Risk Aversion Min-Period Retiming Under Process Variations

Jia Wang
Illinois Institute of Technology
Chicago, Illinois, USA

Hai Zhou
Fudan University, China
Northwestern University, USA

January, 2009

Outline

Process Variations

- VLSI feature sizes keep shrinking with technology scaling.
- Process variations become significant.
  - Chips are realized randomly during manufacturing around nominal values, instead of precisely as expected by the chip designers.
- Affect manufacturing yield and system reliability.
- A critical issue of Design for Manufacturability (DFM)
  - Statistical circuit analysis: Monte Carlo, SSTA
  - Statistical circuit optimization

Statistical Circuit Optimization

- Statistical optimization is difficult
  - Optimize a circuit for many or infinity number of variation corners at the same time.
- Gate sizing under process variations
  - Majority of previous statistical circuit optimization works.
  - Sensitivity guided iterative improvement heuristics
  [Guthaus et al. 2005], [Sinha et al. 2006], [Srivastava et al. 2008]
  - Convex optimization for the worst cases
   [Mani et al. 2007], [Singh et al. 2008]
  - Two-stage stochastic program with fixed recourse [Davoodi et al. 2006]
- Min-period retiming under process variations
  - Iterative push-down heuristics [Wang et al. 2004]
  - No theoretical guarantee for optimality

Statistical Circuit Optimization

- Statistical optimization is difficult
  - Optimize a circuit for many or infinity number of variation corners at the same time.
- Gate sizing under process variations
  - Majority of previous statistical circuit optimization works.
  - Sensitivity guided iterative improvement heuristics
  [Guthaus et al. 2005], [Sinha et al. 2006], [Srivastava et al. 2008]
  - Convex optimization for the worst cases
  [Mani et al. 2007], [Singh et al. 2008]
  - Two-stage stochastic program with fixed recourse [Davoodi et al. 2006]
- Min-period retiming under process variations
  - Iterative push-down heuristics [Wang et al. 2004]
  - No theoretical guarantee for optimality

Statistical Circuit Optimization

- Statistical optimization is difficult
  - Optimize a circuit for many or infinity number of variation corners at the same time.
- Gate sizing under process variations
  - Majority of previous statistical circuit optimization works.
  - Sensitivity guided iterative improvement heuristics [Guthaus et al. 2005], [Sinha et al. 2006], [Srivastava et al. 2008]
  - Convex optimization for the worst cases [Mani et al. 2007], [Singh et al. 2008]
  - Two-stage stochastic program with fixed recourse [Davoodi et al. 2006]
- Min-period retiming under process variations
  - Iterative push-down heuristics [Wang et al. 2004]
  - No theoretical guarantee for optimality

Our Contribution

- Formulate the Risk Aversion Min-Period Retiming problem
  - Based on two-stage stochastic programs with fixed recourse and the conditional value-at-risk measure
- Prove that the proposed formulation is an integer convex program
- Derive an analytical formula for the subgradient of the objective function.
  - Simplify sensitivity computation
  - Handle arbitrary distribution by sampling
- Present an incremental algorithm to solve the proposed problem heuristically.

Deterministic Retiming

- Relocate flip-flops (FFs) w/o changing circuit functionality. [Leiserson and Saxe 83]
  - Powerful sequential transformation to reschedule both computation and communication.
- Circuit graph $G = (V, E)$
  - Gate delay: $d(v)$, $v \in V$
  - # FFs on interconnects: $w(u, v)$, $(u, v) \in E$
- Retiming is represented by an integer-valued vertex label.

$$r : V \to \mathbb{Z}$$

For gate v, $r(v)$ is the # FFs moved from all its fanouts to all its fanins.

Deterministic Retiming

- Relocate flip-flops (FFs) w/o changing circuit functionality. [Leiserson and Saxe 83]
  - Powerful sequential transformation to reschedule both computation and communication.
- Circuit graph $G = (V, E)$
  - Gate delay: $d(v)$, $v \in V$
  - # FFs on interconnects: $w(u, v)$, $(u, v) \in E$
- Retiming is represented by an integer-valued vertex label.

$$r : V \rightarrow \mathbb{Z}$$

For gate v, $r(v)$ is the # FFs moved from all its fanouts to all its fanins.

Deterministic Retiming

- Relocate flip-flops (FFs) w/o changing circuit functionality. [Leiserson and Saxe 83]
  - Powerful sequential transformation to reschedule both computation and communication.
- Circuit graph $G = (V, E)$
  - Gate delay: $d(v)$, $v \in V$
  - # FFs on interconnects: $w(u, v)$, $(u, v) \in E$
- Retiming is represented by an integer-valued vertex label.

$$r : V \to \mathbb{Z}$$

For gate v, $r(v)$ is the # FFs moved from all its fanouts to all its fanins.

Deterministic Min-Period Retiming

- Validity constraints:

$$\forall (u, v) \in E : w(u, v) - r(u) + r(v) \geq 0$$

- Timing constraints for the clock period ϕ:

$$\forall (u, v) \in E : w(u, v) = r(u) - r(v) \Rightarrow t(u) + d(v) \leq t(v)$$

$$\forall v \in V : d(v) \leq t(v) \leq \phi$$

- Min-period objective: find r to minimize ϕ

Deterministic Min-Period Retiming

- Validity constraints:

$$\forall (u, v) \in E : w(u, v) - r(u) + r(v) \geq 0$$

- Timing constraints for the clock period ϕ:

$$\forall (u, v) \in E : w(u, v) = r(u) - r(v) \Rightarrow t(u) + d(v) \leq t(v)$$

$$\forall v \in V : d(v) \leq t(v) \leq \phi$$

- Min-period objective: find r to minimize ϕ

Deterministic Min-Period Retiming

- Validity constraints:

$$\forall (u, v) \in E : w(u, v) - r(u) + r(v) \geq 0$$

- Timing constraints for the clock period ϕ:

$$\forall (u, v) \in E : w(u, v) = r(u) - r(v) \Rightarrow t(u) + d(v) \leq t(v)$$

$$\forall v \in V : d(v) \leq t(v) \leq \phi$$

- Min-period objective: find r to minimize ϕ

Two-Stage Stochastic Program with Fixed Recourse

- Decision making under uncertainty
  - The output depends on decision variables and uncertain parameters not available at the time of decision.
  - E.g. designer specifies circuit parameters at design time – the fabricated chips will be affected by process variations
- First stage: determine the values of decision variables
  - Incur an initial cost
- Second stage: uncertain parameters are realized
  - Incur a second stage cost through a fixed recourse, i.e.,
  - A known deterministic program of both the decision variables and the realized uncertain parameters
- Objective: "minimize" the total cost
  - The total cost is a random number.
  - What is the meaning of "minimize"?

Two-Stage Stochastic Program with Fixed Recourse

- Decision making under uncertainty
  - The output depends on decision variables and uncertain parameters not available at the time of decision.
  - E.g. designer specifies circuit parameters at design time – the fabricated chips will be affected by process variations
- First stage: determine the values of decision variables
  - Incur an initial cost
- Second stage: uncertain parameters are realized
  - Incur a second stage cost through a fixed recourse, i.e.,
  - A known deterministic program of both the decision variables and the realized uncertain parameters
- Objective: "minimize" the total cost
  - The total cost is a random number.
  - What is the meaning of "minimize"?

Two-Stage Stochastic Program with Fixed Recourse

- Decision making under uncertainty
  - The output depends on decision variables and uncertain parameters not available at the time of decision.
  - E.g. designer specifies circuit parameters at design time – the fabricated chips will be affected by process variations
- First stage: determine the values of decision variables
  - Incur an initial cost
- Second stage: uncertain parameters are realized
  - Incur a second stage cost through a fixed recourse, i.e.,
  - A known deterministic program of both the decision variables and the realized uncertain parameters
- Objective: "minimize" the total cost
  - The total cost is a random number.
  - What is the meaning of "minimize"?

Two-Stage Stochastic Program with Fixed Recourse

- Decision making under uncertainty
  - The output depends on decision variables and uncertain parameters not available at the time of decision.
  - E.g. designer specifies circuit parameters at design time – the fabricated chips will be affected by process variations
- First stage: determine the values of decision variables
  - Incur an initial cost
- Second stage: uncertain parameters are realized
  - Incur a second stage cost through a fixed recourse, i.e.,
  - A known deterministic program of both the decision variables and the realized uncertain parameters
- Objective: "minimize" the total cost
  - The total cost is a random number.
  - What is the meaning of "minimize"?

Coherent Measure of Risk

- Measure of risk: map a random variable into a real number
  - $\mathrm{E}\left[X\right]$, expectation of X
  - $\mathrm{Yield}_{\phi}[X] \triangleq P(X \leq \phi)$
- Coherent measure of risk [Rockafellar 2007]
  - Transfer the convexity of the second stage program to the objective function.
  - $\mathrm{E}\left[X\right]$ is coherent, but not quite interesting.
- Conditional value-at-risk for a risk aversion level α

$$\mathrm{CVaR}_{\alpha}[X] \triangleq \mathrm{E}\left[X | X > \mathrm{VaR}_{\alpha}[X]\right]$$

where $\mathrm{VaR}_{\alpha}[X]$ is the value satisfying $P(X \leq \mathrm{VaR}_{\alpha}[X]) = \alpha$

Coherent Measure of Risk

- Measure of risk: map a random variable into a real number
  - $\mathrm{E}\big[X\big]$, expectation of X
  - $\mathrm{Yield}_\phi[X] \triangleq P(X \leq \phi)$
- Coherent measure of risk [Rockafellar 2007]
  - Transfer the convexity of the second stage program to the objective function.
  - $\mathrm{E}\big[X\big]$ is coherent, but not quite interesting.
- Conditional value-at-risk for a risk aversion level α

$$\mathrm{CVaR}_\alpha[X] \triangleq \mathrm{E}\big[X | X > \mathrm{VaR}_\alpha[X]\big]$$

where $\mathrm{VaR}_\alpha[X]$ is the value satisfying $P(X \leq \mathrm{VaR}_\alpha[X]) = \alpha$

Coherent Measure of Risk

- Measure of risk: map a random variable into a real number
  - $\mathrm{E}\big[X\big]$, expectation of X
  - $\mathrm{Yield}_\phi[X] \triangleq P(X \leq \phi)$
- Coherent measure of risk [Rockafellar 2007]
  - Transfer the convexity of the second stage program to the objective function.
  - $\mathrm{E}\big[X\big]$ is coherent, but not quite interesting.
- Conditional value-at-risk for a risk aversion level α

$$\mathrm{CVaR}_\alpha[X] \triangleq \mathrm{E}\big[X | X > \mathrm{VaR}_\alpha[X]\big]$$

where $\mathrm{VaR}_\alpha[X]$ is the value satisfying $P(X \leq \mathrm{VaR}_\alpha[X]) = \alpha$

Outline

Gate Delay under Process Variations

- Probabilistic space for process variations: Ω
- For a particular variation $\omega \in \Omega$,
  - Random gate delays $d_\omega : V \rightarrow \mathbb{R}^*$
  - Minimum clock period for valid retiming r: $\phi_\omega(r)$

Risk Aversion Min-Period Retiming

Risk aversion min-period retiming for risk aversion level α

Minimize $\mathrm{CVaR}_\alpha[\phi_\omega(r)]$ s.t.

$$w_r(u,v) \geq 0, \forall (u,v) \in E, \text{ and } r(v) \in \mathbb{Z}, \forall v \in V,$$

where $\phi_\omega(r)$ is the minimum objective of

Minimize ϕ s.t.

$$w_r(u,v) = 0 \Rightarrow t(v) \geq d_\omega(v) + t(u), \forall (u,v) \in E,$$

$$d_\omega(v) \leq t(v) \leq \phi, \forall v \in V.$$

Continuous Relaxation

- The second stage program is not mathematical programming.
- Compute $\phi_\omega(r)$ by enumerating paths

$$\exists \text{ simple path } p^* \text{ in } G, \phi_\omega(r) = d_\omega(p^*) \wedge w_r(p^*) = 0,$$
$$\forall \text{ simple path } p \text{ in } G, \phi_\omega(r) \geq \frac{d_\omega(p)}{w_r(p)+1}.$$

- Continuous relaxation

$$\text{Minimize } \mathrm{CVaR}_\alpha[\phi_\omega(r)] \text{ s.t. } w_r(u,v) \geq 0, \forall (u,v) \in E,$$

where $\phi_\omega(r) = \max_{\text{simple path } p \text{ in } G} \frac{d_\omega(p)}{w_r(p)+1}$.

Continuous Relaxation

- The second stage program is not mathematical programming.
- Compute $\phi_\omega(r)$ by enumerating paths

$$\exists \text{ simple path } p^* \text{ in } G, \phi_\omega(r) = d_\omega(p^*) \wedge w_r(p^*) = 0,$$
$$\forall \text{ simple path } p \text{ in } G, \phi_\omega(r) \geq \frac{d_\omega(p)}{w_r(p)+1}.$$

- Continuous relaxation

$$\text{Minimize } \mathrm{CVaR}_\alpha[\phi_\omega(r)] \text{ s.t. } w_r(u,v) \geq 0, \forall (u,v) \in E,$$

$$\text{where } \phi_\omega(r) = \max_{\text{simple path } p \text{ in } G} \tfrac{d_\omega(p)}{w_r(p)+1}.$$

Continuous Relaxation

- The second stage program is not mathematical programming.
- Compute $\phi_\omega(r)$ by enumerating paths

$$\exists \text{ simple path } p^* \text{ in } G, \phi_\omega(r) = d_\omega(p^*) \wedge w_r(p^*) = 0,$$
$$\forall \text{ simple path } p \text{ in } G, \phi_\omega(r) \geq \frac{d_\omega(p)}{w_r(p)+1}.$$

- Continuous relaxation

$$\text{Minimize } \mathrm{CVaR}_\alpha[\phi_\omega(r)] \text{ s.t. } w_r(u,v) \geq 0, \forall (u,v) \in E,$$

$$\text{where } \phi_\omega(r) = \max_{\text{simple path } p \text{ in } G} \tfrac{d_\omega(p)}{w_r(p)+1}.$$

Convexity of the Minimal Clock Period

- p_ω: critical path for retiming r from u_ω to v_ω, i.e.

$$\phi_\omega(r) = \frac{d_\omega(p_\omega)}{w_r(p_\omega) + 1}$$

- Intuition: inserting(removing) 1 FF to(from) p_ω will decrease(increase) $\phi_\omega(r)$ by at most(least) $\frac{\phi_\omega(r)}{w_r(p_\omega)+1}$
- Let $s_\omega(u_\omega) = 1$, $s_\omega(v_\omega) = -1$, and $s_\omega(x) = 0$ for any other $x \in V$.

$$\phi_\omega(r') - \phi_\omega(r) \geq \sum_{u \in V} \frac{\phi_\omega(r) s_\omega(u)}{w_r(p_\omega) + 1}(r'(u) - r(u)).$$

Convexity of the Minimal Clock Period

- p_ω: critical path for retiming r from u_ω to v_ω, i.e.

$$\phi_\omega(r) = \frac{d_\omega(p_\omega)}{w_r(p_\omega) + 1}$$

- Intuition: inserting(removing) 1 FF to(from) p_ω will decrease(increase) $\phi_\omega(r)$ by at most(least) $\frac{\phi_\omega(r)}{w_r(p_\omega)+1}$
- Let $s_\omega(u_\omega) = 1$, $s_\omega(v_\omega) = -1$, and $s_\omega(x) = 0$ for any other $x \in V$.

$$\phi_\omega(r') - \phi_\omega(r) \geq \sum_{u \in V} \frac{\phi_\omega(r) s_\omega(u)}{w_r(p_\omega) + 1} (r'(u) - r(u)).$$

Convexity of the Minimal Clock Period

- p_ω: critical path for retiming r from u_ω to v_ω, i.e.

$$\phi_\omega(r) = \frac{d_\omega(p_\omega)}{w_r(p_\omega) + 1}$$

- Intuition: inserting(removing) 1 FF to(from) p_ω will decrease(increase) $\phi_\omega(r)$ by at most(least) $\frac{\phi_\omega(r)}{w_r(p_\omega)+1}$
- Let $s_\omega(u_\omega) = 1$, $s_\omega(v_\omega) = -1$, and $s_\omega(x) = 0$ for any other $x \in V$.

$$\phi_\omega(r') - \phi_\omega(r) \geq \sum_{u \in V} \frac{\phi_\omega(r) s_\omega(u)}{w_r(p_\omega) + 1} \big(r'(u) - r(u)\big).$$

Convexity of the Continuous Relaxation

- $I_\omega(r)$: 1 if $\phi_\omega(r) \geq \mathrm{VaR}_\alpha[\phi_\omega(r)]$ and 0 otherwise.
- Convexity of the objective function from that of $\phi_\omega(r)$

$$\begin{aligned}&\mathrm{CVaR}_\alpha[\phi_\omega(r')] - \mathrm{CVaR}_\alpha[\phi_\omega(r)] \\ &\quad \geq \sum_{u \in V} \frac{r'(u) - r(u)}{1-\alpha} \mathrm{E}\big[I_\omega(r) \frac{\phi_\omega(r) s_\omega(u)}{w_r(p_\omega)+1}\big]\end{aligned}$$

- Convex constraints: $w_r(u,v) \geq 0, \forall (u,v) \in E$
- The continuous relaxation is a convex program.
- Risk aversion min-period retiming requires an integer optimal solution of the continuous relaxation.

Convexity of the Continuous Relaxation

- $I_\omega(r)$: 1 if $\phi_\omega(r) \geq \mathrm{VaR}_\alpha[\phi_\omega(r)]$ and 0 otherwise.
- Convexity of the objective function from that of $\phi_\omega(r)$

$$\begin{aligned}&\mathrm{CVaR}_\alpha[\phi_\omega(r')] - \mathrm{CVaR}_\alpha[\phi_\omega(r)] \\ &\quad \geq \sum_{u \in V} \frac{r'(u) - r(u)}{1-\alpha} \mathrm{E}\left[I_\omega(r) \frac{\phi_\omega(r) s_\omega(u)}{w_r(p_\omega)+1}\right]\end{aligned}$$

- Convex constraints: $w_r(u,v) \geq 0, \forall (u,v) \in E$
- The continuous relaxation is a convex program.
- Risk aversion min-period retiming requires an integer optimal solution of the continuous relaxation.

Convexity of the Continuous Relaxation

- $I_\omega(r)$: 1 if $\phi_\omega(r) \geq \mathrm{VaR}_\alpha[\phi_\omega(r)]$ and 0 otherwise.
- Convexity of the objective function from that of $\phi_\omega(r)$

$$\begin{aligned} &\mathrm{CVaR}_\alpha[\phi_\omega(r')] - \mathrm{CVaR}_\alpha[\phi_\omega(r)] \\ &\quad \geq \sum_{u \in V} \frac{r'(u) - r(u)}{1-\alpha} \mathrm{E}\big[I_\omega(r) \frac{\phi_\omega(r) s_\omega(u)}{w_r(p_\omega)+1} \big] \end{aligned}$$

- Convex constraints: $w_r(u,v) \geq 0, \forall (u,v) \in E$
- The continuous relaxation is a convex program.
- Risk aversion min-period retiming requires an integer optimal solution of the continuous relaxation.

Outline

Subgradient Computation

- Subgradient of $\mathrm{CVaR}_\alpha[\phi_\omega(r)]$ was derived when proving its convexity.
- Compute subgradient by drawing independent samples from a black box model representing Ω.
  - Can handle arbitrary distribution of process variations.
  - Reuse existing deterministic analysis algorithms.
  - May develop analytical methods to speed up computation for specific distributions.
- Most previous works on statistical gate sizing approximated subgradient (sensitivity) by computing secant directions.
  - Require multiple runs of SSTA – time consuming.
  - Trade-off accuracy for running time.

Subgradient Computation

- Subgradient of $\mathrm{CVaR}_\alpha[\phi_\omega(r)]$ was derived when proving its convexity.
- Compute subgradient by drawing independent samples from a black box model representing Ω.
  - Can handle arbitrary distribution of process variations.
  - Reuse existing deterministic analysis algorithms.
  - May develop analytical methods to speed up computation for specific distributions.
- Most previous works on statistical gate sizing approximated subgradient (sensitivity) by computing secant directions.
  - Require multiple runs of SSTA – time consuming.
  - Trade-off accuracy for running time.

Subgradient Computation

- Subgradient of $\mathrm{CVaR}_\alpha[\phi_\omega(r)]$ was derived when proving its convexity.
- Compute subgradient by drawing independent samples from a black box model representing Ω.
  - Can handle arbitrary distribution of process variations.
  - Reuse existing deterministic analysis algorithms.
  - May develop analytical methods to speed up computation for specific distributions.
- Most previous works on statistical gate sizing approximated subgradient (sensitivity) by computing secant directions.
  - Require multiple runs of SSTA – time consuming.
  - Trade-off accuracy for running time.

Subgradient Guided Optimization

- An intuitive idea: iteratively incremental improvement

$$\text{Minimize } \sum_{v \in V} \hat{g}_r(v)\big(r'(v) - r(v)\big) \text{ s.t.}$$
$$w_{r'}(u,v) \geq 0, \forall (u,v) \in E, \text{ and } 0 \leq r'(v) - r(v) \leq 1, \forall v \in V,$$

  where $\hat{g}_r$ is the subgradient.
  - Dual of network-flow problem: integer-valued optimal solution
  - Not good in practice since even changing r by 1 for some vertices will result in huge changes in the minimum clock period.
  - Need additional constraints to improve the accuracy of the estimation.
- Cutting plane techniques cannot guarantee an integer-valued optimal solution.

Subgradient Guided Optimization

- An intuitive idea: iteratively incremental improvement

$$\text{Minimize } \sum_{v \in V} \hat{g}_r(v)\big(r'(v) - r(v)\big) \text{ s.t.}$$
$$w_{r'}(u,v) \geq 0, \forall (u,v) \in E, \text{ and } 0 \leq r'(v) - r(v) \leq 1, \forall v \in V,$$

  where $\hat{g}_r$ is the subgradient.
  - Dual of network-flow problem: integer-valued optimal solution
  - Not good in practice since even changing r by 1 for some vertices will result in huge changes in the minimum clock period.
  - Need additional constraints to improve the accuracy of the estimation.
- Cutting plane techniques cannot guarantee an integer-valued optimal solution.

Statistical Timing Critical Path

- Use statistical timing critical paths as additional constraints without affecting optimality

$$w_{r'}(p) \geq 1, \forall\ p \text{ satisfying } \mathrm{CVaR}_\alpha[d_\omega(p)] > \mathrm{CVaR}_\alpha[\phi_\omega(r)].$$

- Remain dual of network-flow problem – integer-valued optimal solution
- Inefficient in practice, replace with

$$w_{r'}(p) \geq 1, \forall \text{ simple path } p \text{ satisfying } \overline{d}(p) > \beta\overline{\phi}(r).$$

where $\overline{d}$ are the nominal delays, $\overline{\phi}(r)$ is the nominal minimum clock period, and $\beta \geq 1$ is a parameter specified by the designer.

Statistical Timing Critical Path

- Use statistical timing critical paths as additional constraints without affecting optimality

$$w_{r'}(p) \geq 1, \forall\ p \text{ satisfying } \mathrm{CVaR}_\alpha[d_\omega(p)] > \mathrm{CVaR}_\alpha[\phi_\omega(r)].$$

- Remain dual of network-flow problem – integer-valued optimal solution
- Inefficient in practice, replace with

$$w_{r'}(p) \geq 1, \forall \text{ simple path } p \text{ satisfying } \overline{d}(p) > \beta\overline{\phi}(r).$$

where $\overline{d}$ are the nominal delays, $\overline{\phi}(r)$ is the nominal minimum clock period, and $\beta \geq 1$ is a parameter specified by the designer.

Statistical Timing Critical Path

- Use statistical timing critical paths as additional constraints without affecting optimality

$$w_{r'}(p) \geq 1, \forall\ p \text{ satisfying } \mathrm{CVaR}_\alpha[d_\omega(p)] > \mathrm{CVaR}_\alpha[\phi_\omega(r)].$$

- Remain dual of network-flow problem – integer-valued optimal solution
- Inefficient in practice, replace with

$$w_{r'}(p) \geq 1, \forall \text{ simple path } p \text{ satisfying } \overline{d}(p) > \beta\overline{\phi}(r).$$

where $\overline{d}$ are the nominal delays, $\overline{\phi}(r)$ is the nominal minimum clock period, and $\beta \geq 1$ is a parameter specified by the designer.

Incremental Risk Aversion Retiming

- Incremental risk aversion retiming

$$\text{Minimize} \sum_{v \in V} \hat{g}_r(v)\big(r'(v) - r(v)\big) \text{ s.t.}$$
$$w_{r'}(u,v) \geq 0, \forall (u,v) \in E, \text{ and } 0 \leq r'(v) - r(v) \leq 1, \forall v \in V,$$
$$w_{r'}(p) \geq 1, \forall \text{ simple path } p \text{ satisfying } \overline{d}(p) > \beta\overline{\phi}(r).$$

- Similar to deterministic min-area retiming
- Can be solved by incremental deterministic min-area retiming algorithm [Wang et al. 2008]

Incremental Risk Aversion Retiming

- Incremental risk aversion retiming

$$\text{Minimize} \sum_{v \in V} \hat{g}_r(v)\big(r'(v) - r(v)\big) \text{ s.t.}$$
$$w_{r'}(u, v) \geq 0, \forall (u, v) \in E, \text{ and } 0 \leq r'(v) - r(v) \leq 1, \forall v \in V,$$
$$w_{r'}(p) \geq 1, \forall \text{ simple path } p \text{ satisfying } \overline{d}(p) > \beta\overline{\phi}(r).$$

- Similar to deterministic min-area retiming
- Can be solved by incremental deterministic min-area retiming algorithm [Wang et al. 2008]

The Incremental Risk Aversion Retiming Algorithm

- Iteratively improve any initial valid retiming r.
- In each iteration
  - Compute $\mathrm{CVaR}_\alpha[\phi_\omega(r)]$ and record the best retiming so far
  - Compute the subgradient
  - Formulate and solve the incremental risk aversion retiming problem for r'
  - Claim optimality if $r' = r$ and stop
  - Update r to r'. Stop if a predefined number of iterations have been reached.

Outline

Experimental Setup

- Benchmarks: ISCAS89 sequential circuits
  - Placed onto a 4x4 grid to build a model for process variations
- Compared to a risk-aware deterministic approach similar to [Wang et al. 2004]
  - Assign each gate a delay of $\mathrm{E}\big[d_\omega(v)\big] + \gamma\sqrt{\mathrm{E}\big[(d_\omega(v) - \mathrm{E}\big[d_\omega(v)\big])^2\big]}$
  - Run [Zhou 2005] for a min-period retiming
  - Take the one with the best CVaR for $\gamma = 0, 1, 3$

Experimental Results

	Statistics		CVaR of Det.		Ours		
name	$\|V\|$	$\|E\|$	init	best	CVaR	impr.	t(s)
s1196	530	1023	66.75	65.95	65.88	0.11%	0.3
s1238	509	1055	70.67	70.67	70.66	0.01%	0.3
s1423	658	1169	211.37	160.50	158.91	0.99%	1.5
s1488	654	1406	190.86	161.60	161.60	0.00%	6.7
s1494	648	1412	196.48	174.32	174.32	0.00%	6.6
s5378	2780	4261	66.64	66.64	64.05	3.89%	29.4
s9234.1	5598	4604	114.28	114.28	113.63	0.57%	50.6
s13207.1	7952	11082	193.40	122.62	118.85	3.08%	120.8
s15850.1	9773	13566	243.37	119.51	109.88	8.06%	156.0
s35932	16066	28589	187.44	170.62	171.84	-0.71%	265.3
s38417	22180	31127	173.83	96.62	93.69	3.03%	457.3
s38584.1	19254	33060	267.32	245.73	242.41	1.35%	445.3
			runtimes $\leq$ 1s				

Experimental Result: Timing Yield

	Deterministic Approach				Ours
name	init	$\gamma = 0$	$\gamma = 1$	$\gamma = 3$	
s1196	88.2%	89.8%	89.8%	89.8%	90.0%
s1238	90.0%	90.0%	90.0%	90.0%	90.0%
s1423	22.3%	88.4%	88.4%	88.4%	90.0%
s1488	63.0%	90.0%	90.0%	90.0%	90.0%
s1494	71.2%	90.0%	90.0%	90.0%	90.0%
s5378	82.5%	82.5%	82.5%	82.5%	90.0%
s9234.1	90.6%	90.6%	90.0%	90.0%	90.0%
s13207.1	82.8%	88.9%	88.9%	88.9%	90.0%
s15850.1	46.5%	87.2%	87.5%	87.5%	90.0%
s35932	74.2%	91.5%	89.5%	84.7%	90.0%
s38417	0.2%	83.1%	84.3%	82.2%	90.0%
s38584.1	85.3%	90.0%	90.0%	90.0%	90.0%

Outline

Conclusions

- Formulate the Risk Aversion Min-Period Retiming problem based on two-stage stochastic programs with fixed recourse and the conditional value-at-risk measure.
- Prove the convexity of the proposed formulation and derive an analytical formula for the subgradient of the objective function.
- Present an incremental algorithm to solve the proposed problem heuristically.
- We expect similar techniques to be applied to other statistical circuit optimizations.

Q & A

Thank you!