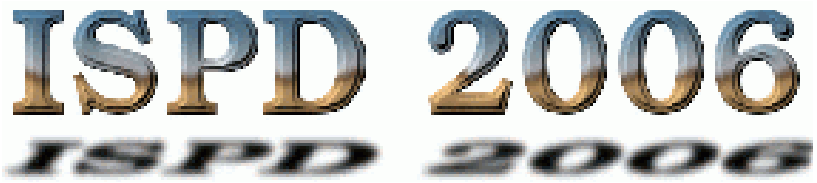


Fast Buffer Insertion Considering Process Variation

Jinjun Xiong, Lei He

EE Department

University of California, Los Angeles

Sponsors: NSF, UC MICRO, Actel, Mindspeed

Agenda

- Introduction and motivation
- Modeling
- Problem formulation
- Detailed algorithms with complexity analysis
- Experimental results
- Conclusion

Buffer Insertion Flashback

- **Buffer insertion and sizing is a commonly used technique for high-performance chip designs to minimize delay**
- **Classic results on buffer insertion**
  - Two-pin nets: closed form for optimal solution [Bakoglu 90]
  - Multi-pin nets: dynamic-programming based algorithm to find the optimal solution [Van Ginneken 90]
- **Extensions**
  - Multiple buffer libraries considering power minimization [Lillis 96]
  - Wire segmentation [Alpert DAC97]
  - Simultaneous buffer insertion and wire sizing [Chu, ISPD97]
  - Simultaneous tree construction and buffer insertion [Okamoto DAC96]
  - Simultaneous dual Vdd assignment and buffered tree construction [Tam DAC05]
  - …..

Design Optimization in Nanometer Manufacturing

- **Probabilistic design approaches showed great promise to achieve better design quality**
  - Compared to deterministic approaches, statistical circuit tuning achieved
    - 20% area reduction [Choi DAC04]
    - 17% power reduction [Mani DAC05]
- **Buffer insertion considering process variation is also gaining attention recently**
  - Limited consideration of process variation
    - Wire-length variation [Khandelwal ICCAD03]
  - Independency assumption of process variati
    - Ignores global and spatial correlati
  - High complexity
    - Numerical integration to obtai
  - Applicable to only special rout
    - Two-pin nets only [Deng ICCAD05]

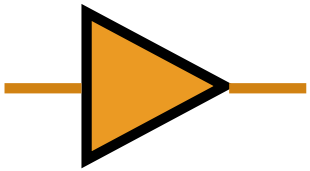
Our major contributions: theoretical foundations that lifts these limitations

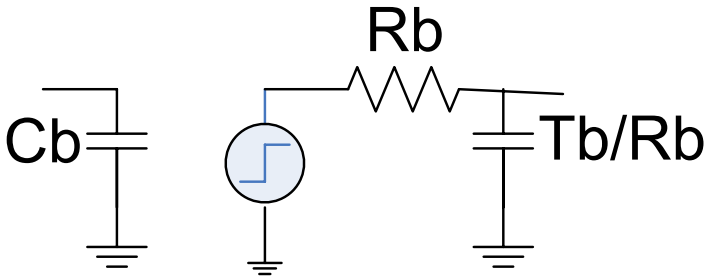
Agenda

- Introduction and motivation
- Modeling
- Problem formulation
- Detailed algorithms with complexity analysis
- Experimental results
- Conclusion

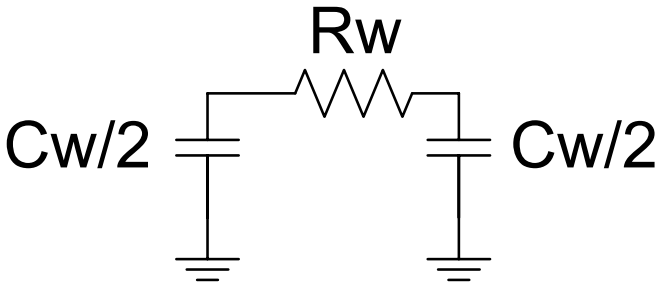
Modeling

- **Linear delay model for buffer**
  - Input capacitance (Cb), output resistance (Rb), and intrinsic delay (Tb)





- **π-model for interconnect**
  - Wire capacitance (Cw) and wire resistance (Rw)



- **How to model these quantities with correlated process variation?**

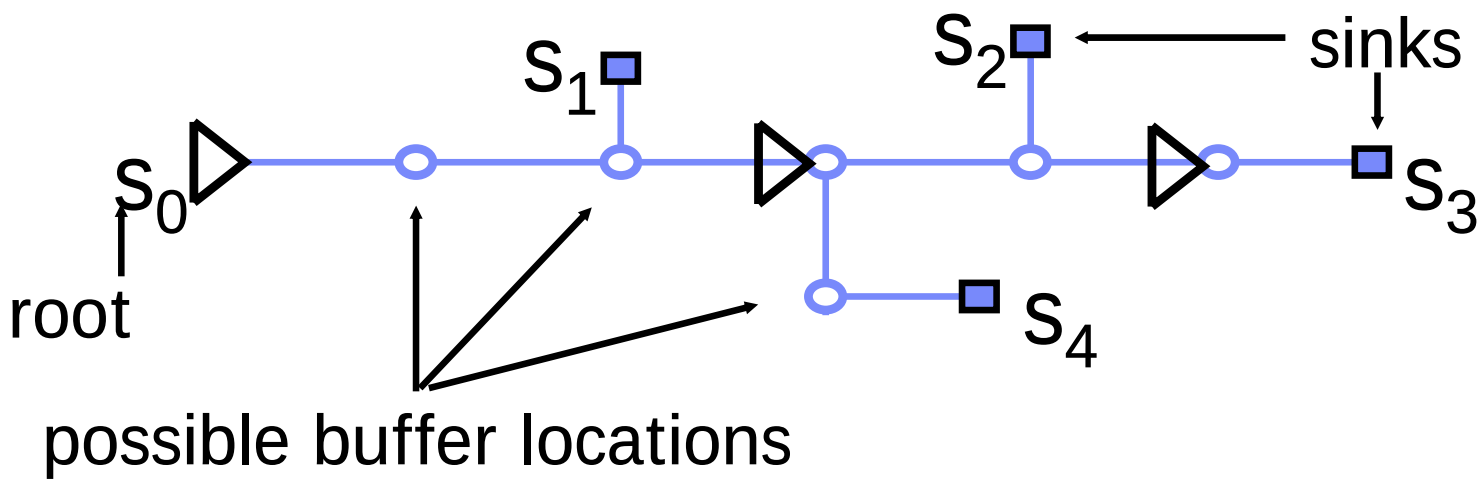
First-order Canonical Form for Variation Modeling

$$A = a_0 + a_1 X_1 + a_2 X_2 + \cdots + a_n X_n + a_R X_{Ra}$$

- **Mean value E(A) = a_0**
- **Random variables $X_1, X_2, \ldots, X_n$ model**
  - Die-to-die global variation: instances are affected in the same way
  - Within-die spatial correlation: instances physically nearby are more likely to be similar [Agarwal ASPDAC03, Chang ICCAD03, Khandelwal DAC05]
- **Random variable X_{Ra} model**
  - Independent variation: instances next to each other are different
- **All X_i follow independent normal distributions**
  - Well accepted practice in SSTA [Chang ICCAD03, Visweswariah DAC04]
- **In vector form, write device and interconnect with process variation**
  - Device: $T_b = T_{b0} + \gamma_b^T X$, $C_b = C_{b0} + \eta_b^T X$, $R_b = R_{b0} + \zeta_b^T X$
  - Interconnect: $C_w = C_{w0} + \eta_w^T X$, $R_w = R_{w0} + \zeta_w^T X$

Buffer Insertion Considering Process Variation

- **Given: a routing tree with required arrival time (RAT) and loading capacitance specified at sinks, and N possible buffer locations**
- **Considering: both FEOL device and BEOL interconnect process variations**
- **Find: locations to insert buffers**
- **So that: the timing slack at the root is maximized**
  - Timing slack: $\min_i (RAT_i - delay_i)$



Agenda

- **Introduction and motivation**
- **Modeling**
- **Problem formulation**
- **Detailed algorithms**
  - Key operations for buffering solutions
  - Transitive-closure pruning rule
  - Complexity analysis
- **Experimental results**
- **Conclusion**

Key Operations in Van Ginneken Algorithm

- **Associate each node with two metrics (C_t, T_t)**
  - Downstream loading capacitance (C_t) and RAT (T_t)
  - DP-based alg propagates potential solutions bottom-up [Van Ginneken, 90]
- **Add a wire**

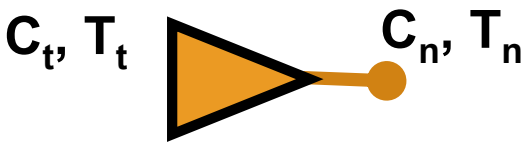
$$C_t = C_n + C_w$$

$$T_t = T_n - R_w \cdot L_n - \frac{1}{2} R_w \cdot C_w$$

- **Add a buffer**

$$C_t = C_b$$

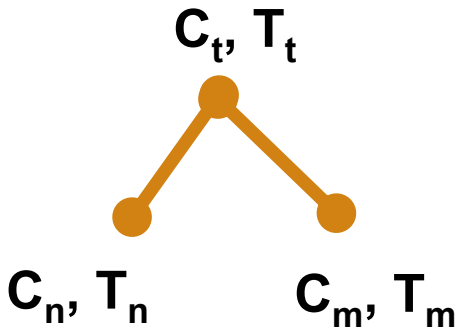
$$T_t = T_n - T_b - R_b \cdot L_n$$



- **Merge two solutions**

$$C_t = C_n + C_m$$

$$T_t = \min(T_n, T_m)$$



- **How to define these operations in statistical sense?**

Atomic Operations

- **Keep all quantities in canonical form after operations**
  - Maintain correlation w.r.t. sources of variation
  - Updated solutions can still be handled by the same set of operations
- **Add a wire**

$$C_t = C_n + C_w$$

$$T_t = T_n - R_w \cdot L_n - \frac{1}{2} R_w \cdot C_w$$

- **Add a buffer**

$$C_t = C_b$$

$$T_t = T_n - T_b - R_b \cdot L_n$$

- **Merge two solutions**

$$C_t = C_n + C_m$$

$$T_t = \min(T_n, T_m)$$

Addition/subtraction of two canonical forms is another canonical form

$$A + B = (a_0 + \alpha^T X) + (b_0 + \beta^T X)$$
$$= (a_0 + b_0) + (\alpha + \beta)^T X$$

Multiplications

No longer a canonical form

Minimum

Approximate Multiplication as Canonical Form

- **Multiplication of two canonical forms results in a quadratic term**

$$C = A \cdot B = (a_0 + \alpha^T X)(b_0 + \beta^T X)$$
$$= a_0 b_0 + (b_0 \alpha^T + a_0 \beta^T) X + X^T \alpha \beta^T X$$
$$= a_0 b_0 + \gamma^T X + X^T \Gamma X$$

  - Matrix $\Gamma = \alpha\beta^T$

- **Approximate it as a canonical form by matching the mean and variance with that of the exact solution**

$$C' = E(C) + \sqrt{\frac{E(C^2) - E(C)^2}{\gamma^T \gamma}} \gamma^T X = c_0 + \eta^T X$$

  - E(C) is the mean value (first moment) of C
  - $E(C^2)$ is the second moment of C→$E(C^2)$-$E(C)^2$ is the variance
  - C' is a new canonical form with the same mean and variance as C

Closed Form for Moment Computation

1st Moment

$$E(C) = c_0 + \gamma^T E(X) + E(X^T \Gamma X)$$

2nd Moment

$$E(C^2) = E(c_0^2 + 2c_0\gamma^T X + 2X^T \Gamma X \gamma^T X + 2c_0 X^T \Gamma X + X^T \gamma\gamma^T X + (X^T \Gamma X)^2)$$

$$= c_0^2 + 2c_0\gamma^T E(X) + 2E(X^T \Gamma X \gamma^T X) + E(X^T (2c_0\Gamma + \gamma\gamma^T)X) + E((X^T \Gamma X)^2)$$

- **Theorem: If X is an independent multivariate normal distribution ~N(0,I), then for any vector γ and matrix Γ**

$$E(X^T \Gamma X) = tr(\Gamma)$$

$$A \cdot B \approx C' = E(C) + \sqrt{\frac{E(C^2) - E(C)^2}{\gamma^T \gamma}}\gamma^T X = c_0 + \eta^T X$$

  - Trace of a matrix (tr) equals to the sum of all diagonal elements

- **In general, tr(Γ) and tr(Γ^2) are expensive, but if $\Gamma = \alpha\beta^T + \varepsilon\eta^T$ (a row rank matrix), we can show**

$$tr(\Gamma) = \beta^T \alpha + \eta\varepsilon^T, \quad tr(\Gamma^2) = (\beta^T \alpha)^2 + (\eta\varepsilon^T)^2 + 2(\beta^T \alpha)(\eta\varepsilon^T)$$

Approximate Minimum as Canonical Form

- **Minimum of two canonical forms is also not a canonical form**
- **Approximate it as a canonical from by matching the exact mean and variance**

$$\min(A,B) = c_0 + (T_A\beta^T + T_B\alpha^T)X + c_R X_R$$

  - Tightness probability of A: $T_A = P(A > B) = \Phi\left(\frac{a_0 - b_0}{\theta}\right)$
    - Φ is the CDF of a standard normal distribution
    - θ is given by $\theta = \sqrt{\sigma_A^2 + \sigma_B^2 - 2\,\mathrm{cov}(A,B)}$
  - Exact mean and variance can be computed in closed form [Clack 65]
    - Well known for statistical timing analysis
- **Design for mean value ≠ design for nominal value because of mean shift**

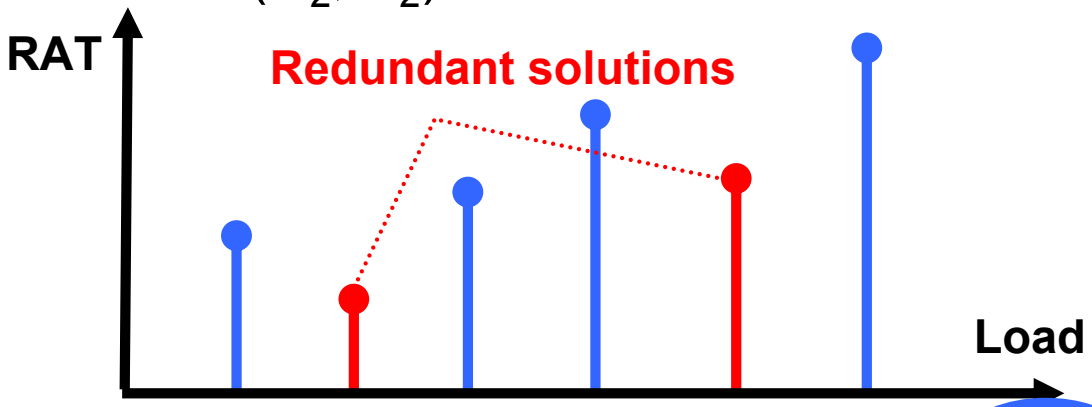
$$\underbrace{E(\min(A,B)) = T_A a_0 + T_B b_0 - \theta\phi\left(\frac{b_0 - a_0}{\theta}\right)}_{\text{Design for mean value}} \neq \underbrace{\min(a_0, b_0)}_{\text{Design for nominal value}}$$

Agenda

- **Introduction and motivation**
- **Modeling**
- **Problem formulation**
- **Detailed algorithms**
  - Key operations for buffering solutions
  - Transitive-closure pruning rule
  - Complexity analysis
- **Experimental results**
- **Conclusion**

Deterministic Pruning Rule

- **If $T_1 > T_2$ and $C_1 < C_2$ ➔(C_1, T_1) dominates (C_2, T_2)**
  - Dominated solution (C_2, T_2) is redundant



- **Deterministic pruning has linear time co… …of the following two desired properti…**
  - Ordering property
    - Either A>B or A<B holds
  - Transitive ordering (transitiv…
    - A>B, B>C → A>C
  - Make it possible to sort soluti…s in order
    - Assume sorted by load → linear time to prune redundant solutions

Can we achieve the same time complexity for statistical pruning?

Statistical Pruning Rule

- **(C_1, T_1) dominates $(C_2, T_2) \Leftrightarrow P(C_1 < C_2) \geq 0.5$ and $P(T_1 > T_2) \geq 0.5$**
- **Properties of this statistical pruning rule**
  - Ordering property
    - Given: T_1 and T_2 as two dependent random variables
    - Then: either $P(T_1>T_2) \geq 0.5$ or $P(T_1<T_2) \geq 0.5$ holds
  - Transitive-closure ordering property
    - Given T_1, T_2, and T_3 as three dependent random variables with a joint normal distribution,
    - If: $P(T_1>T_2) \geq 0.5$, $P(T_2>T_3) \geq 0.5$
    - Then: $P(T_1>T_3) \geq 0.5$
      - Transitive-closure property can be extended to the more general case
        > $P(T_1>T_2) \geq p$, $P(T_2>T_3) \geq p \rightarrow P(T_1>T_3) \geq p$ for any p 2 [0.5, 1]
- **Statistical pruning has the same linear time complexity as deterministic pruning**

Deterministic vs Statistical Buffering

For solution (C_n, T_n) in node t

Z_1 = ADD-WIRE(C_n, T_n);

Z_2 = ADD-BUF…(Z_1);

……

For s…

F…

(C_t, T_t)=MERGE(…m),(C_n, T_n));

……

Z = PRUNE(Z);

All quantities are deterministic values

For solution (C_n, T_n) in node t

Z_1 = STAT-ADD…WIRE(C_n, T_n);

Z_2 = STAT-ADD…ER(Z_1);

……

For soluti…

For sol…

(C_t, T_t)=STA…T_m),(C_n, T_n));

……

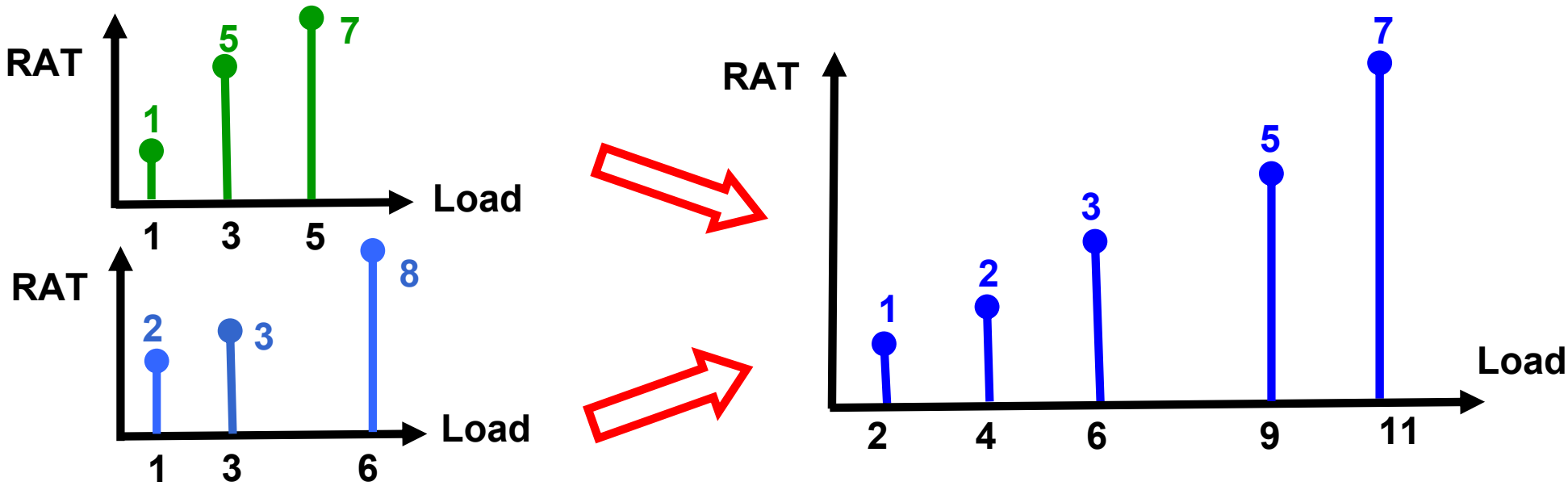
Z = STAT-PRUNE(Z);

All quantities are canonical forms

- **Same $O(N^2)$ complexity as the classic deterministic buffering algorithm**
- **Deterministic merge and pruning operations can be combined into one linear time operation**
  - New complexity result: $O(N*\log^2(N))$ [Wei, DAC 03]
- **Statistic merge and pruning can not be combined**
  - Statistic version's complexity is higher

When Merge and Prune can be Combined?

- **Made possible via merge-sort like operation in deterministic case**



  - Because of the following property: $\text{Min}(A_1,B_1) \le \text{Min}(A_2,B_1)$ if $A_1 \le A_2$
    - $\text{Min}(A,B) \le \text{Min}(A+\delta A,B) \rightarrow \text{Min}(A,B)$ is a nondecreasing function of inputs
- **In statistic case, such a property does not hold (even for mean)**

$$\frac{\partial E(\min(A,B))}{\partial E(A)} > 0 \qquad \frac{\partial E(\min(A,B))}{\partial \sigma_A} < 0 \qquad \frac{\partial E(\min(A,B))}{\partial \rho(A,B)} > 0$$

Agenda

- **Introduction and motivation**
- **Modeling**
- **Problem formulation**
- **Detailed algorithms**
  - Key operations for buffering solutions
  - Transitive-closure pruning rule
  - Complexity analysis
- **Experimental results**
- **Conclusion**

Experimental Setting

- **Variation setting**
  - Global, spatial, and independent variations all to be 5% of the nominal value
  - Spatial variation used a grid model similar to [Chang, ICCAD03]
    - Grid size 500um
    - Correlation distance about 2mm (beyond that, no spatial correlation)
- **Benchmarks**
  - Two sets of benchmarks from public domain [Shi, DAC03]
- **Deterministic design for worst case (WORST)**
  - All parameters projected to its respective 3-sigma values

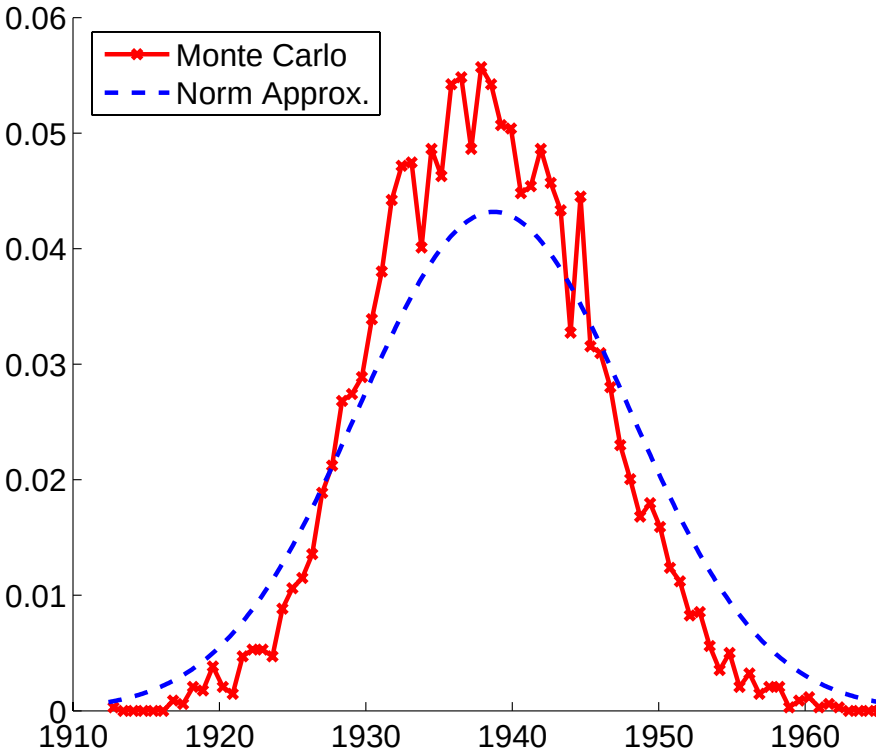
Runtime Comparison

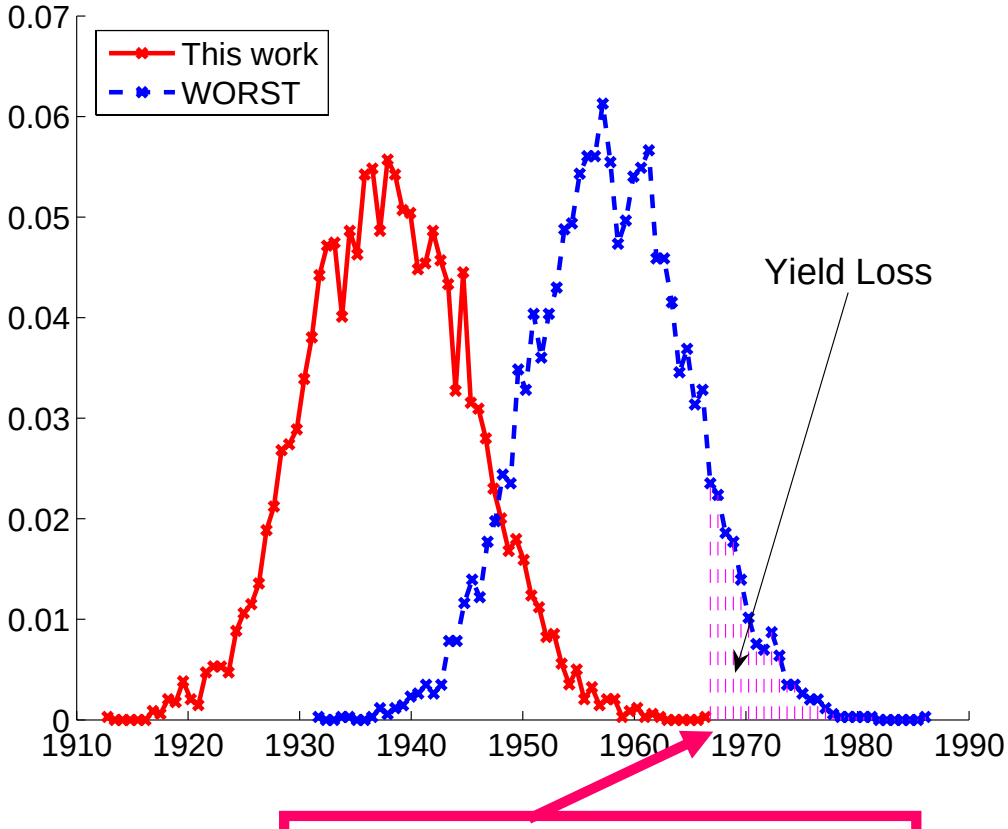
- **Compared with T2P proposed in [Xiong DATE05]**
  - Only known work that considered both device and interconnect variations
  - JPDF computed via expensive numerical integration
  - No global and spatial correlation considered
  - Heuristic pruning rules (T2P)
- **Re-implement T2P under the same first-order variation model, but still use its heuristic pruning rule**

Bench	Sink	Buf Loc	WORST	T2P	This work
p1	269	537	0	25.4	1.0 (25.4x)
p2	603	1205	0.01	-	4.3
r1	267	533	0	-	3.6
r2	598	1195	0	-	15.0
r3	862	1723	0.02	-	27.5
r4	1903	3805	0.04	-	88.9
r5	3101	6201	0.08	-	195.8

Monte Carlo Simulation Results

- **For a given a buffered routing tree with 10K MC runs, delay PDF at the root**
  - PDF from Monte Carlo roughly follows a normal distribution
  - Our approximation technique captures the PDF well





- **Figure-of-merits: 3-sigma delay vs yield loss**

3-sigma delay for red PDF

Timing Optimization Comparison based on MC

	WORST				This Work			
	Buffer	Mean	3-sigma Delay	Yield Loss	Buffer	mean	3-sigma Delay	Yield
p1	58	2374	2403	0%	60 (3.3%)	2375	2403	100%
p2	149	3161	3203	0%	156 (4.5%)	3161	3204	100%
r1	59	772	790	0%	65 (9.2%)	771	790	100%
r2	112	1109 (1.7%)	1128 (1.5%)	35.3%	135 (17%)	1090	1111	100%
r3	173	1127 (0.7%)	1147 (0.5%)	1.6%	188 (8%)	1119	1142	100%
r4	320	1700 (1.5%)	1723 (1.4%)	54.9%	374 (14.4%)	1674	1699	100%
r5	544	1958 (1%)	1986 (1.0%)	17.9%	608 (10.5%)	1938	1966	100%
Avg		0.7%	0.6%	15.7%	9.6%			

- **Buffer insertion considering process variation improves timing yield by 15% on average**
  - More effective for large benchmarks
  - Relative mean (or 3-sigma) delay improvement is small ←large mean values
  - More buffers are inserted in order to achieve this gain

Conclusion and Future Work

- **Developed a novel algorithm for buffer insertion considering process variation**
- **Two major theoretical contributions**
  - An effective approximation technique to handle nonlinear multiplication operation, all through closed form computation
  - A provably transitive-closure pruning rule
  - Maybe useful for other applications
- **Timing optimization shown that considering process variation can improve timing yield by more than 15%**
- **Future work**
  - Theoretically examine the impact of process variation on buffering
  - Apply the theories in this work to other design applications

Questions?

Thank You!