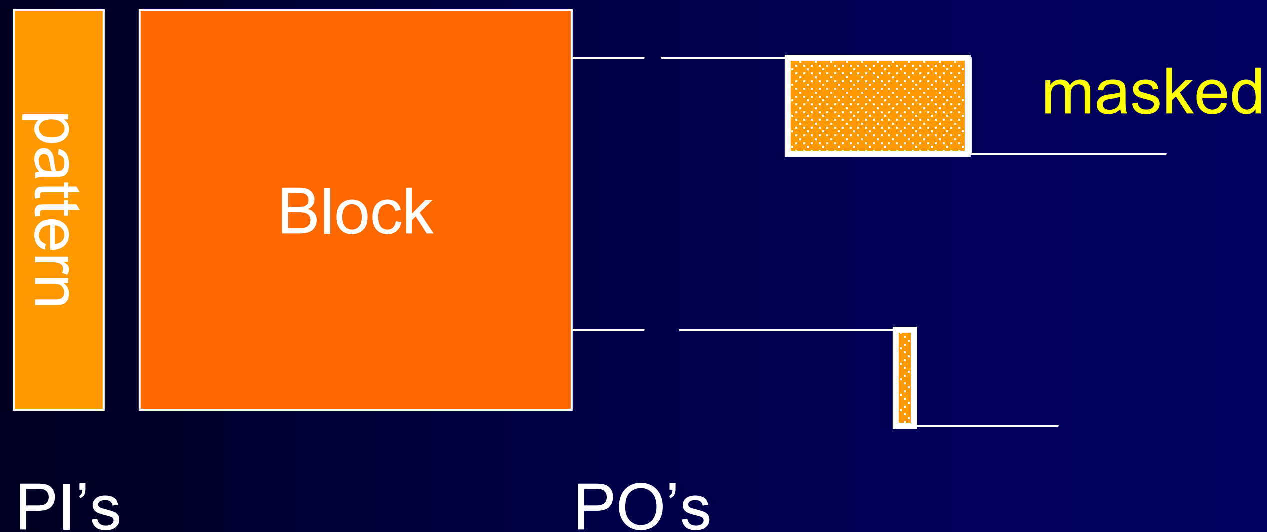
Uncertainty window waveform



- In our case, we just need to quantify uncertainty window as a random variable (in addition to quantify the pattern delay random variable)

Masking uncertain POs

- For each pattern
  - Propagate arrival time/uncertainty window r.v.'s from PI's to PO's
  - Mask primary outputs that have uncertainty window width above threshold (-3σ lower bound)



Hazard-aware run-time

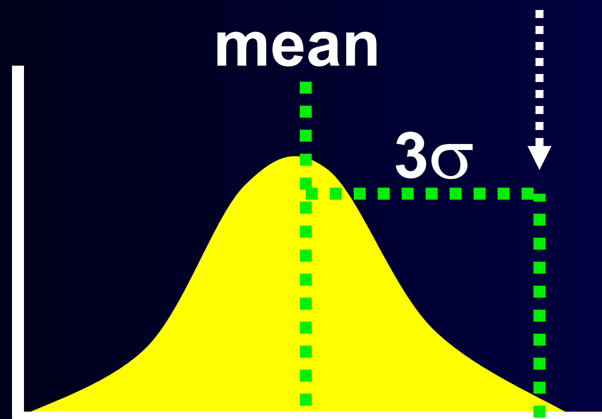
Runtime (**secs**) for 15-detect transition fault set

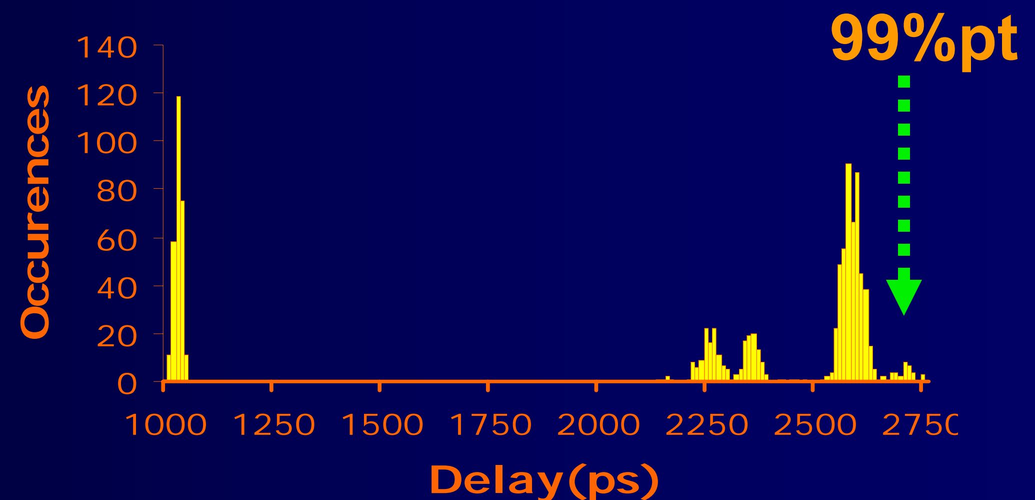
	C432	C880	C2670	c5315
Nominal	**.77**	**4.65**	**25.56**	**19.24**
MonteCarlo	**754**	**4993**	**23582**	**17284**
PB-STA	**1.19**	**9.56**	**60.97**	**59.84**
Hazard aware	**5.93**	**20.91**	**161.92**	**167.98**

- 6-10x slower than nominal

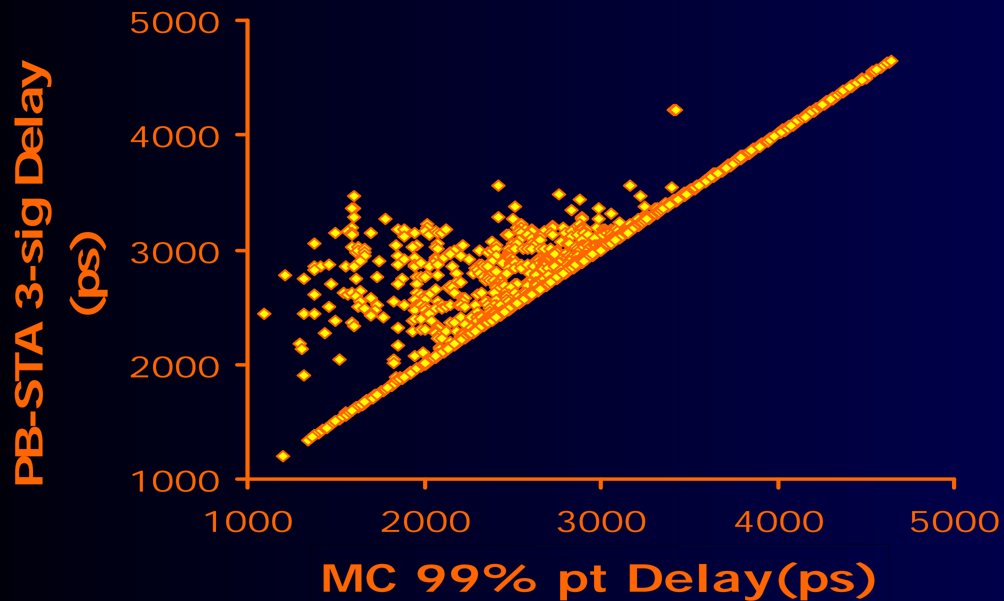
Comparing hazard-aware PB-STA to Monte-Carlo

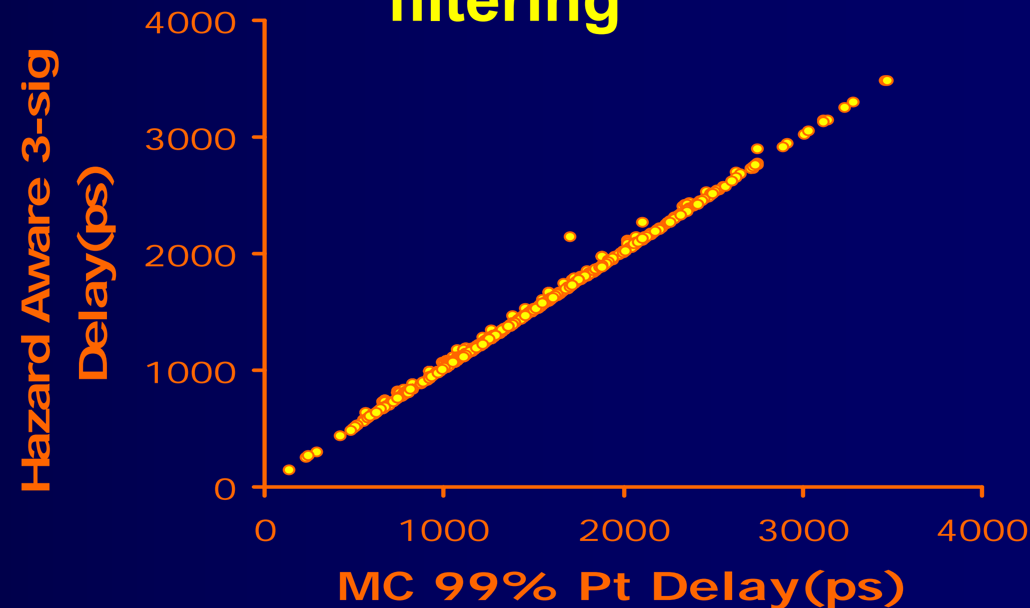
- 99% pt is delay that is greater than 99% of Monte-Carlo samples
- Compare 3σ delay point to Monte-Carlo 99% point to assess PB-STA's accuracy





Results after hazard-based analysis

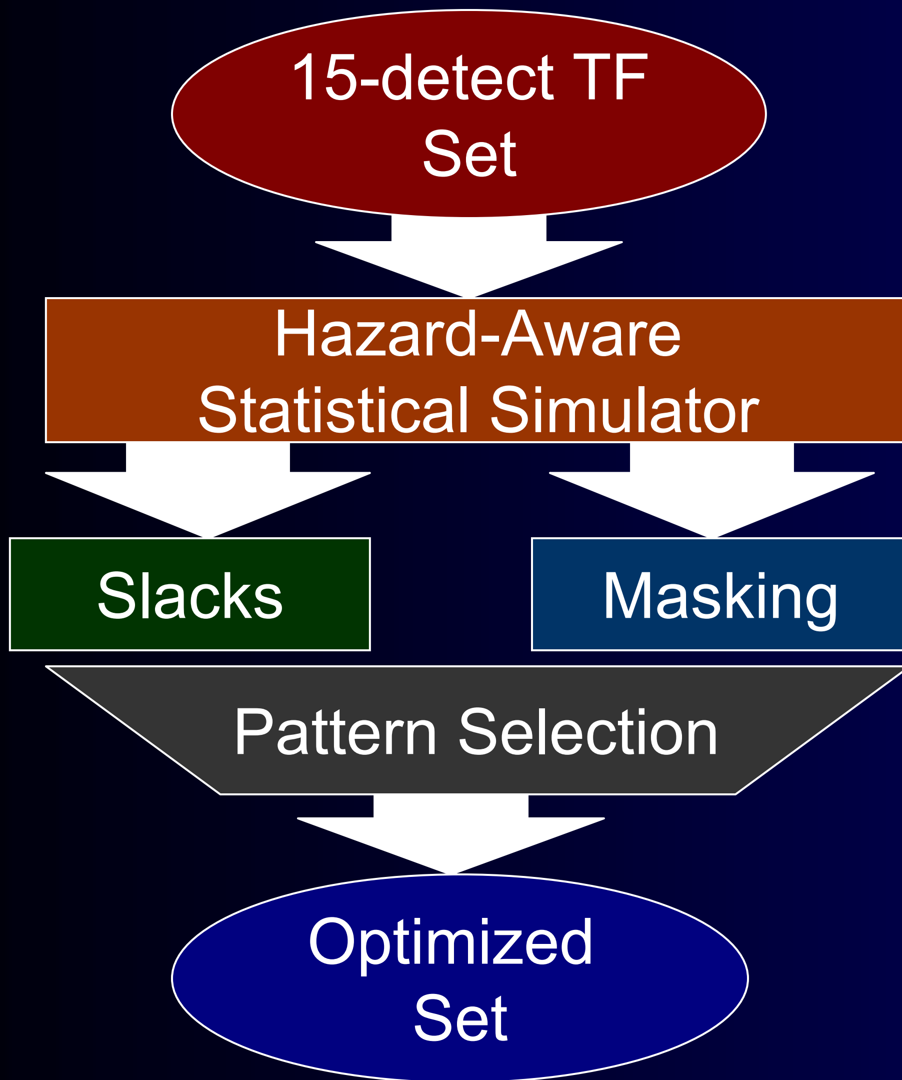




– Hazard-aware improves accuracy (Lee, et al. ITC05)

– Facilitates development of pattern selection methods

Application – pattern selection/filtering

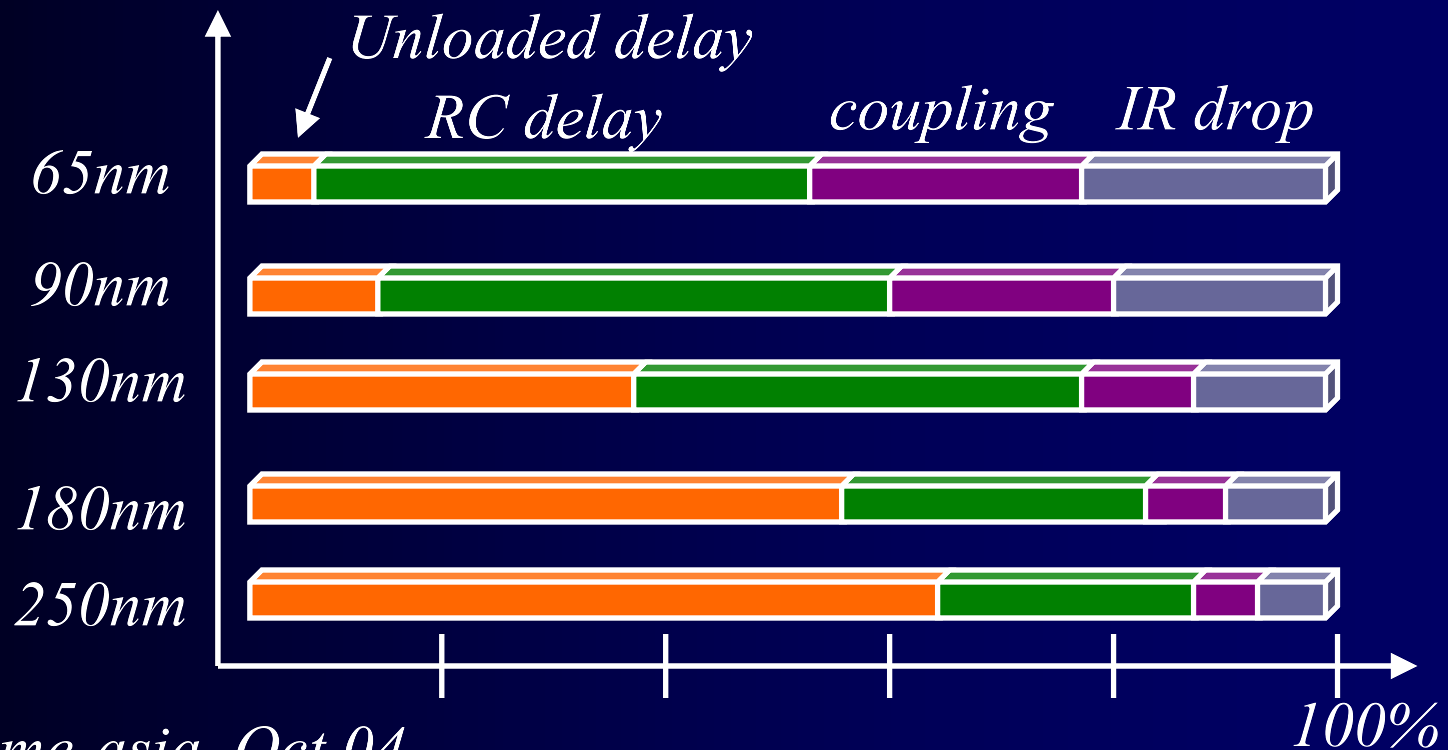


- Used 15-detect TF as a superset
- Forward/reverse propagation - for masking/ slack info
- Goal: reduce set size, maintain delay defect capture capability
- See Lee ITC 05

Break 5 minutes for questions

Next, we will switch topic on
DSM timing effects

Growing parasitic effects



G. Bell, eetime-asia, Oct 04

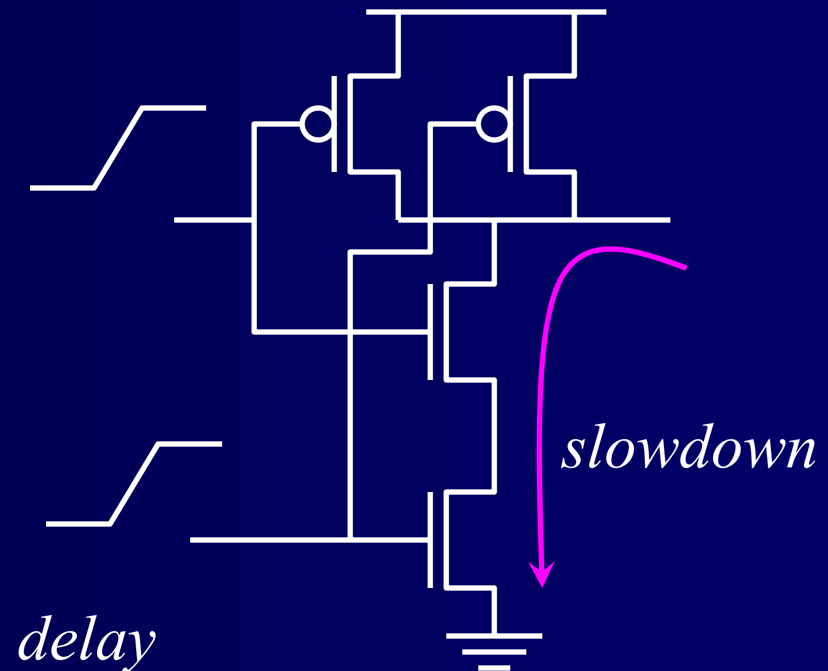
- RC delay, coupling and IR drop become dominating for delay
- Coupled with variations, this complicates timing analysis

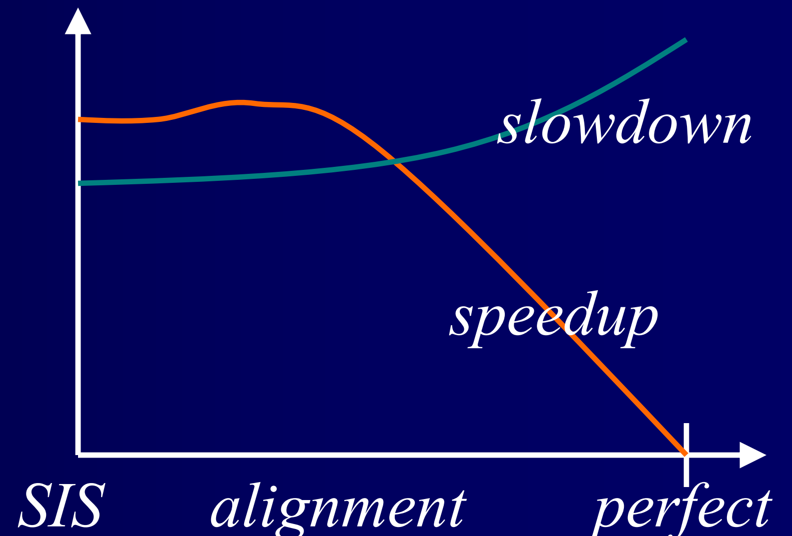
Considerations in timing analysis

- Process variations
  - Inter-die and intra-die process variations
    - We spent a great deal of time to talk about it already
- Noise and signal integrity
  - Cross coupling
  - Power noise/IR drop
  - Interconnect RC
    - In general, hard to model and calculate exactly
    - Variation modeling for interconnect is also an issue
  - Inductance noise
    - Usually impact long buses
- Modeling issues
  - Multiple input switching (MIS)
    - Cell macro-modeling issue; will talk about it here
  - Waveform model
    - Ramp model may not be accurate to describe the actual waveform

MIS

- Comparing to single input switching (SIS) delay
  - MIS causes slowdown at series stack of transistor
  - MIS causes speedup at parallel stack of transistors
- Delay effects
  - Speedup percentage is usually much larger than slowdown percentage





General thinking

- The probability of signal alignment is diminishing after passing through a few stages of gates
  - Therefore, most MIS effects occur at the gates closer to the launching latches
- MIS affect short paths more severely than long paths
- Need to check hold time violation (minimum delay) carefully
  - Speed up amount is greater than slowdown amount

General approach - filtering

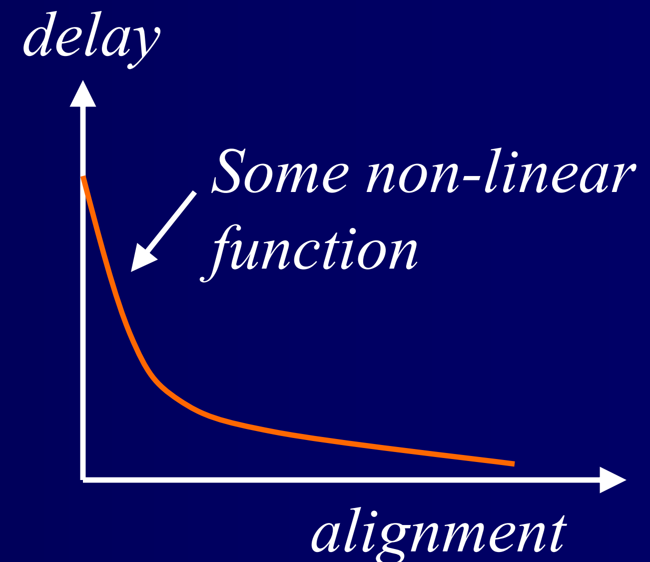
- Because MIS may not occur often, we usually take a filtering approach to rule out gates or cells that MIS are impossible to happen
- Filtering based on worst-case timing windows from STA
  - If time windows of two signals do not overlap at all, we say that MIS cannot happen for these two signals
  - We need to pursue an *iterative algorithm* until STA results converge, because if timing windows do not overlap, we need to update the gate's output delay and propagate the change to all downstream gates whose delays are affected
- Adding statistical process variations in the analysis
  - See Agarwal, A.; Dartu, F.; Blaauw, D.; DAC 04, pages:658 - 663

Crosstalk

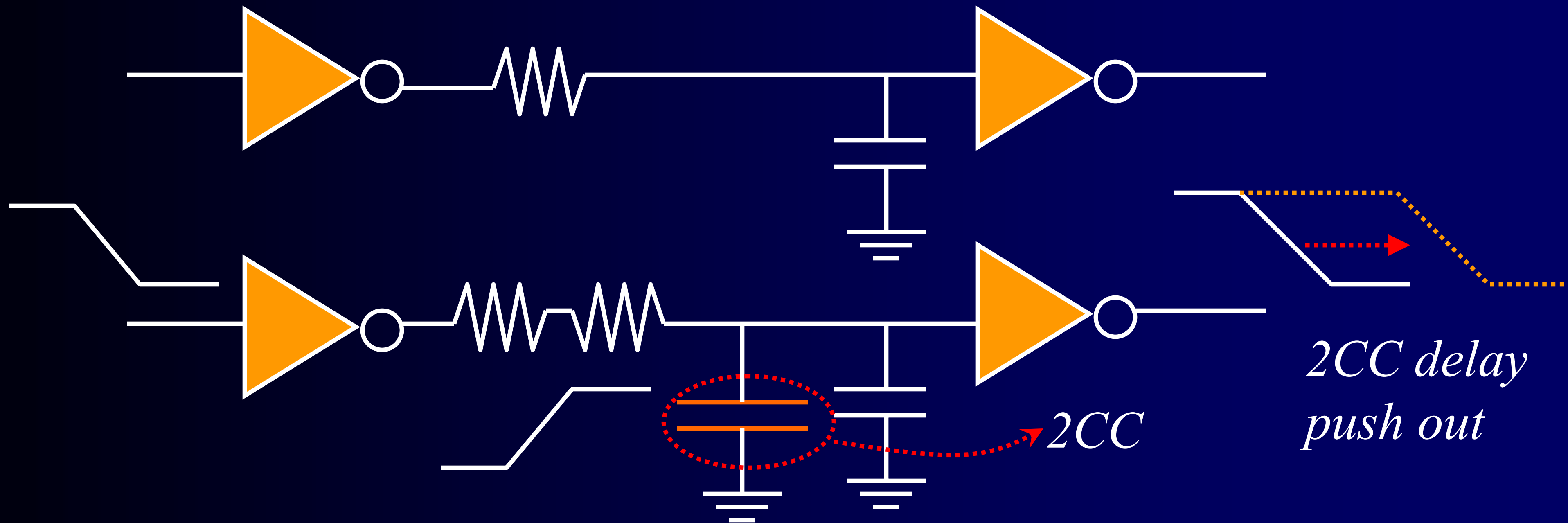
CC

Delay push out

- Delay push out can be up to 80% of the path delay
- The amount of delay change depends on the signal timing alignment at the two coupled wires



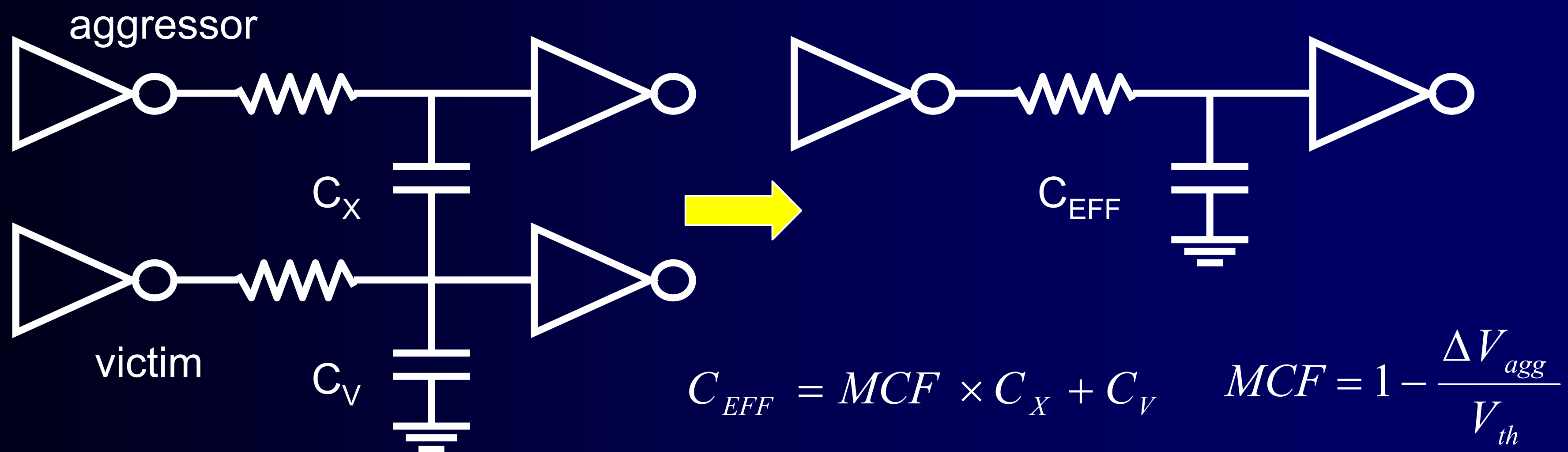
Basic model



- Historically, people use switch factor 2 multiplying the coupling capacitance as the worst case
  - Use 2CC factor to perform worst-case STA
  - In general, this gives very pessimistic results
- On a single stage 2CC may not be the worst case

Miller Factor

- The use of switch factor is popular
  - If 2 is too much, people can use a number SF = [0, 2] such as 1.5
  - Typically, complete waveform accuracy is not required for crosstalk aware static timing analysis because we only want to *bound* the delays
- Miller Capacitance Factor – a more sophisticated switch factor
  - Assumes equal charge transfer and $V_{th} = 0.5V_{DD}$, MCF = [-1, 3] from 0% to 50% transition
  - ΔV_{agg} = amount of voltage change in aggressor signal while victim transitions from 0 to V_{th} or from V_{DD} to V_{th} (assuming $V_{th} = 0.5V_{DD}$)



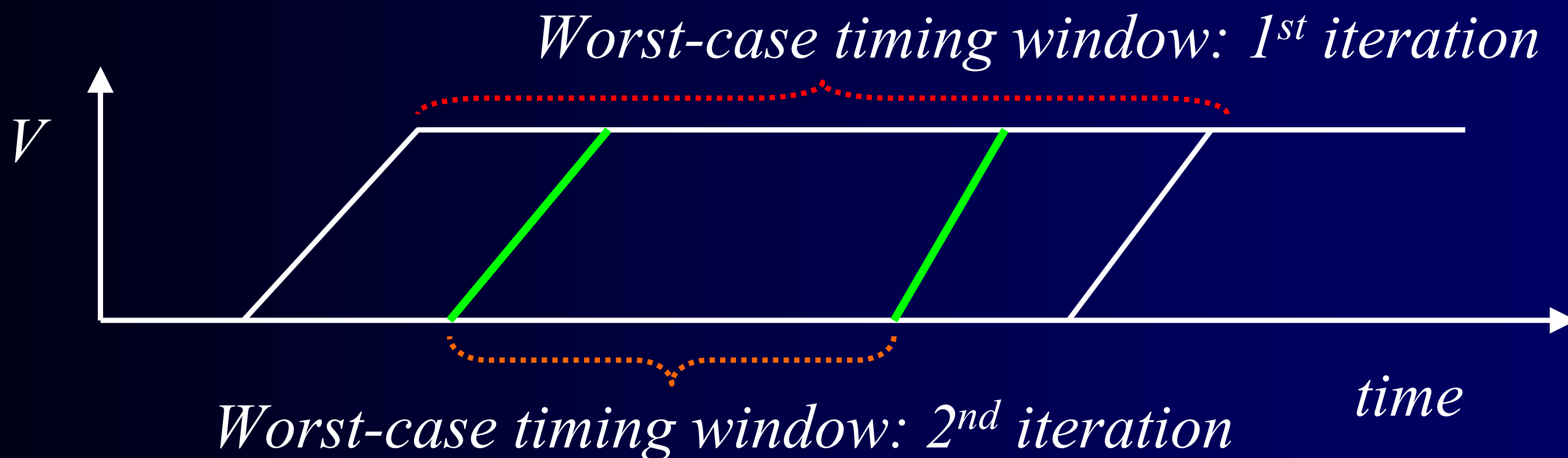
$$C_{EFF} = MCF \times C_X + C_V \qquad MCF = 1 - \frac{\Delta V_{agg}}{V_{th}}$$

Other models

- T. Sakurai TED 1993
  - Derives closed form equations to model the waveform of an RC line
- J. Qian, S. Pullela, L. Pillage TCAD 1994
  - Derive new model for effective capacitance, because others have ±10% error, and optimism is generally unacceptable
  - Introduce π-model to separate the capacitive element into 2 elements, one before and one after the resistor
- H. Kawaguchi, T. Sakurai ASP-DAC 1998
  - n-line coupling capacitance equations without victim and aggressor relationship
- A. Kahng, S. Muddu, and D. Vidhani ASIC/SOC 1999
  - Extend π-model by separating the resistive element into 2 elements, one before the π, and one in the π
  - Done to reduce the over pessimism and over optimism of SF

STA with crosstalk (TACO: DAC 00)

- Like MIS, the way to deal with crosstalk in STA would also be following a filtering approach
  - Start by assuming the worse case
  - Iterate the following two steps until converge
    - ✓ Based on the timing windows calculated so far, identify those aggressor-victim pair whose coupling capacitance should be smaller than that calculated in the previous iteration
    - ✓ Re-calculate (shrink) the timing windows based on the reduced coupling capacitances

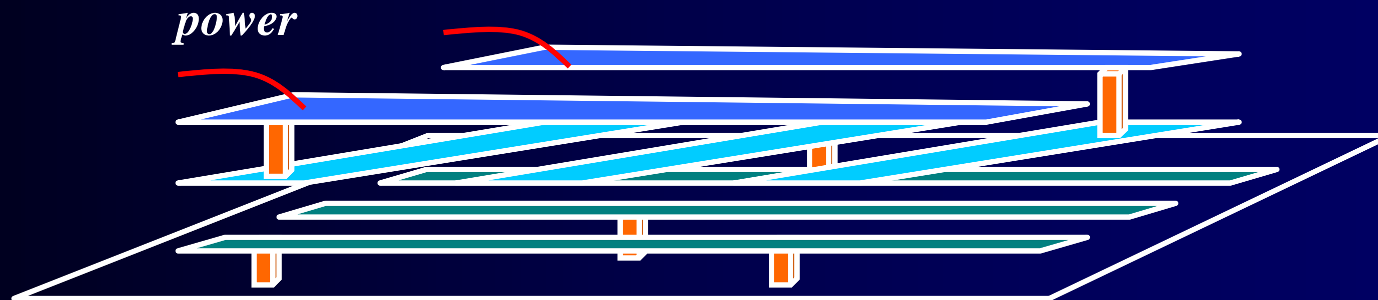


More recent crosstalk-aware STA examples

- K. Agarwal, Y. Cao, T. Sato, D. Sylvester, C. Hu ASP-DAC 2002
  - Instead of using timing windows, proposes a noise-aware STA
  - Crosstalk overlap could be caused by noise instead of just timing windows

- D. Sinha, H. Zhou ICCAD 2005
  - Statistical timing analysis to consider crosstalk and MIS

Power noise



- Typically, power distributes from the top layer down to devices through metal lines and Vias
- Trends:
  - Supply voltage decreases
  - Device threshold voltage lower
  - Circuits are more sensitive to noise tolerance
  - Adaptive voltage control becomes more popular
- When circuits switching, current flows from power bus - or into ground bus
  - $dV = IR + L\, dI / dt$
  - Effect can be split into IR drop effect and inductive dI effect

Dealing with power noise

- If want to accurately characterize power-induced timing effects, the essential problem is how to simulate both the power grid and the non-linear switching circuit together
  - Timing and power affect each other
  - This can be too complex (time-consuming)

- In practice, consider one independent of the other
  - For power-grid analysis, circuit is abstracted into time-varying current sources
  - For circuit simulation, the power supply variation can be abstracted to worst-case bounds of voltages
  - So the idea is (1) extract power map (2) STA with the map

Power grid analysis

- Model power-grid as a RLC network
  - Circuit abstracted into time-varying piecewise-linear current sources
  - Simulate circuit with the ideal power grid to obtain current profile
- Modified Nodal Analysis (MNA) is used to solve for power grid node voltages
- Converts the problem into solving a sparse, symmetric-postive-defintite linear system
  - $G\,x(t) + C\,\partial x(t)/\partial t = b(t)$
  - G: conductance matrix
  - C: admittance matrix due to C,L
  - x(t): time-varying vector of voltages at nodes
  - b(t): time-varying current sources

IR drop and dI/dt noise

- IR drop
  - Usually refers to decrease/increase in power/ground rail voltage due to resistance of devices between rail and a node of interest
  - Common practice is to budget a max-per-rail static voltage drop tolerable
  - Static IR-drop can be calculated from extracted parasitic / average power consumption - (DC analysis)
  - Dynamic-IR drop- require vector based analysis

- dI/dt noise
  - Inductive dI/dt noise used to occur mostly on package
  - On-chip interconnect's impedance is no longer ignorable due to higher frequencies
  - Change in current (dI)
    - Simultaneous switching – big current swing

Various studies

- H Kriplani, FN Najm, IN Hajj, IEEE TCAD '95
  - Linear time algorithm: finds upper-bound estimate of current wave-forms at all contact points
- HH Chen and David Ling DAC '97 (cited by 111)
  - Describes models used for power bus / switching circuits/decoupling capacitors
- H.H. Chen and J.S. Neely, IEEE Transactions on Components, Packaging and Manufacturing Technology, Aug 1998
  - Analyze IR drop and inductive dI/dt noise
  - Notes: worst-case dI noise and worst-case IR drop do not occur at same time
  - Power-supply distribution model
  - Switching-circuit model

Various studies

- Yi-Min Jiang, K-T Cheng, An-Chang Deng, ISLPED 98
  - Genetic-algorithm approach to generate patterns
  - Estimate IR drop and dI noise based on charge/discharge current cell library
- Yi-min Jiang, K-T Cheng, DAC '99
  - Statistical model derived by simulating characterization patterns
    - Use GA search to find patterns (last paper)
    - Find average voltage for each cell for each pattern - average voltages form distribution
- A. Dharchoudhury, et al, DAC 98 (based on PowerPC)
  - Describes methodology for power supply design/analysis
  - IR-drop analysis is discussed
    - Transistor level is infeasible
    - OTS blocks (standard cells) macro-modeled as current source
    - Each block has an IR-drop budget (voltage drop)
    - If budget violated, power grid that supplies block is augmented
- P. Larsson, IEEE Custom Int. Circuits Conf 1999
  - Describes noise suppression techniques
  - Makes some predictions for the future based on process parameters

Various studies

- Sani Nassif, Joseph Kozhaya, DAC 2000 (fast simulation)
  - PDE-like multi-grid method for simulation of power grid (computation wire, not macro-modeling)
  - Circuit abstracted as time-varying current sources
  - Grid-reduction technique
- M.Zhao, et al DAC 2000 (Hierarchical analysis)
  - Difficulties in power network analysis:
  - Network is huge, typically 1-100 million nodes
    - ✓ Sparse linear system solution methods: conjugate gradient
  - Network is nonlinear due to switching devices
    - ✓ Solution: simulate individual blocks without power network, then simulate power network using time-variant current profiles
  - Speed-up proposed:
    - ✓ Macro-model local power grids
- J. Saxena, K. Butler, V. Jayaram, et al, ITC 2003
  - Structural-tests have a lot of switching activity
    - ✓ Worst-case seario for IR-drop
  - Analyzed chips - increased switching activity with structural test induced IR drop caused failure

Various studies

- D. Kouroussis, Rubil Ahmadi, Farid Najm, DAC 2004
  - Abstract circuit in terms of current constraints (peak current constraint)
  - Use a upper/lower bound of supply variation
  - Extract critical paths
  - Verify that voltage of critical paths are within bounds
  - Solve for max. delay of paths given current constraints

- Jing Wang , et al. VTS ‘05
  - Power region model
    - ✓ Assume supply voltage within a region is uniform
    - ✓ On-chip Ldi/dt drop is neglected
  - Switching Model
    - ✓ Triangle/Trapezoid current model
    - ✓ Gates see constant average Vdd

Break 5 minutes for questions

Next, we will switch topic to studies of speed binning

Study: correlating structure test to functional test

- Motivations
  - Examine the correlation between the frequencies measured using various structural testing and functional testing
  - Investigate structural testing as an option for speed binning
    - Reduce tester cost for speed binning
  - Reduce the cost of testing delay defects

Functional Testing

- Utilization of functional vectors for frequency measurement and speed binning is the industry norm
  - Long simulation time for development
  - Expensive, high performance testers needed
  - High degree of timing and edge accuracy during at-speed application
  - Fails are hard to debug

Structural Testing

- Structural testing provides an attractive complementary/alternative solution
  - Relaxed speed and accuracy requirements on the external pins
  - Number of high performance tester channels are minimized
  - Low cost testers can be used
  - Easier debugging
  - Can achieve high fault coverage

Previous Work

- Earlier studies shown poor correlation due to the lack of coverage of paths around memories (Belete et al, ITC 2001)
- Cory et al, IEEE Design & Test, 9-10/2003, found a linear relationship between the frequencies of the functional and latch-to-latch path delay tests.
- We could not duplicate D&T 2003 result for high performance designs (>1 GHz).

Types of Structural Tests

- At-speed memory BIST test
- Transition tests:
  - Simple transition tests: transition tests w/o going through memories.
  - Complex transition tests: transition tests going through memories.
- Path delay tests:
  - Simple path delay tests: latch to latch path delay tests.
  - Complex path delay tests: path delay tests involving memories or Cycle-stealing path

Chip Used for Experimentation

- MPC7455 microprocessor executing to the PowerPC™ instruction set architecture

Frequency	# Logic Transistors	# of Latches	# of Stuck-at faults
1Ghz+	6.8M	123k	6.2M

Structural Tests Used

- Simple transition tests: 13K with 70% fault coverage
- Complex transition tests: 12K with 78% fault coverage
- Path delay tests: top 2490 critical timing paths
  - Latch-to-latch paths: 1463
  - Memory paths: 91
  - Cycle-stealing paths: 231
  - Misc. paths, like clock or pre-charge paths: 700

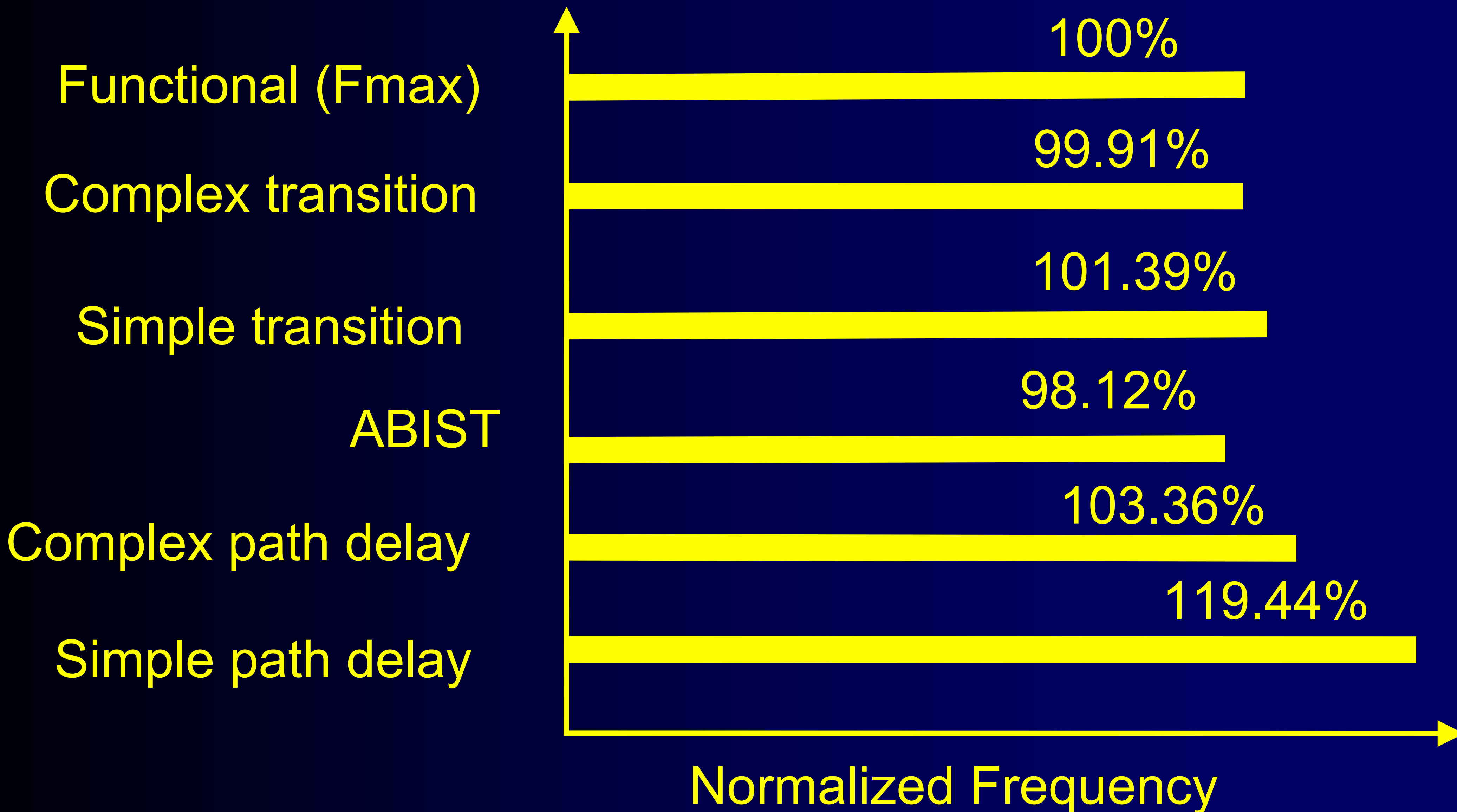
Path Delay Test Coverage

Path type	# of paths	Path coverage	# of Path tests	Test efficiency
Latch to latch	1463	60%	878	96.7%
Memory	91	95%	86	100%
Cycle stealing	231	63%	146	100%

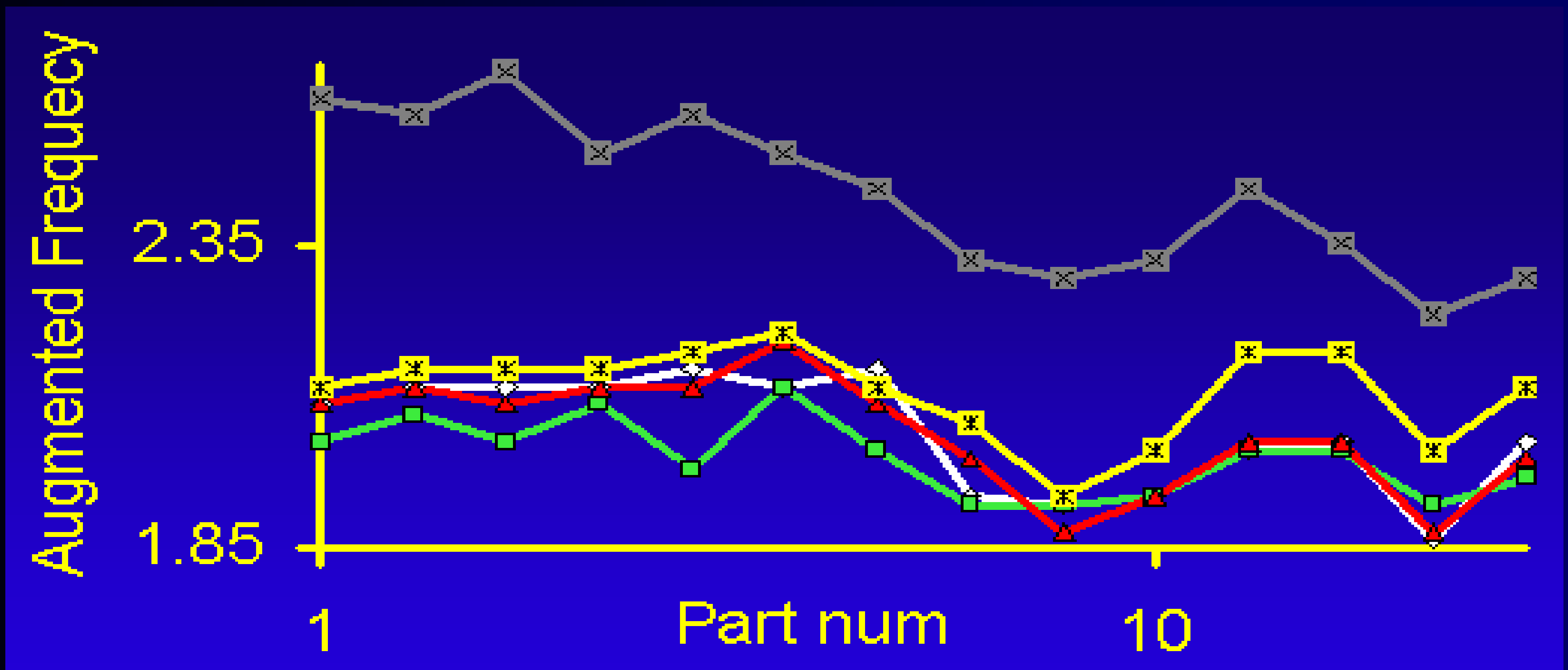
Experiment #1

- Purpose: trailblaze the methodology
- 14 packaged parts were used

- Measured maximum frequency of the functional and various structural tests
  - Structural frequency data normalized using the corresponding functional frequency
  - For each type of test, the average frequency of all parts was computed


Experiment #1 Results
100%
Functional (Fmax)
99.91%
Complex transition
101.39%
Simple transition
98.12%
ABIST
103.36%
Complex path delay
119.44%
Simple path delay
Normalized Frequency

Experiment #1 Results



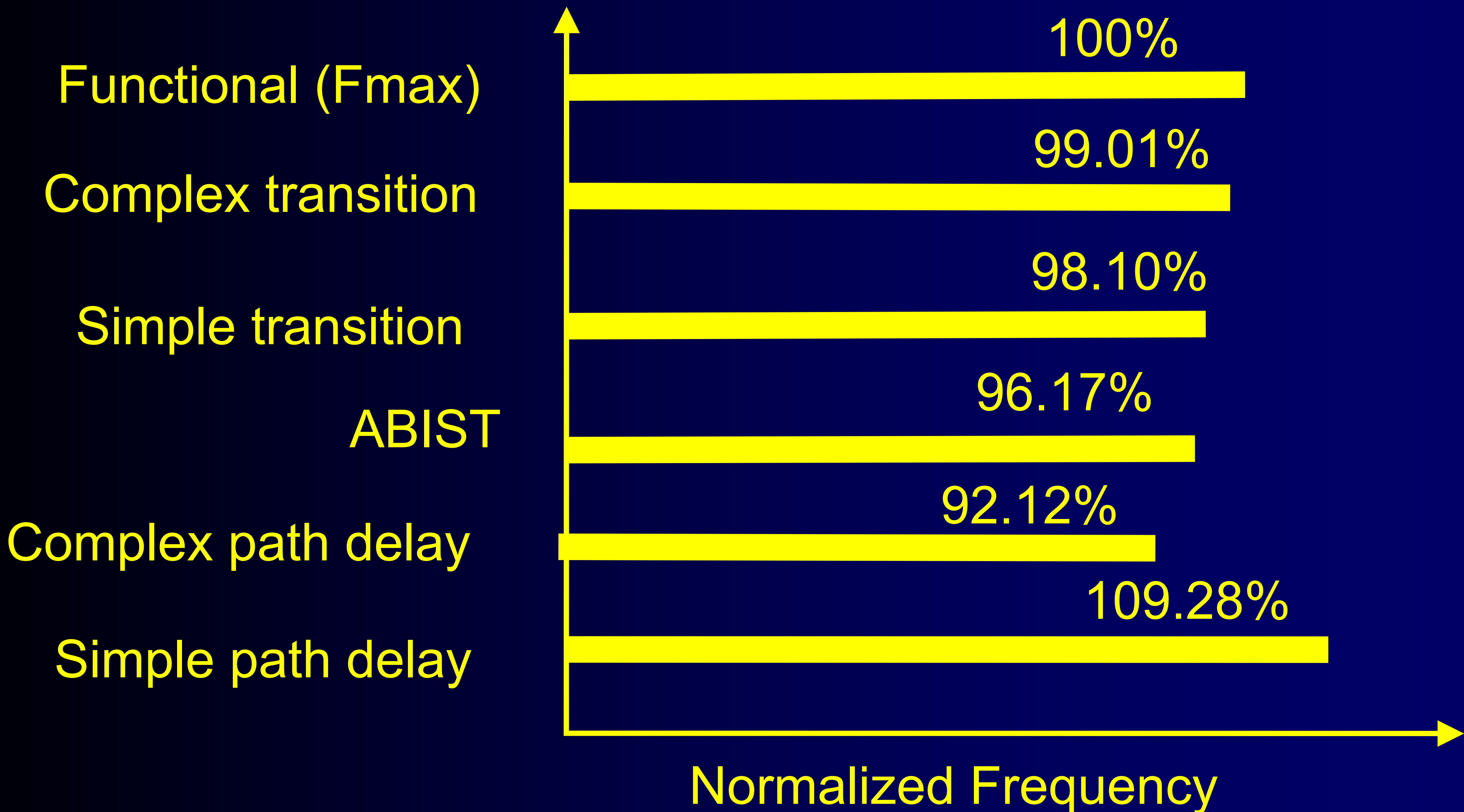
- Functional (Fmax)
- Simple path delay
- Complex path delay
- ABIST
- Complex Transition

Analysis of Experiment #1 Results

- Path Delay Correlation
  - Simple path delay (878 tests) ~20% faster than functional
  - Complex path delay (232 tests) ~3.5% faster than functional
  - No Linear relationship found between path delay frequency and Fmax
- Transition Delay Correlation
  - Complex transition tests correlated well with Fmax
  - Simple transition tests slightly faster on average
- ABIST Delay correlation
  - ABIST frequencies tracked closely but were primarily pessimistic (BIST activates test-only path)

Experiment #2

- Wafer probe experiment:
  - Frequency data of 411 die were collected from various sites on 7 wafers from a recent manufacturing lot.
  - Wafer probe test was performed on a Teradyne tester.
  - The average of normalized structural frequencies are computed


Experiment #2 Results
Functional (Fmax)
100%
Complex transition
99.01%
Simple transition
98.10%
ABIST
96.17%
Complex path delay
92.12%
Simple path delay
109.28%
Normalized Frequency

Trend Analysis

- Complex transition test provided the closest match to Fmax (on average) both at probe and at final.
- Simple path test was faster than Fmax
  - 19.44% faster during packaged test
  - 9.28% faster during probe test
- Complex path test (compared to Fmax) was
  - 3% faster during packaged test
  - 8% slower during probe test
- ABIST test frequencies were relatively lower (by 2%) at probe than at packaged test

Result Analysis

- Possible explanation for the performance difference between the probe and package tests:
  - Wafer data collected from newer and faster parts relative to the ones used in the initial package test experiment
  - Electrical environment differences
  - Difference in cooling between wafer-probe and package tests.

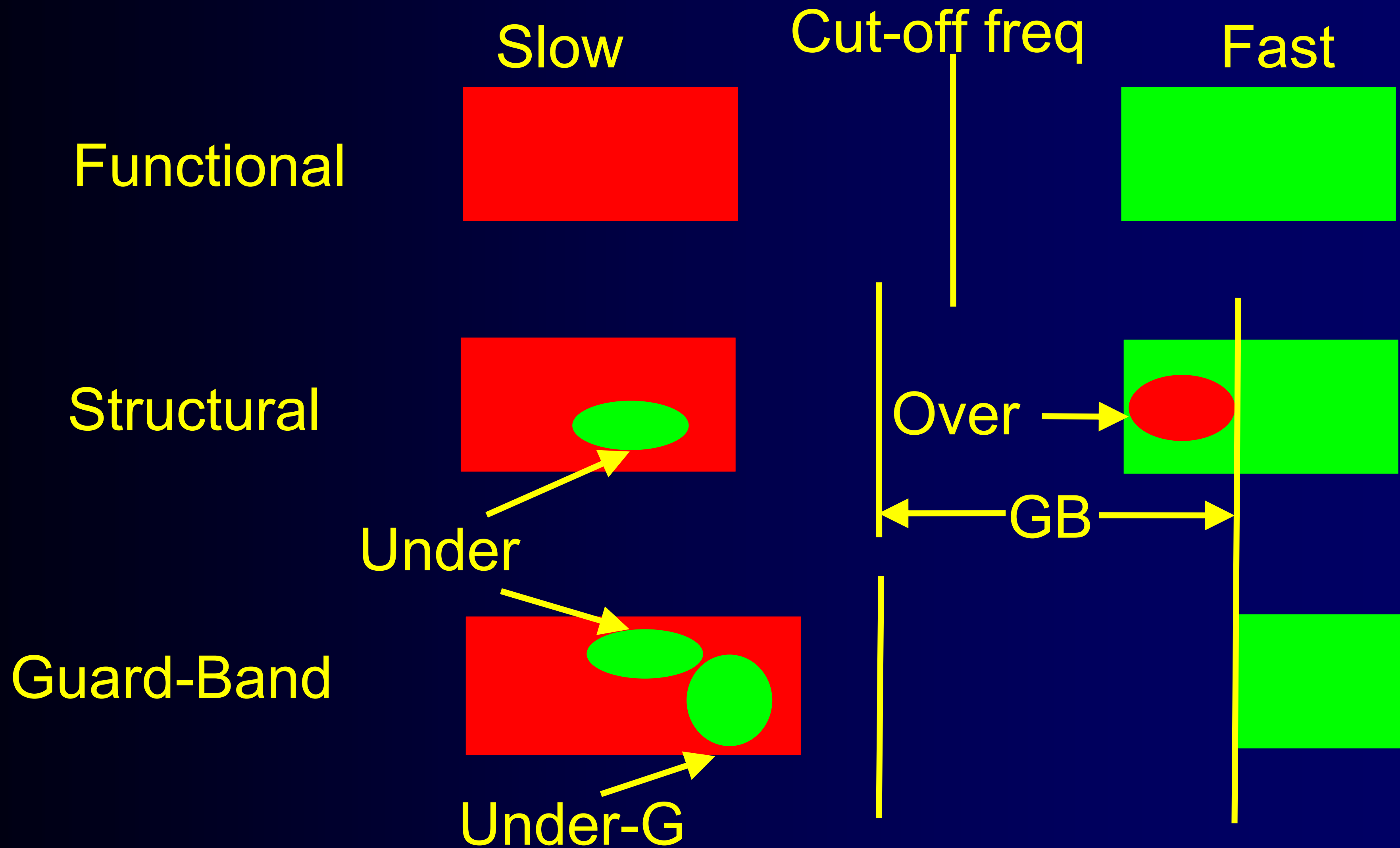
Potential Test Escapes

- We analyzed the limiting-speed paths of several die where the frequencies of structural tests were noticeably slower than that of Fmax
- In 88% of the complex transition test cases, the speed limiting paths were associated with complex memory transaction scenarios.
- That coincided with chips that passed functional tests but were failing in system tests associated with the same memory transactions. Investigation is ongoing.
- Analysis of fail data of other structural tests led to the identification of test-only paths.

Experiment #3: Speed Binning

- Speed binning data were collected for the 411 dies using functional tests:
  - Dies are divided into **slow** and **fast** speed bins.
  - The cut-off frequency between the bins defined arbitrarily as the average of the measured Fmax:
    - 179 in the slow bin, 232 in the fast bin.
    - Functional speed binning results is used as the reference point

Binning Metrics



Speed Binning Results

Corresponding average frequency was used for each type of structural test as the cut-off frequency.

Test type	Under	Over	GB
Complex Transition	4.4%	6.6%	2.2%
Simple Transition	3.2%	6.1%	2.2%
ABIST	3.9%	5.4%	2.2%
Complex Path	**1.9%**	4.8%	2.2%
Simple Path	5.8%	7.3%	6.4%

Guard Band Effects

Cut-off Frequencies = Average functional & structural

Under-G: additional parts which go into slow bin due to guard bands

Test type	Under	Over	GB	Under-G
Func	0%	0%	3%	18.3%
Func	0%	0%	5%	32.6%

Test type	Under	Over	GB	Under-G
Complex Transition	4.4%	6.6%	2.2%	16.7%
Simple Transition	3.2%	6.1%	2.2%	20.4%
ABIST	3.9%	5.4%	2.2%	22.6%
Complex Path	1.9%	4.8%	2.2%	17.0%
Simple Path	5.8%	7.3%	6.4%	36.9%

Summary

- Correlation between functional frequency and structural tests frequencies are encouraging
- Complex transition tests give the best correlation to the functional frequencies
- Almost all the structural tests performed reasonably well in speed binning the parts
- The results clearly demonstrate the importance of including structural delay path going through the memory arrays
- The data also suggests that some test escapes can be screened by structural tests

Break 5 minutes for questions

Next, we will continue on
other studies related to speed binning

Timing Correlation of Pre-silicon & Post-silicon

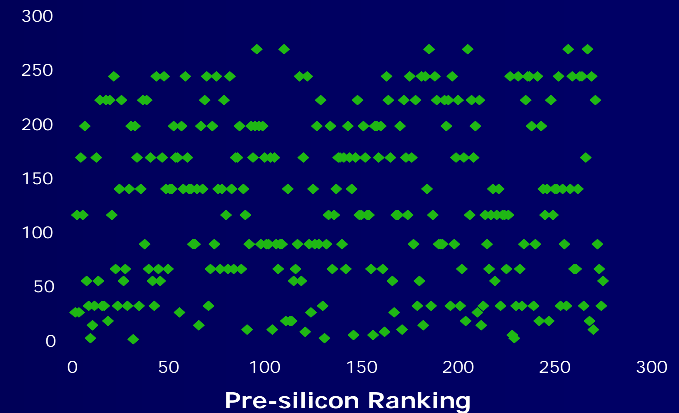
Two Studies

1. Correlating pre-silicon critical paths to post-silicon speed paths
   - How many pre-silicon paths to be tested in order to cover the top 10 speed paths?

2. Correlating structure testing frequency Tmax to functional testing frequency Fmax
   - Which structurally-tested paths can be used for speed binning (deciding fast vs. slow)?

1. Pre-silicon path ranking vs. post-silicon path ranking

- Pre-silicon (STA) most critical paths are not critical paths on the silicon
- Ranking correlation is poor:
  - Example: ranking correlation is .05
- Interesting questions
  - How many of the most critical pre-silicon paths are needed to cover real post-silicon critical paths?
  - Is there a metric that can be used to predict this?

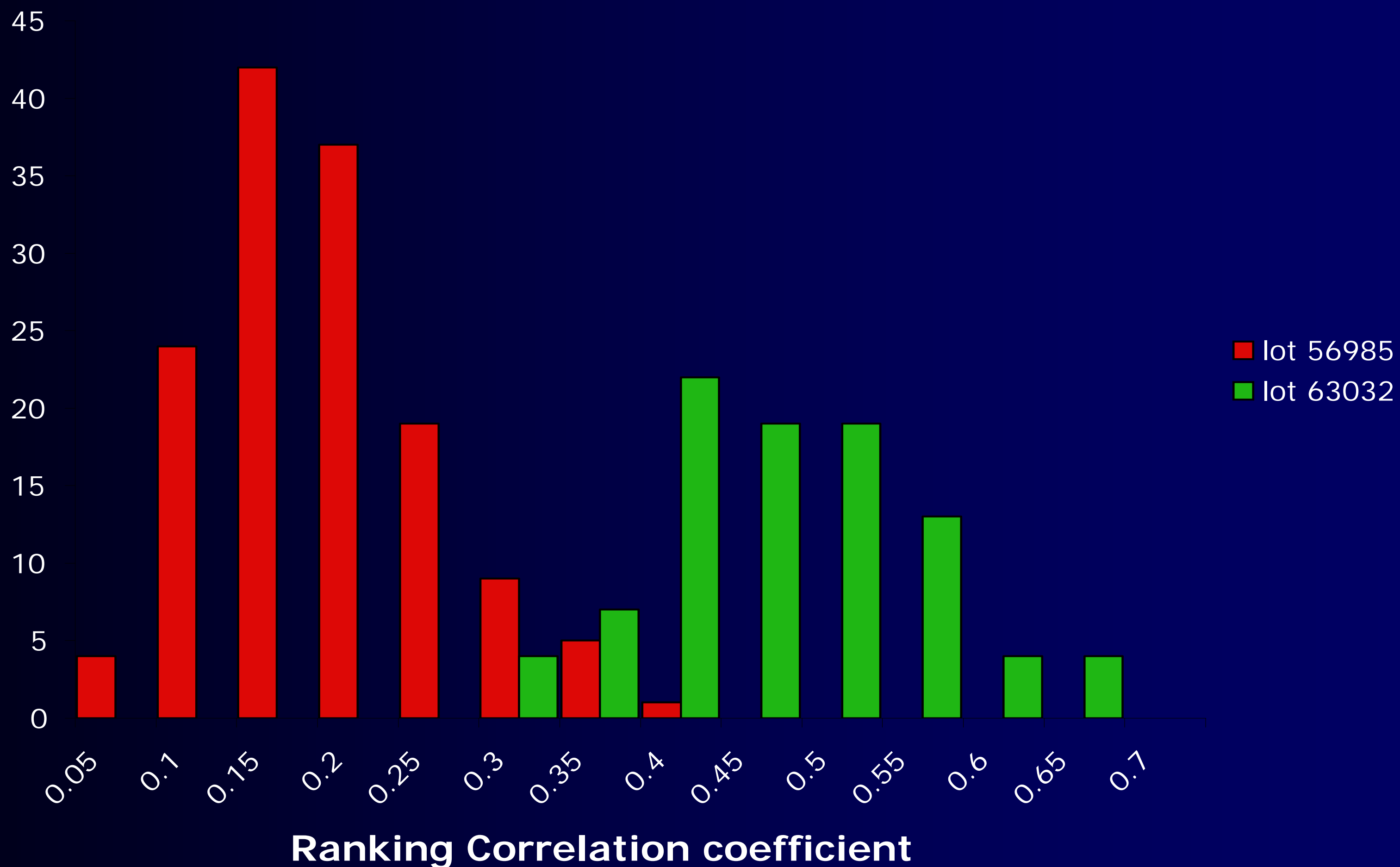


Experimental methodology

- Estimate pre-silicon/post-silicon *ranking correlation coefficient* from sample chips
  - Weighted Spearman Rho - actual most critical paths weighted more
  - MPC 7455 data
    - 130nm process technology
    - ~250 chips
    - Two Predominant Lots: 56985, 63032
  - Separate analysis:
    - Simple paths: **latch to latch**
    - Complex paths : memory, cycle-stealing
- Produce confidence plots for correlation ranges
  - Confidence plot: probability that the *x* most critical paths identified by pre-silicon STA cover the top 10 measured critical paths.
- Given a desired probability of coverage, use confidence plot to predict number of pre-silicon paths needed.

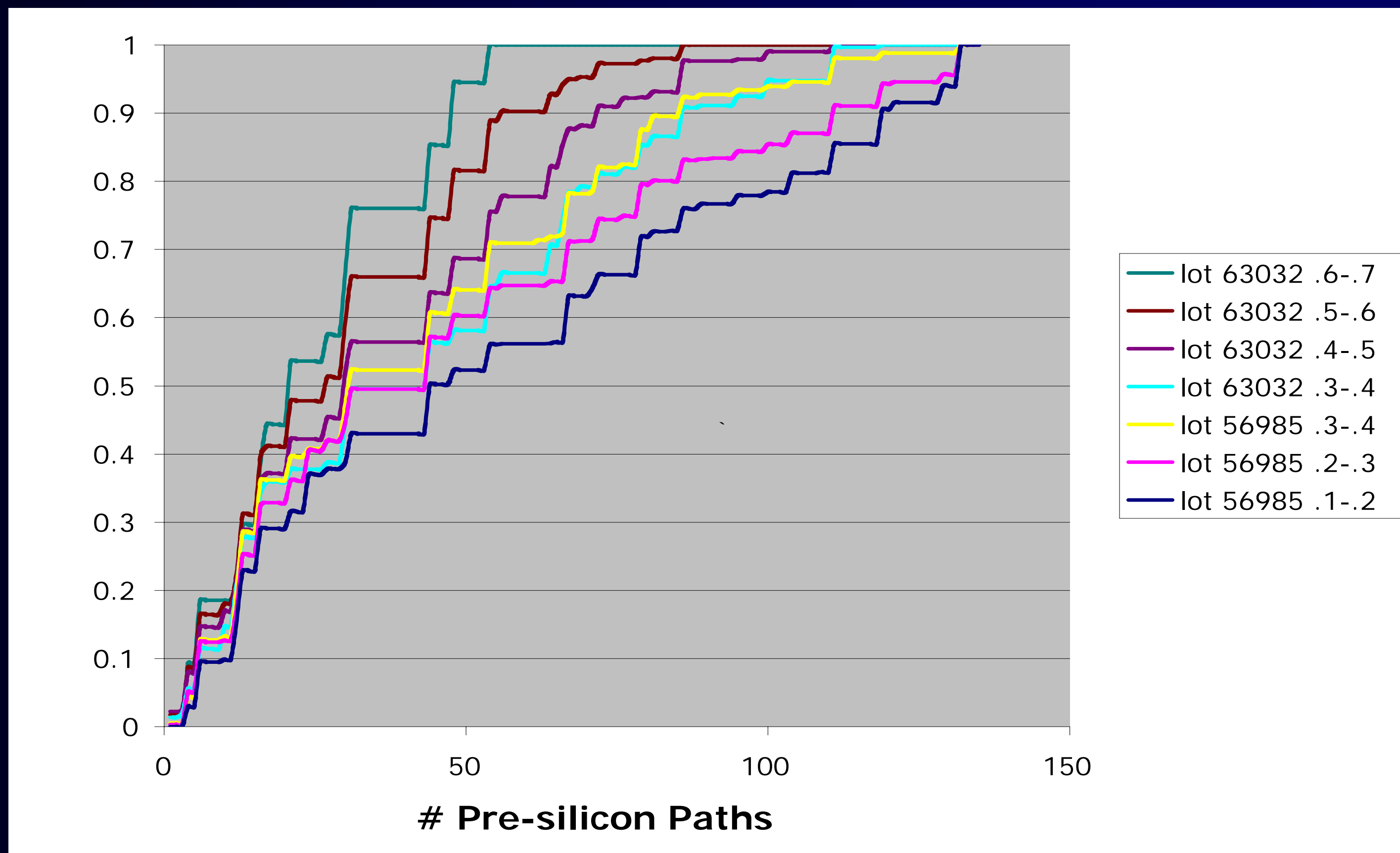
Latch-to-Latch Paths Correlation to Pre-Silicon

- Distributions almost disjoint



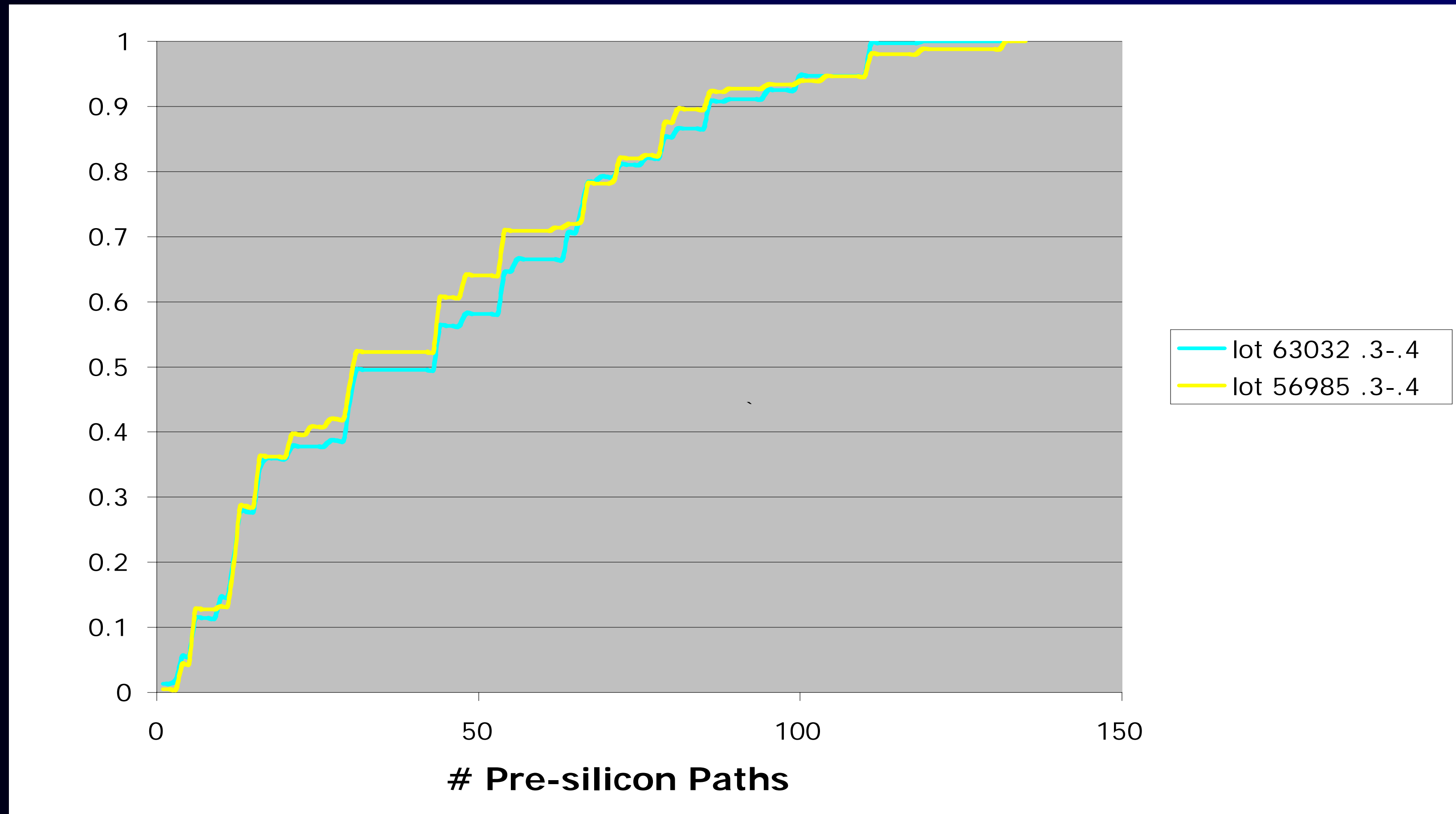
Confidence Plot

- Prob(Top *x* pre-silicon paths covers 10 most critical measured paths on the chip)
- Y-axis - Probability/Confidence
- X-axis - Top *x* pre-silicon paths



Lot-to-Lot comparison

- Early lot's confidence plot can accurately predict later lot's behavior



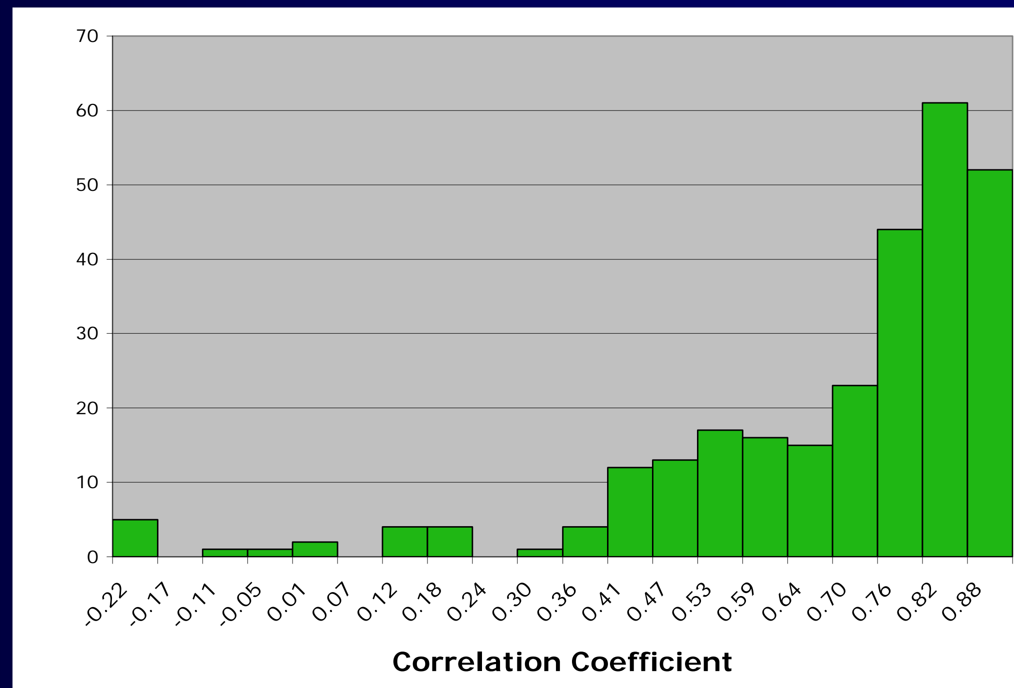
2. Issue of structural testing for speed binning

- For high performance designs, correlation between Tmax and Fmax is not high enough

Struct. Test	Fmax Cor
ABIST	.87
Smpl AC	.81
Cplx. AC	.76
Smpl Path	.83
Cplx Path	.82

Individual path correlation

- Obtain individual maximum frequencies for each path delay test
  - Instead of a maximum frequency for an entire set of tests
- Calculate correlation of each path to Fmax
- Highest correlation = .90 - higher than best structural test set



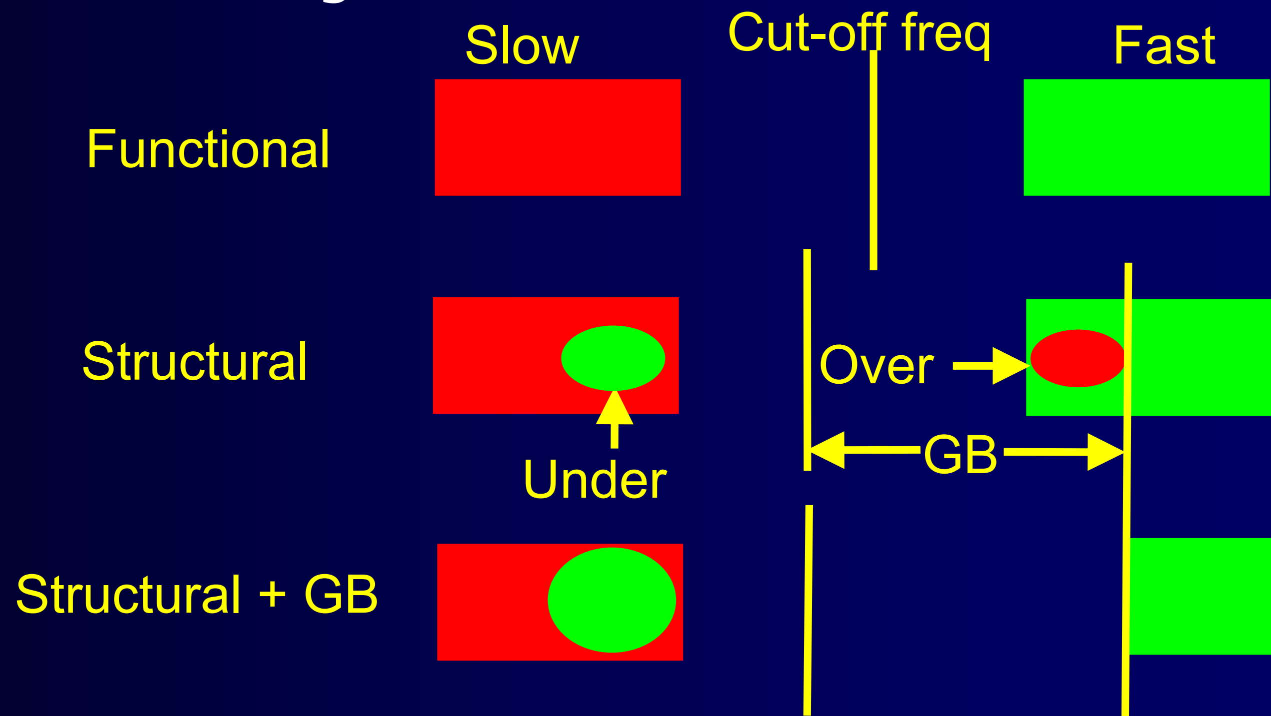
Properties of most correlated paths to Fmax

Path#	Type	Block	Ratio	Corr.
1174	Cplx	A	**1.61**	.90
1092	Cplx	A	1.11	.89
2161	Cplx	A	1.11	.89
3105	latch	V	1.57	.87
1817	Cplx	E	1.39	.87

- Ratio = Avg. Speedup relative to Fmax
- Individual path correlation to Fmax is higher than applying whole path delay test set together.
- Most correlated path is 1.6x faster than Fmax
- Less correlated, but slower paths mask these higher correlated paths out

Binning Accuracy

- Set the bin cut-off arbitrarily at the mean of the Fmax distribution
- 2-fold cross-validation
  - Randomly split set into two
  - Construct model with one half, predict on other half - vice-a-versa - average



Binning Accuracy

Test	Acc.	Under	Over	GB
ABIST	86.9%	8.6%	4.5%	1.9%
Smpl AC	81.8%	13.2%	7%	2.3%
Cplx AC	77.4%	11.1%	11.5%	2.86%
Smpl Path	79.1%	13.9%	7%	2.86%
Cplx Path	82.2%	9.5%	8.3%	2.5%

Path #	Acc.	Under	Over	GB
1174	91%	4.5%	4.5%	4.34%
1092	89.7%	4.1%	6.2%	5.2%
2161	89.3%	4.9%	5.8%	4.9%
3105	86.9%	8.2%	4.9%	3%
1817	86.8%	3.7%	9.5%	2.6%

Summary

- Post-silicon path delay tests can provide a wealth of information
  - Path ranking correlation metrics
  - Structural Speed-Binning

Thank you

Reference:
http://mtv.ece.ucsb.edu/TTEP/

Acknowledgement

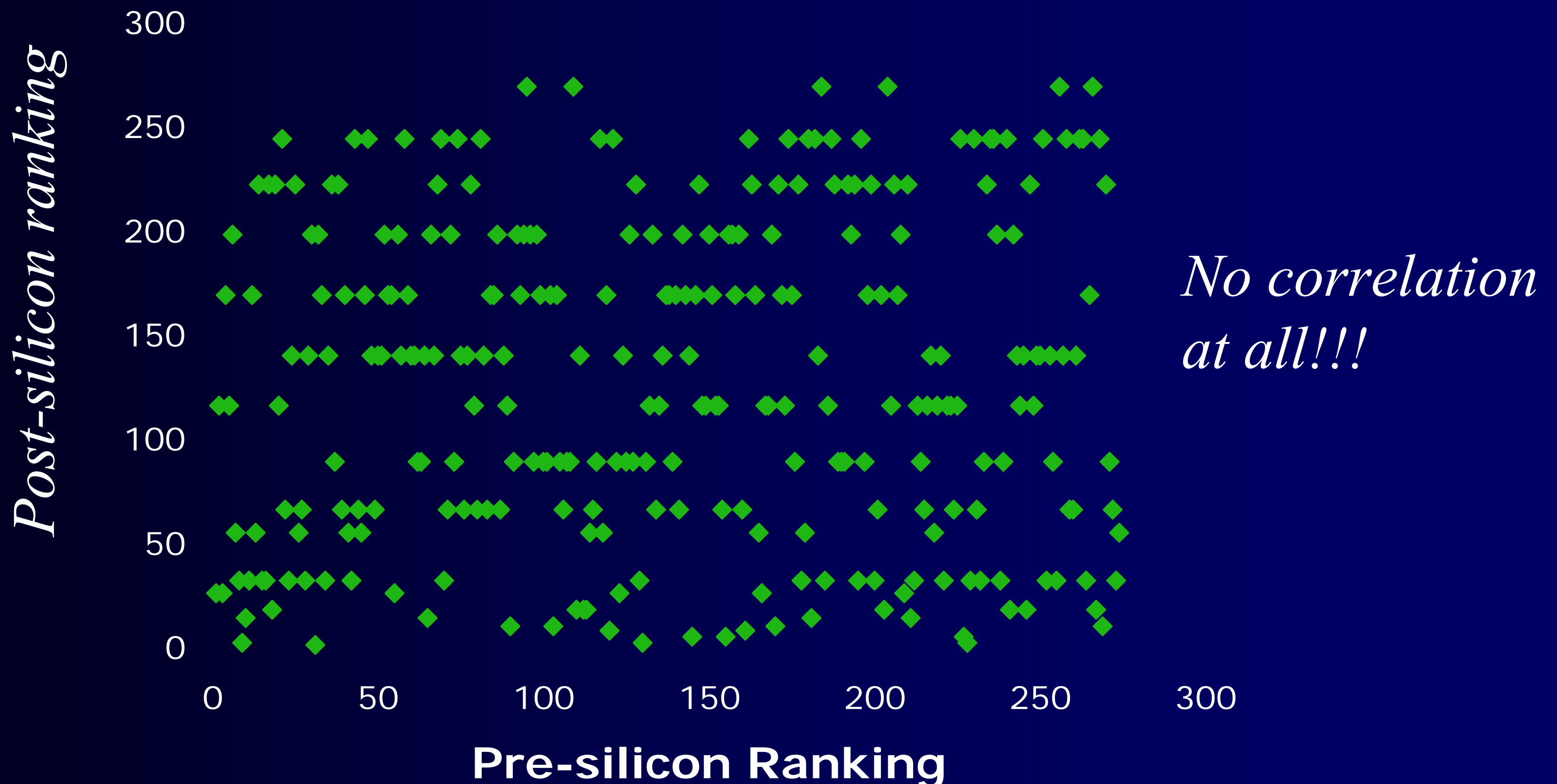
- Many people have directly and indirectly helped the making of this tutorial. Special thank to Noel Menezes at Intel SCL and Sani Nassif at IBM Austin Research for their invaluable insights regarding many current issues in timing analysis, DSM timing effects, variation modeling and process characterization. Benjamin Lee and Leonard Lee from UCSB helped to survey many papers in the areas of timing modeling, crosstalk, and power noise. Special thank to Nagib Hakim at Intel Santa Clara for his insight on binning with respect to inter-die and intra-die variations. Special thank to Jing Zeng at Freescale and Benjamin Lee for their work on the speed binning experiments. Thank to T M Mak at Intel Santa Clara and Praveen Parvarthala at Intel AZ for their valuable inputs on functional speed binning methodology. Thank to Wei-Ping Shi at Texas A&M University for his help on understanding RC extraction issues. And thank to many others who have directly or indirectly helped …

Advanced topics (optional)

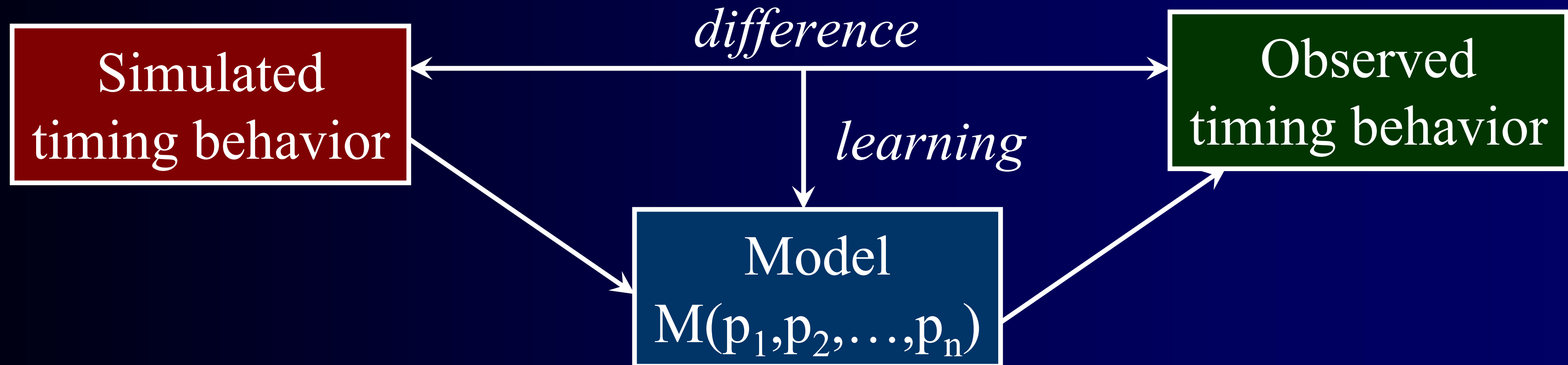
1. Model-based design-silicon correlation
2. Dealing with timing sensitivity

How to begin to explain this?

- Bayesian learning of spatial delay correlations
- See Lee et al. DAC 06



Model-based learning



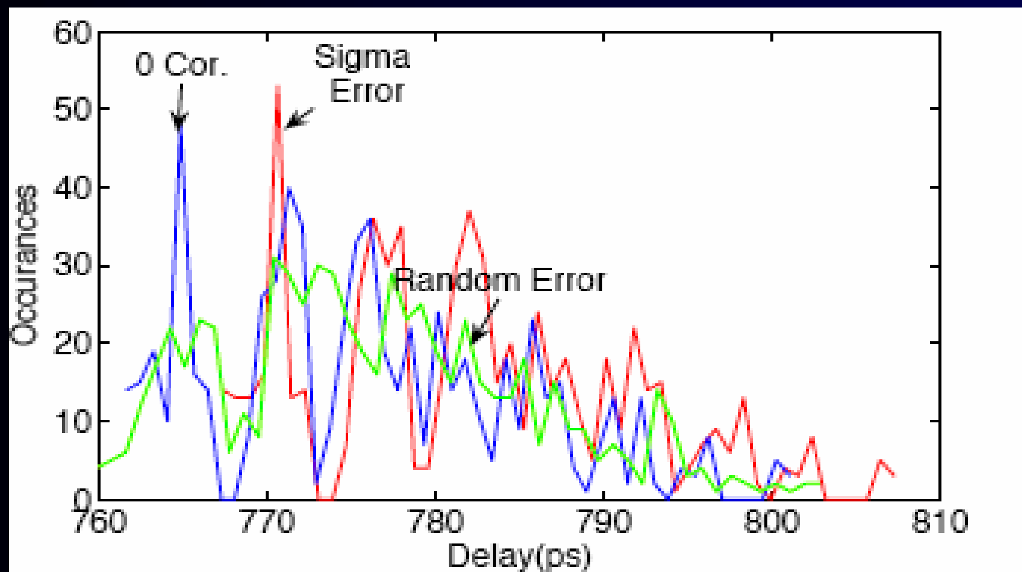
- The difference between simulated behavior and observed behavior is explained through a model
- From the difference, we estimate the model parameters $p_1,p_2,\ldots,p_n$
- The estimated model becomes a *recipe* in the future to fix the simulation results

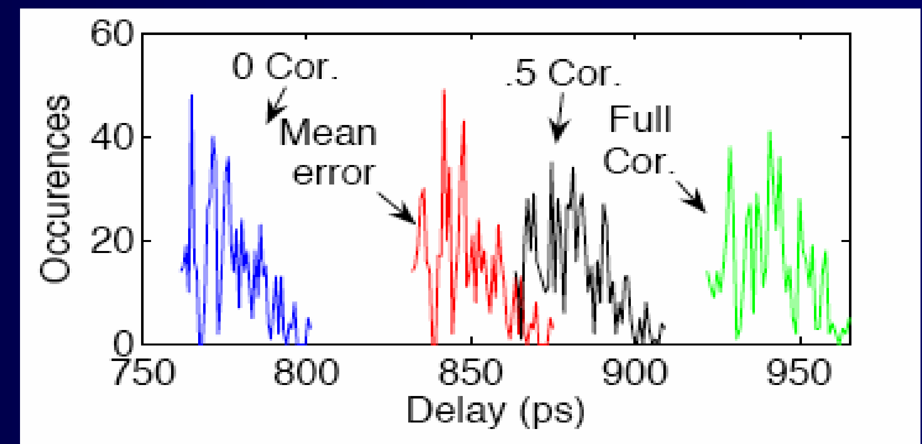
What model to begin with?

- Let’s ask the following questions:
- What information is hard to get in process characterization?
  - Spatial correlation (systematic)
  - Expensive to extract this data
- What impact the timing analysis result the most?
  - Spatial delay correlations
- If we are going to fix the timing model, what to fix first without affecting other effects (mean shift, sigma shift, etc.)
  - Spatial delay correlations

Comparison to other mismatch effects

- Plot shows distributions of 1000 most critical paths
- Other effects: Mean shift 10% , Sigma shift 10%, random shift

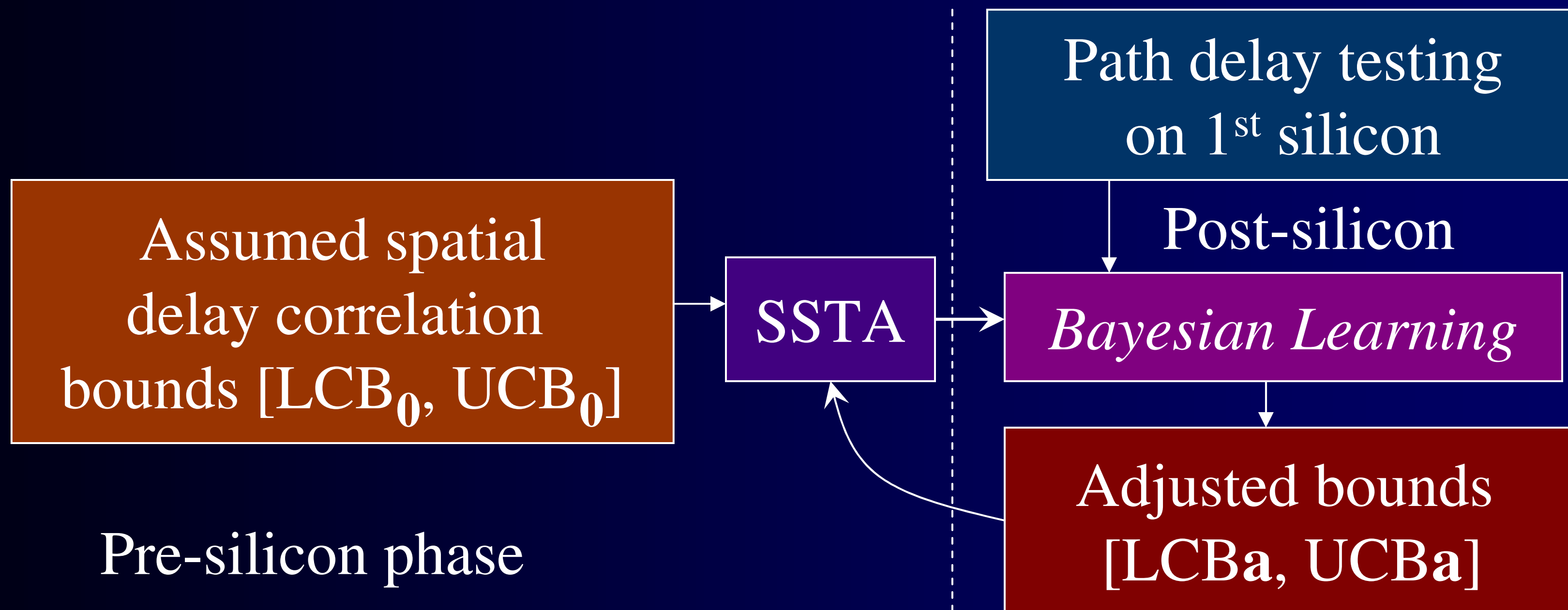




- Correlation is the most important

Bayesian learning of delay test result

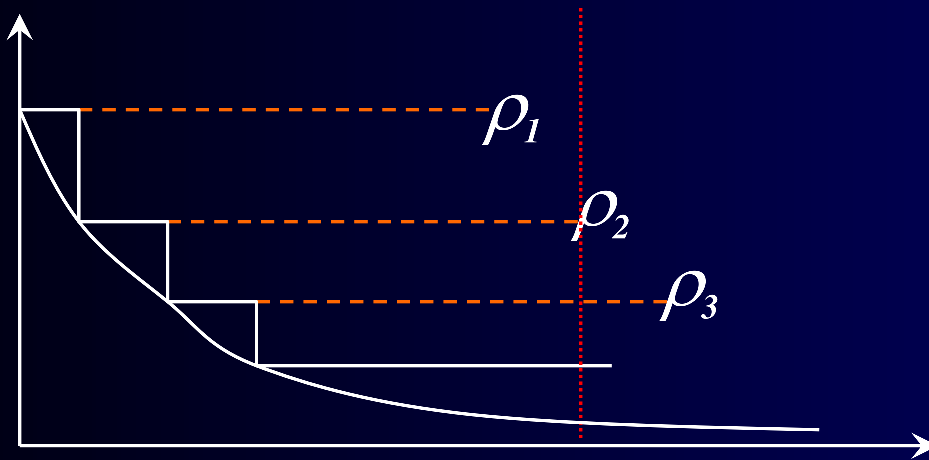
- Learn spatial delay correlations via path delay testing

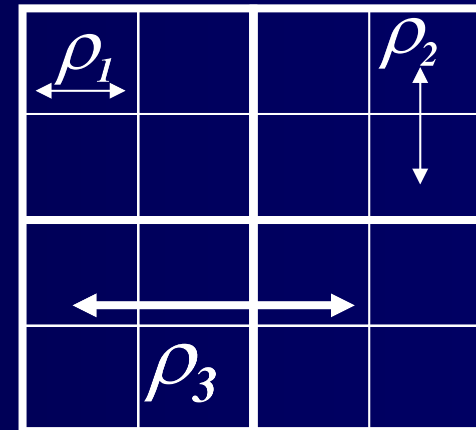


- Apply results back into SSTA to improve accuracy of circuit timing distribution

Using a discretized grid model

- Illustration of discretization



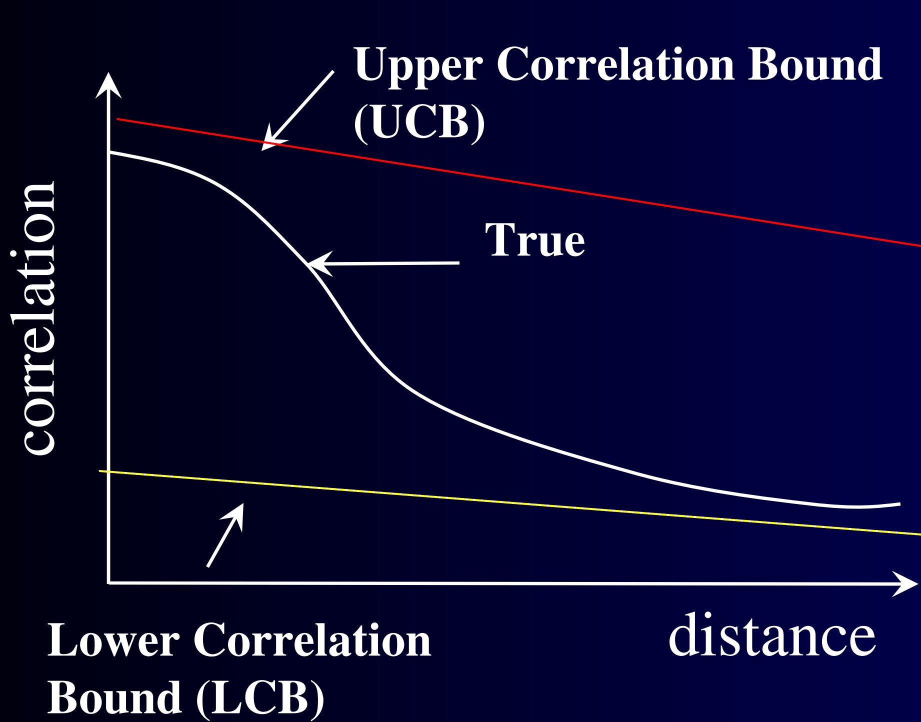


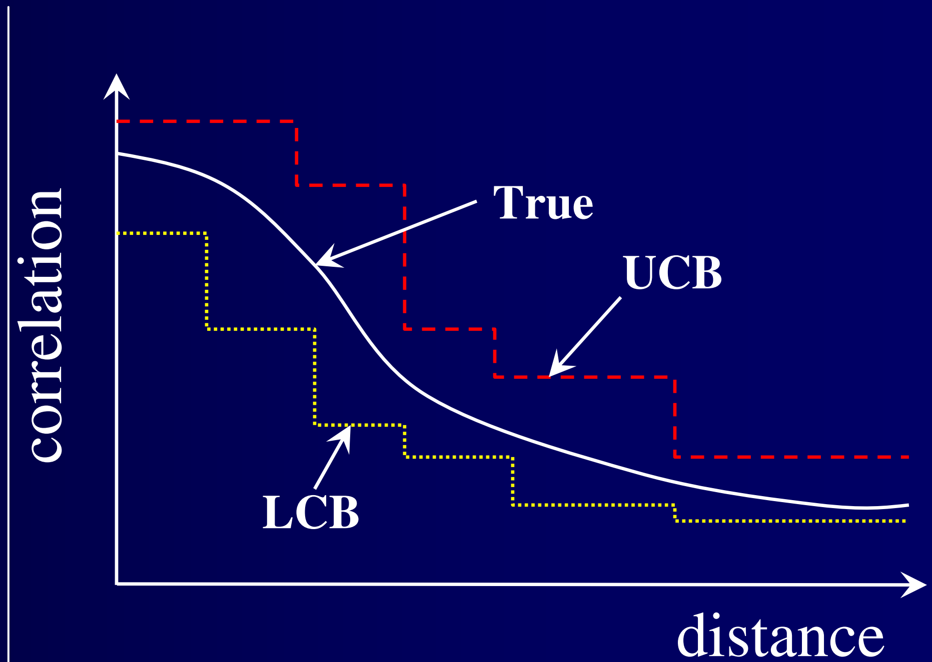
3-par. local view

- Grid model is a good fit for PB-STA
- Creates a learning problem of learning ρ_1, ρ_2, ρ_3 etc.

Correlation bounds

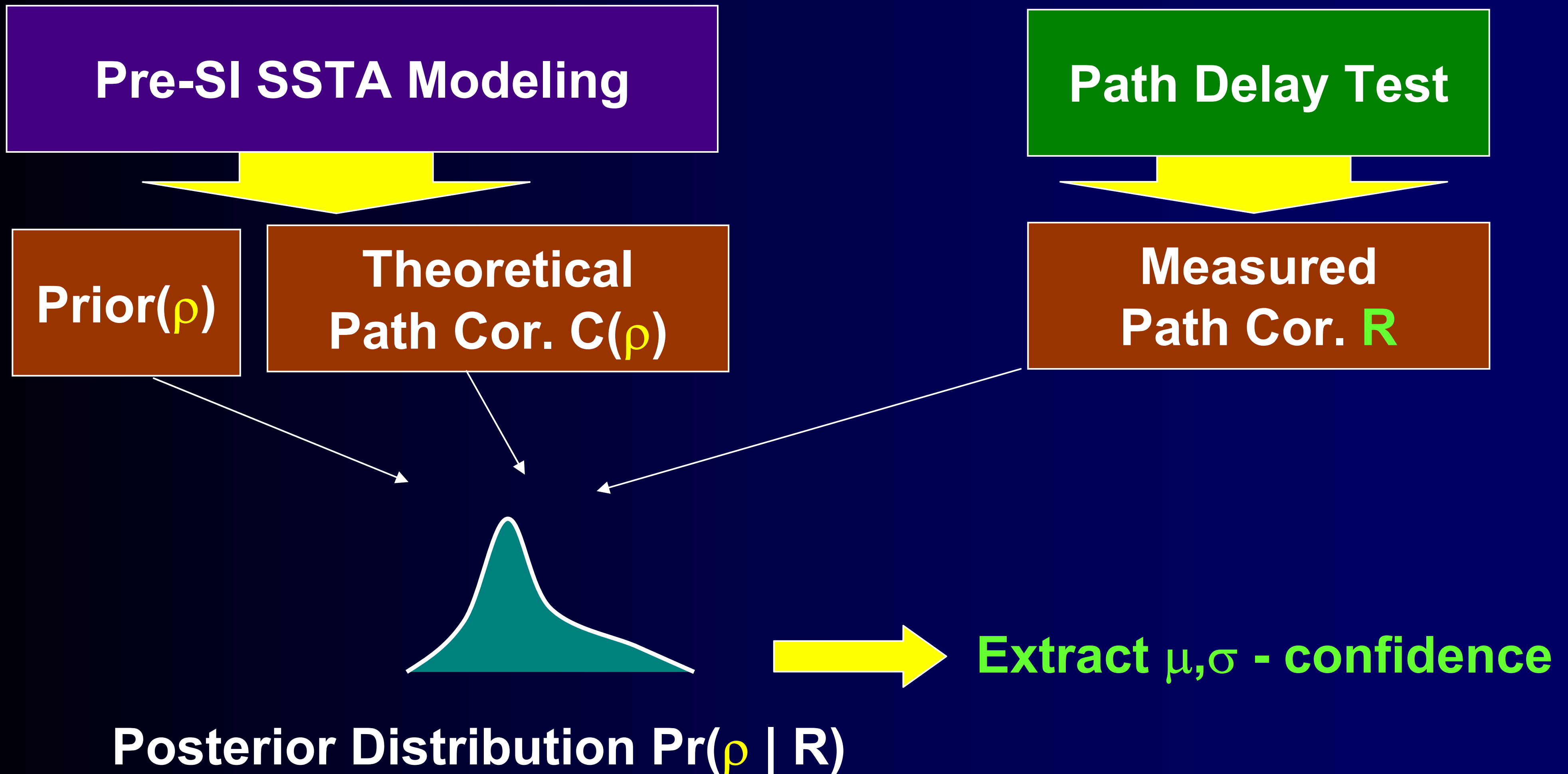
Prior —Bayesian→ Posterior





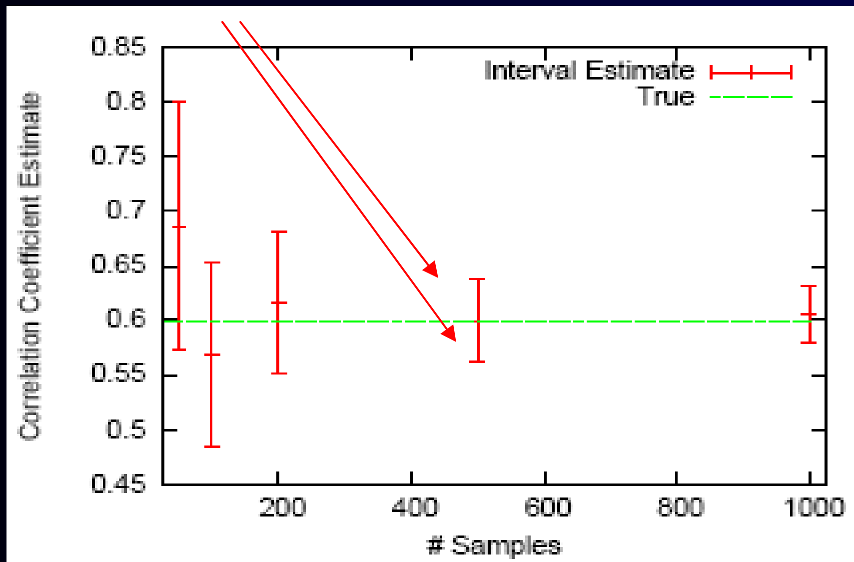
- Before learning
- After learning

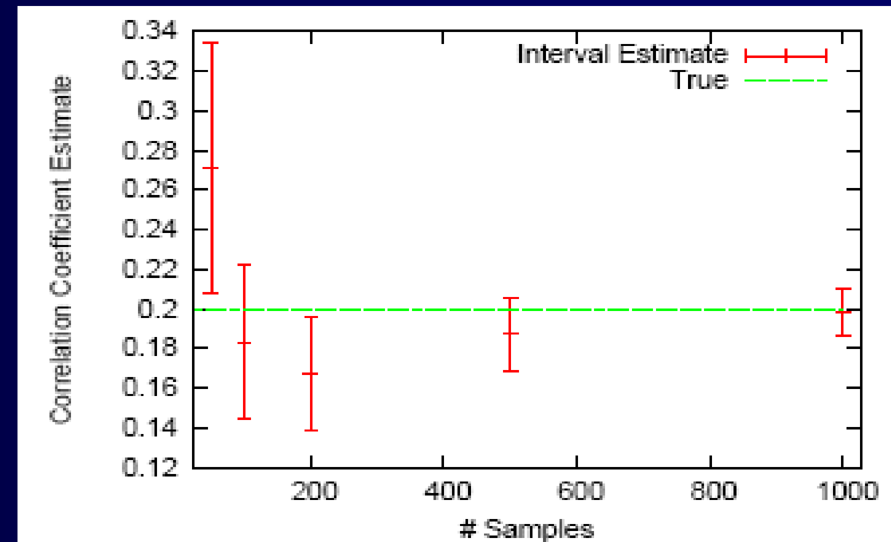
Overview of Bayesian learning



Experimental results

Confidence interval





Local correlation estimate

Global correlation estimate

- As the number of samples increase, estimate gets better and the interval shrinks

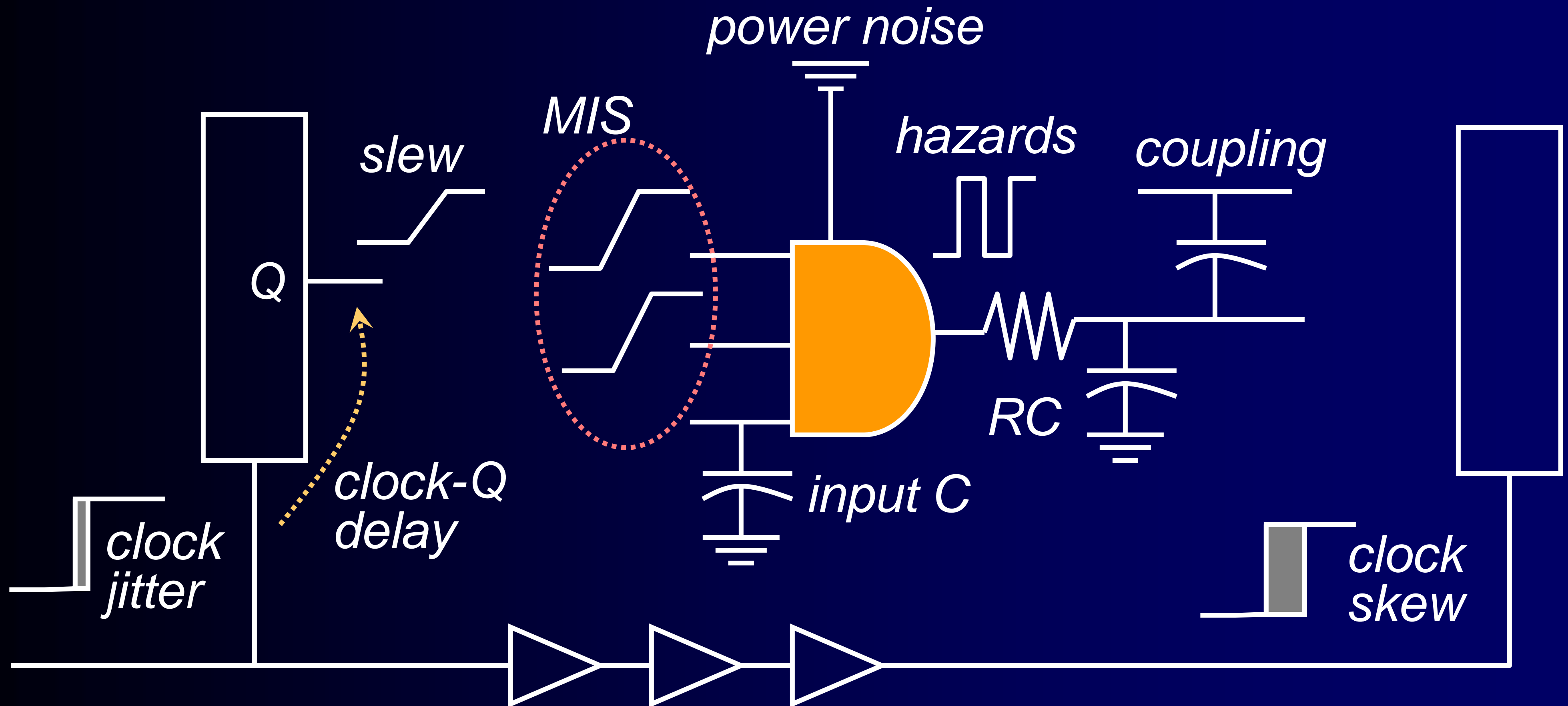
Summary of results

Timing bound (ps) from SSTA

Circuit	Posterior	Prior	Observed	Margin
C880	1310	1431.4	1293.8	9.38%
C1355	1340.5	1394.03	1338.03	4.04%
C2670	1941.4	2104.7	1934.44	8.44%
Ind32	917	982.6	907.8	7.22%

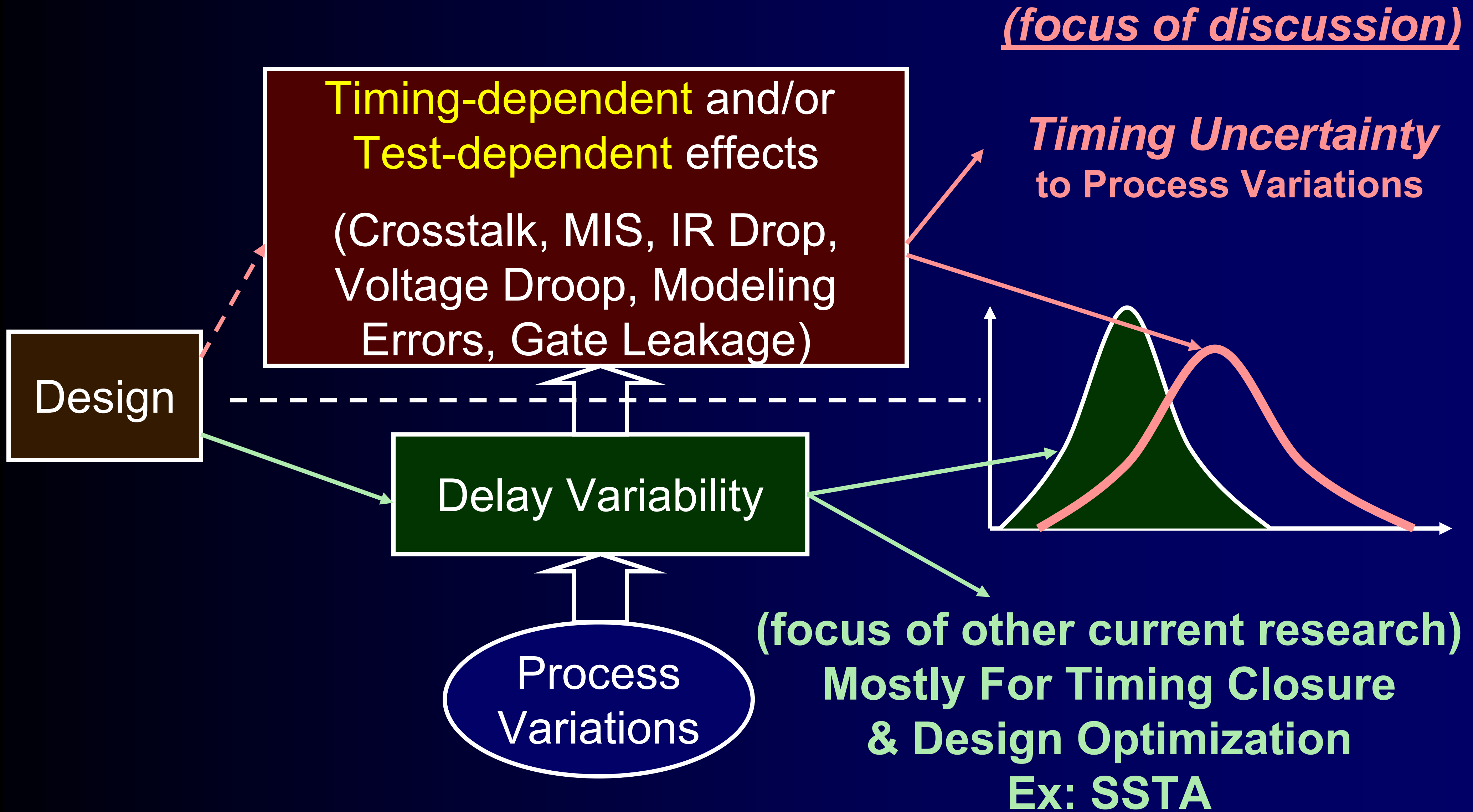
- After learning, posterior is closer to the observed
- For detail, please see DAC06 paper

Recall: Timing uncertainty



- Many effects are "sensitive to delay variation," complicating the simple view
- Many effects also depend on test patterns

Recall: Variability vs. Uncertainty



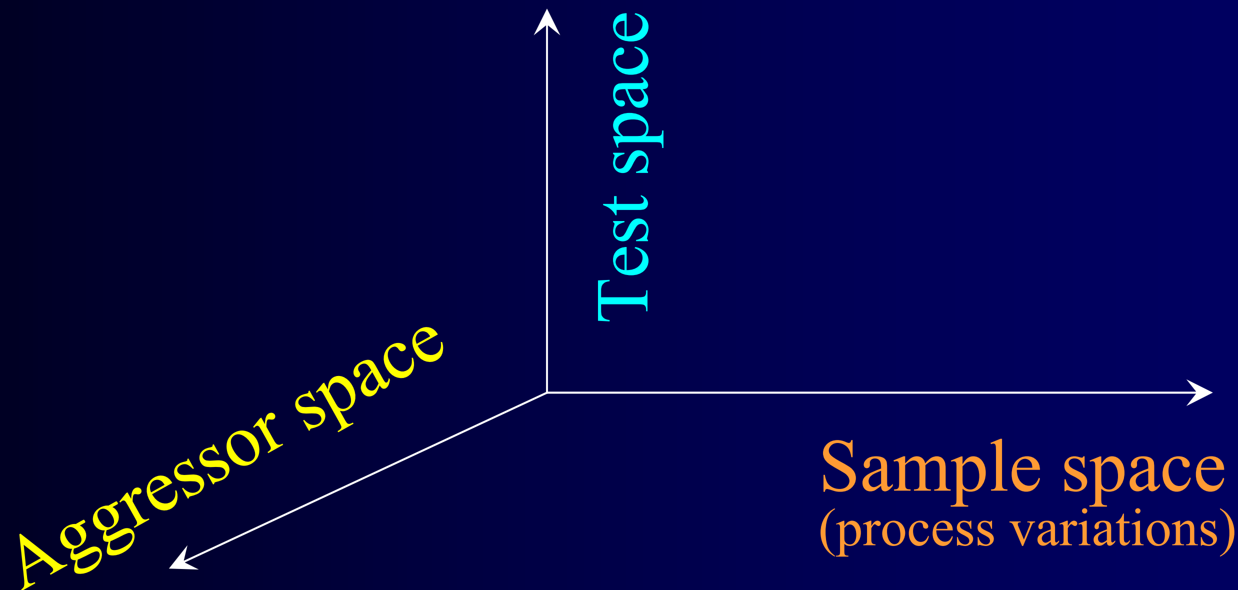
Pick a driver example

- To begin the study, we
  - Select cross-coupling as the driver example
  - Develop an approach to overcome the complexity (in the statistical space)

Issues

- For design, worst-case window analysis can be overly pessimistic
  - How can we shrink the window further?
    - ✓Without assuming an accurate timing model
    - ✓We can analyze the input test pattern space to prune the # of aggressors to be considered
- For test, how can we validate that cross-coupling effect doesn’t cause excessive delay on a path?

Three dimensions of the problem



- Dealing with the three dimensions simultaneously can be a very difficult problem
- Better to separate them so that different methodologies can be applied

Aggressor space

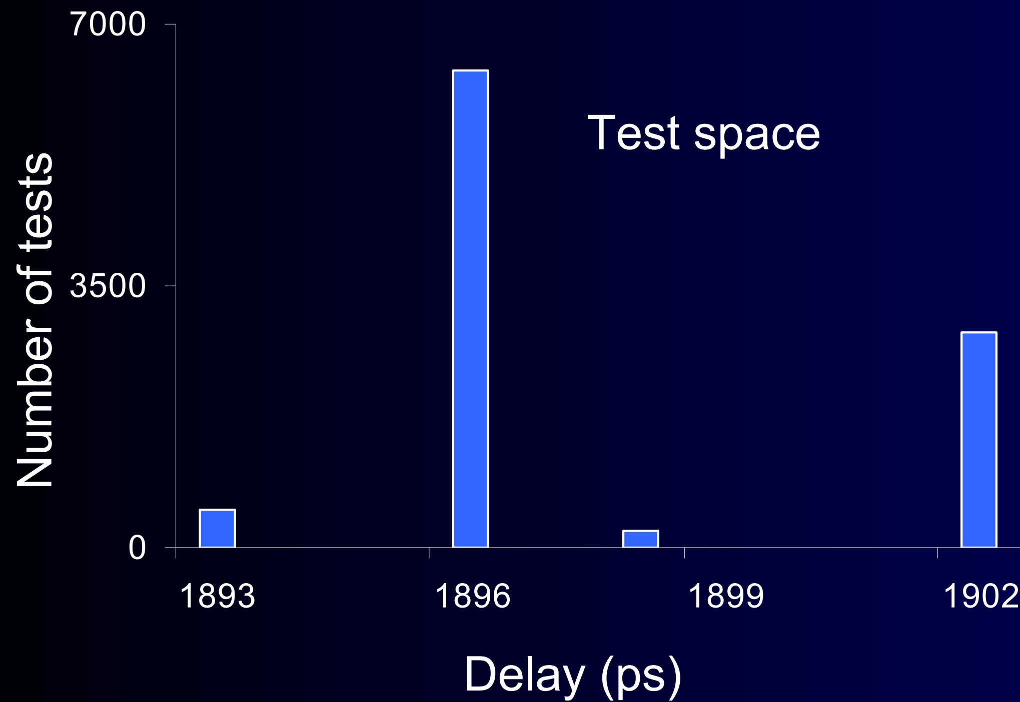
- Given n aggressors, the space consists of 2^n combinations
  - (Pre-processing step taken to fix the worst-case transition of each aggressor)
  - Each combination $(a_1,\ldots,a_n)$ where each $a_i \in \{0,1\}$, indicates the subset of aggressors that can be "activated" together by a test pattern
  - $a_i = 1$ means that there exists a test to produce a transition on the aggressor, opposite to the victim transition

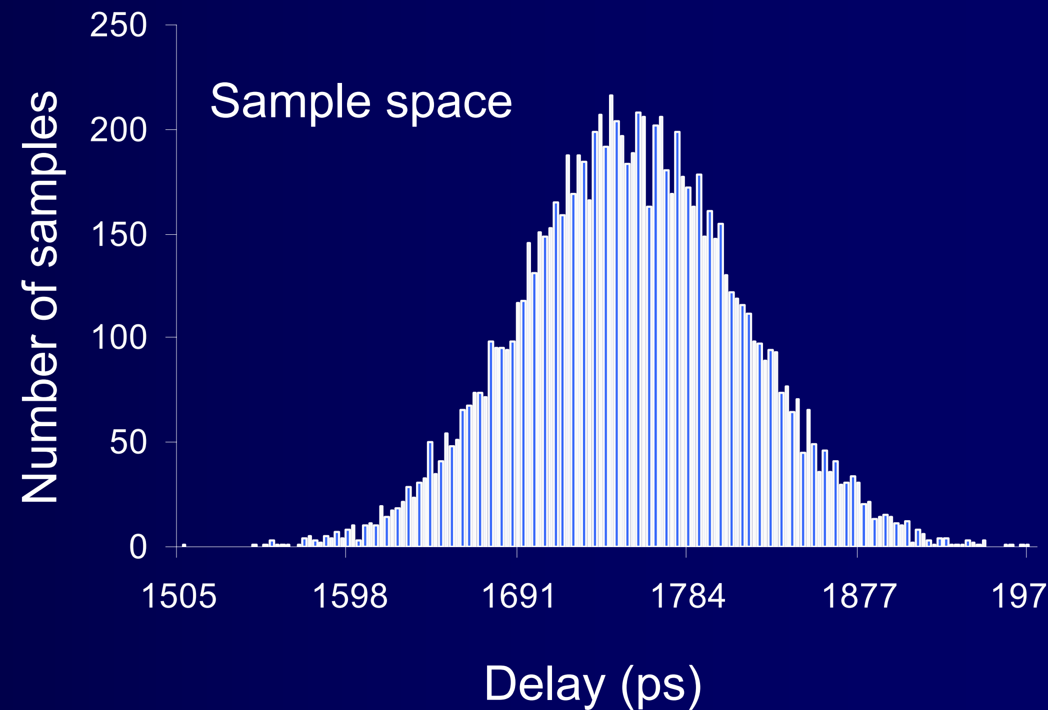
- We can prune this space by analyzing the test input space logically

Test space, sample space

- Given an aggressor combination, there can be many tests for it
  - On a given chip, different tests can produce different timing effects
- Given an aggressor combination and a specific test,
  - On different samples, they can have different alignments, causing different delays

Examples





- Test space (left)
  - Non-parametric (does not fit a continuous distribution well)
- Sample space (right)
  - Fits a Normal distribution well

Worst-case test pattern

- Traditionally, research was done to find the worst-case test pattern for a path under cross-coupling effects
  - ➢Given a timing model, finding the maximum set of alignments
- Issues:
  - ➢Timing model is not accurate
  - ➢There may *not* be just one worst-case test

One approach (ITC06, ICCAD06)

- Two separate methodologies
- In the pre-silicon analysis, we combine
  - (1) logical input test analysis
  - (2) worst-case window analysis
  - to prune the aggressor space
- Test space and sample space are dealt with in the post-silicon
  - By preparing a superset of tests
  - By *learning* from silicon samples
  - Then, by optimize the test set

Path	From extraction	After pruning
c880-1	72	7
c880-2	76	10
c880-3	78	16
c880-4	75	10
c880-5	79	12
c880-6	81	20
c1355-1	81	20
c1355-2	79	18
c1908-1	88	12
c1908-2	81	17
c7552-1	157	22
c7552-2	166	30

- Many aggressors can be pruned by considering logic and/or timing constraints

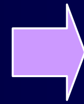
Test space + sample space

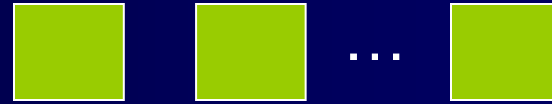
A statistical learning problem

- non-parametric density estimation

Test + sample space?

You don't want
to look inside
(just ignore it)





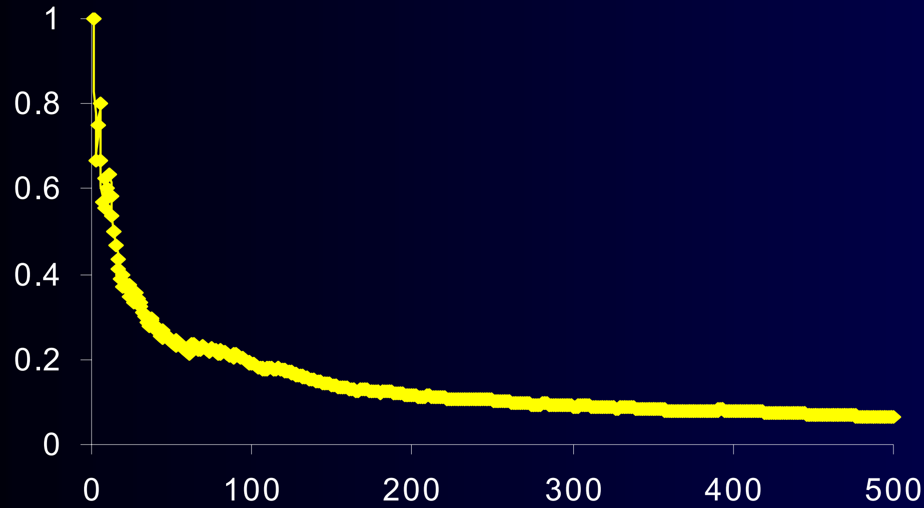
Just use M samples
to analyze what you
really care about
(a statistical learning problem)

- Generate i tests for each remaining aggressor combination to collect a superset of test patterns
- Apply these tests on M samples
- Select tests from the results
- Develop a method to bound the delay of each given sample

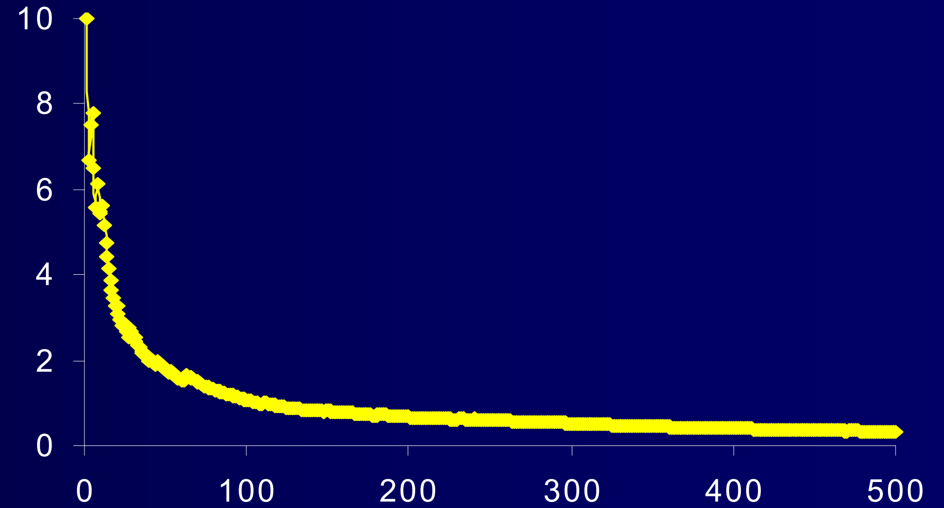
Select the worst j tests from every sample: Test-2-sample ratio (T2S)

- How many tests will I see for a large S
- Condition for a tractable problem
  - Weak condition: T2S $\rightarrow$ 0 as S $\rightarrow$ large
  - Condition: T2S $\rightarrow$ 0 quickly as S $\rightarrow$ large
- Otherwise, the solution space is unbounded
  - Different samples see different worst tests
  - This may be an indicator of design problem

T2S ratio – exponentially decay



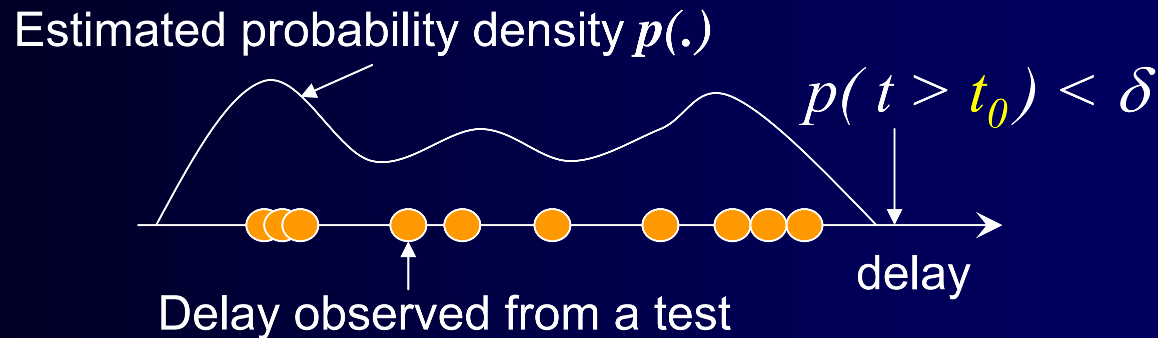
Top 1 test from every sample



Top 10 tests from every sample

- T2S decreasing quickly tells that our solution space is *asymptotically* bounded

“bounding” as density estimation

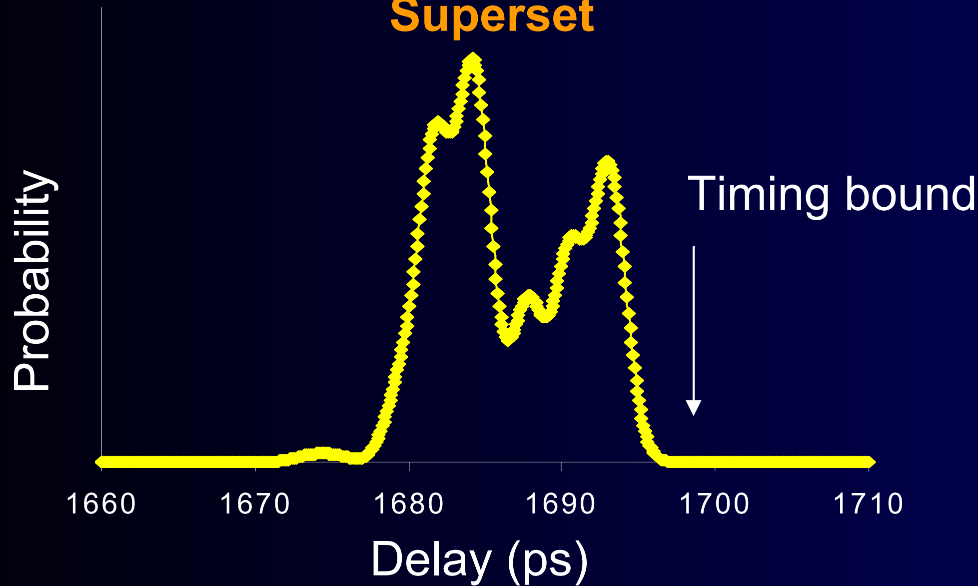


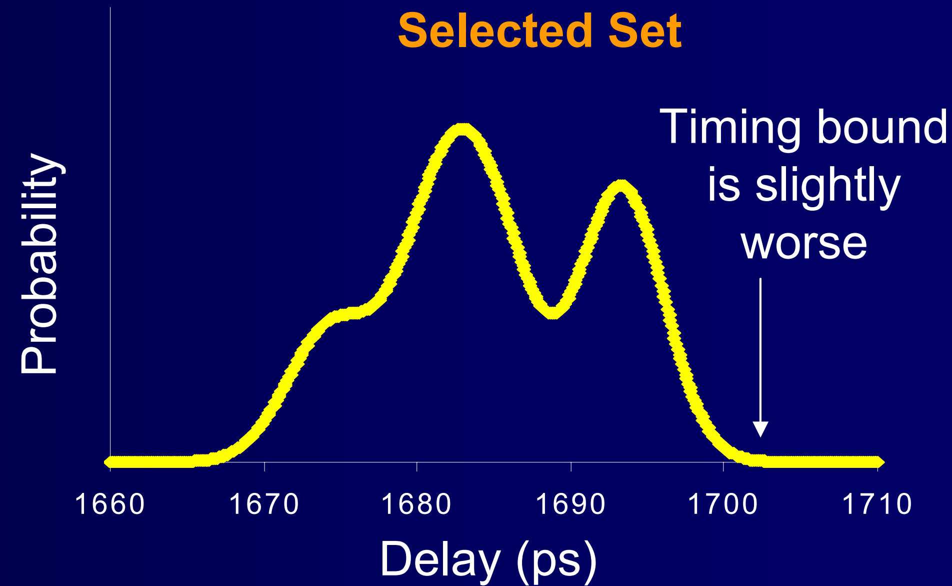
- The test space is too complicated to analyze
- Non-parametric density estimation for test space
  - Kernel density estimation (SVM)
  - We use the statistical tool R to find *density*
    - ✓ Superset will give you a bound
    - ✓ Selected test set will give you another bound
      - Ex. by selecting the top 10 tests from each sample
  - The selected test set will give you a bound that is worst than that given by the superset

Test pattern selection

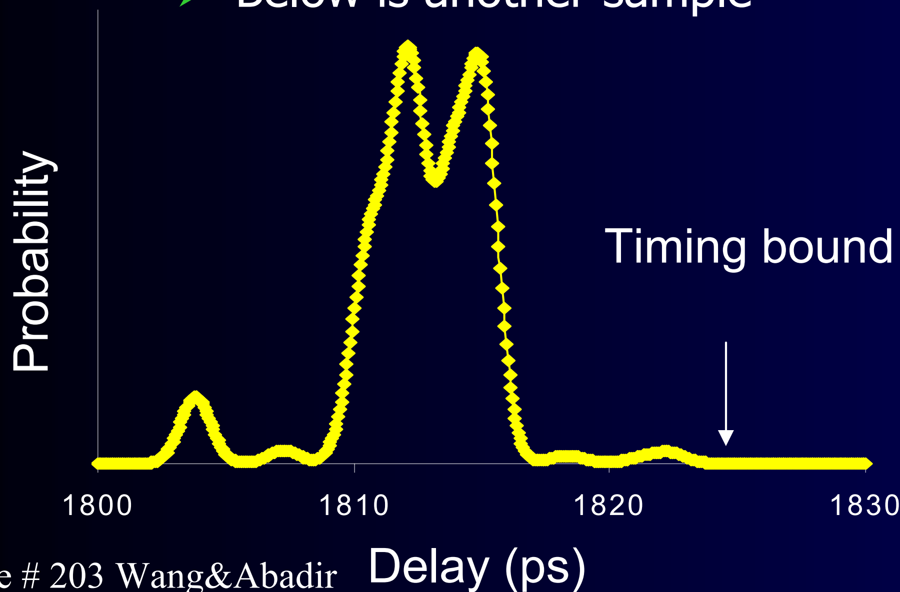
- With SVM estimator and a confidence level, Superset will give you a bound
- Select top *j* tests from every sample
- Selected test set will give you another bound

- We verify that the selected test set gives you a bound that is worst than that given by the superset (This is a nature of the estimator)





- Superset timing bound is tighter than selected set
  - This shows kernel density estimation of one sample on c880-1
  - Below is another sample



- Average superset and selected set bounds across 50 samples
  - Difference in bound is (selected set – superset) / superset
- Number of tests selected by taking the top 10 tests over 50 samples

Circuit	Superset Bound (ps)	Selected Set Bound (ps)	Difference in Bound	Selected Set Size*
c880-1	1762.915	1765.970	0.1733%	60
c880-2	1758.186	1761.543	0.1909%	96
c880-3	1769.120	1770.931	0.1024%	68
c880-4	1753.318	1754.906	0.0906%	99
c880-5	1746.698	1749.692	0.1714%	73
c880-6	1753.572	1755.538	0.1121%	76
c1355-1	1739.008	1739.204	0.0113%	80
c1355-2	1721.848	1722.647	0.0464%	72
c1908-1	1806.478	1807.923	0.0800%	51
c1908-2	1783.986	1784.032	0.0026%	190
c7552-1	2233.757	2235.919	0.0968%	161
c7552-2	2277.706	2282.426	0.2072%	121

*** Superset size is ≈ $10x2^l$ is about 5-10K**