

Stochastic Physical Synthesis for FPGA Performance and Yield Optimization

EDA Lab, Electrical Engineering Department, UCLA
<http://eda.ee.ucla.edu>



Student: Yan Lin,
 Yu Hu
 Advisor: Lei He

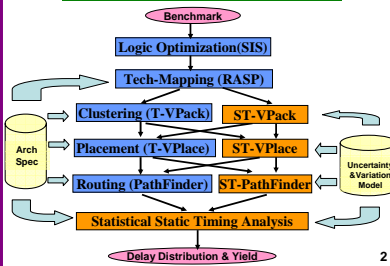
Abstract

Process variation and pre-routing interconnect delay uncertainty affect timing and power for modern VLSI designs in nanometer technologies. We study stochastic physical synthesis algorithms including clustering, placement and routing leveraging statistical timing analysis for FPGAs. The stochastic flow reduces the yield loss per 10K chip from 50 to 5 with the mean delay reduced by 6.2% and standard deviation reduced by 7.5%. In addition, we develop a statistical retiming with flip-flop constraints to further improve timing considering process variation.

Motivations

- Pre-routing interconnect uncertainty
 - Interconnect delay is dominant in modern VLSI designs
 - Timing-driven physical synthesis leverages timing slack
 - With interconnect delay estimated
 - May incur inaccuracy on critical path
 - Process variation
 - Any near-critical paths may be statistical timing critical
 - Slack is inaccurate with process variation
- ⇒ Stochastic physical synthesis is needed for FPGAs

Synthesis Framework



Uncertainty/Variation Model

- Pre-routing interconnect uncertainty
 - Modeled as independent Gaussian
 - Mean and standard deviation estimated
 - Evaluate with a fully placed and routed layout
- Process variation on threshold voltage (V_{th}) and effective channel length (L_{eff})
- Global/spatial/local variations modeled as Gaussians

Clustering Algorithms

- Timing-driven clustering T-VPack
 - Iteratively select a BLE for a new cluster
 - Pack BLE with the largest attraction into current cluster
 - The attraction function between BLE B and cluster C is

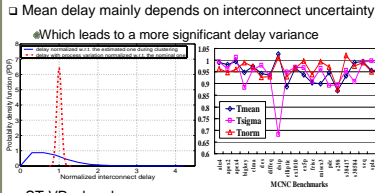
$$Attraction(B) = \lambda \cdot Criticality(B) + (1 - \lambda) \cdot \frac{Nets(B) \cap Nets(C)}{MaxNets}$$
 - STA is performed with constant delay model
- Stochastic clustering ST-VPack
 - Delay model with pre-routing uncertainty and variation

$$d = d_0 + \sigma_p \Delta R_p + \sigma_s \Delta R_s$$
 - ΔR_s models correlated process variation
 - ΔR_p models pre-routing interconnect uncertainty
 - σ_p and σ_s are standard deviations
 - Perform SSTA with statistical criticality calculation
 - The new statistical attraction function

$$Attraction(B) = \lambda \cdot SCriticality(B) + (1 - \lambda) \cdot \frac{Nets(B) \cap Nets(C)}{MaxNets}$$

Clustering Results

	I	II	III
σ_i	0.0	0.0	0.1
σ_p	0.0	0.1	0.0
Tmean (ns)	22.5	22.6	21.6
Tsigma (ns)	3.35	3.36	3.26



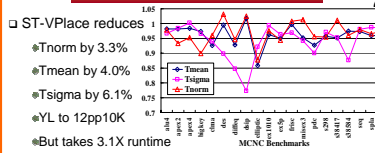
- ST-VPack reduces
 - Tnorm/Tmean/Tsigma by 3.7%/5.0%/6.4%
 - Yield loss (YL) from 50 to 9 pp10K
 - No routability, area and runtime overhead

Placement Algorithms

- Timing-driven placement T-VPace
 - Based on simulated annealing
 - Consider both wiring and timing costs
 - Timing Cost $T_{Cost}(i, j) = d(i, j) \cdot criticality(i, j)^\beta$
 - STA is performed at each annealing temperature
- Stochastic placement ST-VPace
 - Process variation is dominant
 - Estimate delay with variation
 - Perform SSTA
 - Statistical timing cost

$$STiming_Cost(i, j) = d(i, j) \cdot scriticality(i, j)^\beta$$

Placement Results



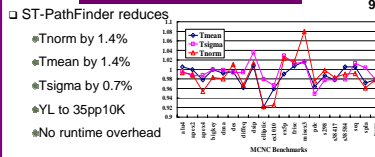
Routing Algorithms

- Timing-driven routing PathFinder
 - An iterative negotiated-based maze routing
 - Net by net routing, more critical sink routed first
 - The cost for node n when routing sink j of net i is

$$Cost(n) = Crit(i, j) \cdot delay(n) + [1 - Crit(i, j)]b(n)h(n)p(n)$$
 - $$Crit(i, j) = \max(\text{MaxCrit} - \frac{slack(i, j)}{D}, 0)$$
- Stochastic routing ST-PathFinder
 - Only consider dominant uncertainty, process variation
 - The new statistical cost function is

$$SCost(n) = SCrit(i, j) \cdot delay(n) + [1 - SCrit(i, j)]b(n)h(n)p(n)$$
 - $$SCrit(i, j) = \min(\text{MaxCrit}, Scriticality(i, j)^\beta)$$
 - Timing & wiring tradeoff and sink order are changed

Routing Results



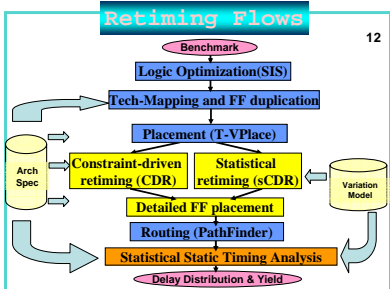
Interaction

Clusterer	D	S	D	D	S	S	D	S
Placer	D	D	S	D	S	D	S	S
Router	D	D	D	S	D	S	S	S
Tnorm(ns)	21.2	-3.7%	-3.3%	-1.4%	-6.4%	-4.1%	-3.6%	-6.3%
Tmean(ns)	22.9	-5.0%	-4.0%	-1.4%	-5.9%	-4.7%	-4.0%	-6.2%
Tsigma(ns)	2.4	-6.4%	-6.1%	-0.7%	-8.8%	-6.1%	-6.3%	-7.5%
Yield loss	50.2	9.3	11.8	35.2	5.3	10.3	11.0	5.4
runtime	1X	0.99X	3.1X	0.96X	3.0X	0.97X	3.1X	3.0X

- The stochastic flow reduces yield loss to 5 with 3X runtime
 - Gain mainly comes from clustering and placement with overlap between gains
- A good flow: stochastic clustering + deterministic place&route
 - Similar gain but no runtime overhead

Retiming Formulations

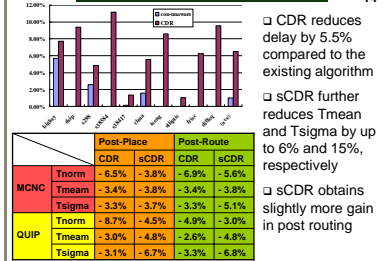
- Retiming graph
 - Vertices: the I/O of basic circuit elements
 - Edges: the local/global interconnect
 - Delay $d(e)$ and FF $w(e)$ are associated with each edge e
- Given the retiming graph $G=(V,E)$, a retiming is an integral-valued vertex-labeling $r: V \rightarrow Z$
- CLB-based FF constraint: an LUT can drive all FFs within its CLB, but not FFs outside its CLB



Retiming Algorithms

- sCDR
 - Inputs: circuit and variation
 - Outputs: retimed circuit
 - Load retiming graph $G=(E,V)$
 - Initialize variation model
 - $T = \text{Stat_Seq_Timer}, \theta = \text{counter}$
 - FIFO queue Q initialization
 - While $\text{counters} \leftarrow \text{MaxIter}$
 - Exit if critical edges forms a cycle
 - Push all critical nodes to Q
 - While Q is not empty
 - Backward move FFs
 - Legalize retiming solution
 - $T = \text{Stat_Seq_Timer}$
 - Push all critical nodes to Q
 - Detailed FF placement
- Retiming performance:
 - $disu(x) = E(x) + w \cdot V(x)$
 - Critical nodes:
 - $Seq_crit(x) = disu(x) \cdot crit(x)$
- Detailed FF placement

Retiming Results



References

- Y. Lin, M. Hutton and L. He, "Placement and Timing for FPGAs Considering Variations"
- Y. Lin, M. Hutton and L. He, "Stochastic Physical Synthesis for FPGAs with Pre-routing Uncertainty and Process Variation"
- Y. Hu, Y. Lin and L. He, "Retiming for High Performance FPGAs Considering Flip-flop Constraints and Process Variations"