Introduction to Convex Optimization and its Applications

Sunghee Yun

CAE Team Samsung Semiconductor

(partially quoted from Boyd, IEEE CDC, 2002)

Outline

- introduction to convex optimization
- convex analysis and optimization
- new standard convex problem classes
- statistical digital circuit optimization using GGP
- **why** convex optimization?
- conclusions

Introdcution to Convex Optimization

Easy and Hard Problems

Least squares (LS)

$$\text{minimize} \quad \|Ax - b\|_2^2$$

$A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$ are parameters; $x \in \mathbf{R}^n$ is variable

- have complete theory (existence & uniqueness, sensitivity analysis . . .)
- several algorithms compute (global) solution reliably
- can solve dense problems with $n = 1000$ vbles, $m = 10000$ terms
- by exploiting structure (e.g., sparsity) can solve **far larger** problems

. . . LS is a (widely used) **technology**

Linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

$c,\ a_i \in \mathbf{R}^n$ are parameters; $x \in \mathbf{R}^n$ is variable

- have nearly complete theory
  (existence & uniqueness, sensitivity analysis . . .)
- several algorithms compute (global) solution reliably
- can solve dense problems with $n = 1000$ vbles, $m = 10000$ constraints
- by exploiting structure (e.g., sparsity) can solve **far larger** problems

. . . LP is a (widely used) **technology**

Quadratic program (QP)

$$\begin{array}{ll} \text{minimize} & \|Fx - g\|_2^2 \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- a combination of LS & LP
- same story . . . QP is a technology
- solution methods reliable enough to be **embedded in real-time control applications** with little or no human oversight
- basis of **model predictive control**

Convex optimization

$$\begin{array}{ll}\text{minimize} & f_0(x) \\ \text{subject to} & f_1(x) \leq 0, \ldots, f_m(x) \leq 0\end{array}$$

$x \in \mathbf{R}^n$ is optimization variable; $f_i : \mathbf{R}^n \to \mathbf{R}$ are **convex**:

$$f_i(\lambda x + (1-\lambda)y) \leq \lambda f_i(x) + (1-\lambda) f_i(y)$$

for all x, y, $0 \leq \lambda \leq 1$

- includes LS, LP, QP, and **many others**
- like LS, LP, and QP, convex problems are **fundamentally tractable**

The good news

convex optimization problems include handful classes of classical optimization problems, $e.g.$, LS, LP, QP, . . .

The bad news

- LS, LP, and QP are **exceptions**
- most optimization problems, even some very simple looking ones, are **intractable**, $e.g.$, circuit yield maximization problem with contraints on (dynamic & leakage) power and area
- example: `NeoCircuit` uses combination of randomization & simulated annealing, but cannot find global optimal solution, and more importantly takes enormous time; not practical for full-chip optimization

Even worse news

very difficult and **very easy** problems can look **quite similar**
(to the untrained eye)

Example: Polynomial minimization

$$\text{minimize} \quad p(x)$$

p is polynomial of degree d; $x \in \mathbf{R}^n$ is variable

- except for special cases (e.g., $d = 2$) this is a **very difficult problem**
- even sparse problems with size $n = 20$, $d = 10$ are essentially intractable
- all algorithms known to solve this problem require effort exponential in n

Example: Robust LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{Prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m \end{array}$$

coefficient vectors a_i IID, $\mathcal{N}(\overline{a}_i, \Sigma_i)$; η is required reliability

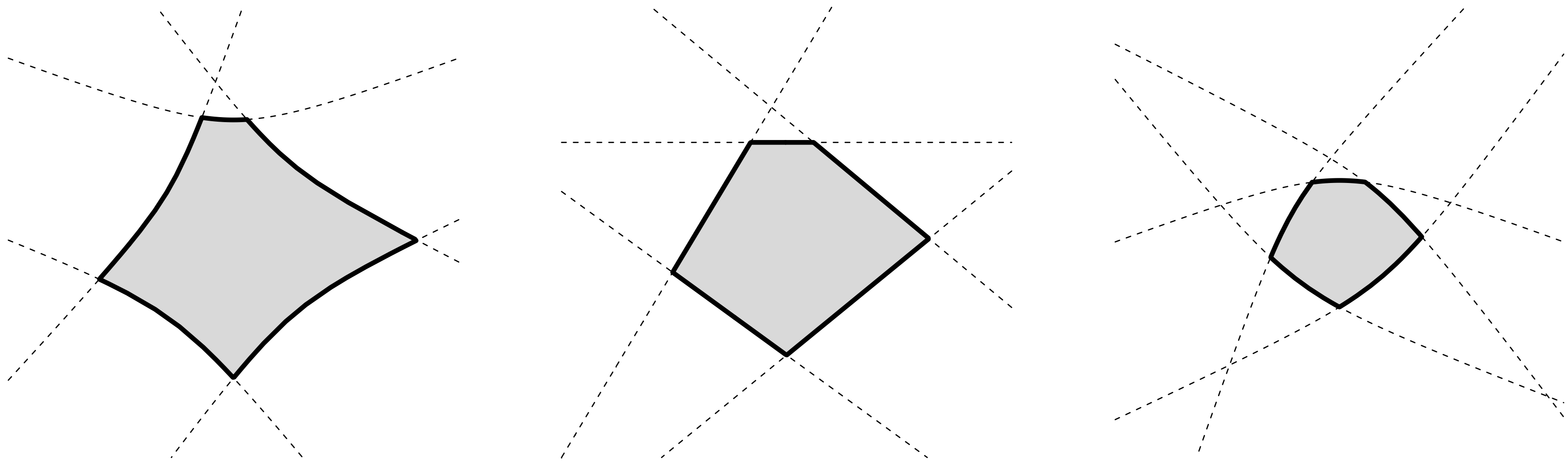
- for fixed x, $a_i^T x$ is $\mathcal{N}(\overline{a}_i^T x, x^T \Sigma_i x)$
- so for $\eta = 50\%$, robust LP reduces to LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \overline{a}_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

  and so is easily solved
- what about other values of η, *e.g.*, $\eta = 10\%$? $\eta = 90\%$?

Hint

$$\{x \mid \mathbf{Prob}(a_i^T x \le b_i) \ge \eta,\ i = 1, \ldots, m\}$$



$\eta = 10\%$ $\eta = 50\%$ $\eta = 90\%$

That's right

robust LP with reliability $\eta = 90\%$ is convex, and **very easily solved**

robust LP with reliability $\eta = 10\%$ is not convex, and **extremely difficult**

moral: **very difficult** and **very easy** problems can look **quite similar**
(to the untrained eye)

What makes a problem easy or hard?

classical view:

- **linear** is easy
- **nonlinear** is hard(er)

What makes a problem easy or hard?

emerging (and correct) view:

. . . the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

— R. Rockafellar, SIAM Review 1993

Convex Analysis and Optimization

Convex analysis & optimization

nice properties of convex optimization problems known since 1960s

- local solutions are global
- duality theory, optimality conditions

convex analysis well developed by 1970s *Rockafellar*

- separating & supporting hyperplanes
- subgradient calculus

What's new (since 1990 or so)

- primal-dual interior-point (IP) methods
  extremely efficient, handle nonlinear large scale problems, polynomial-time complexity results, software implementations

- new standard problem classes
  generalizations of LP, with theory, algorithms, software

- extension to generalized inequalities
  semidefinite, cone programming

- emerging convex optimization programming in circuit area
  generalized geometric programming, with theory, algorithms, software, dual

. . . convex optimization is becoming a **technology**

Applications and uses

- lots of applications
  control, combinatorial optimization, signal processing, circuit design, communications, . . .

- robust optimization
  robust versions of LP, LS, other problems

- relaxations and randomization
  provide bounds, heuristics for solving hard problems

Recent history

- 1984–97: interior-point methods for LP
  - 1984: Karmarkar's interior-point LP method
- 1988: Nesterov & Nemirovsky's self-concordance analysis
- 1989–: LMIs and semidefinite programming in control
- 1990–: semidefinite programming in combinatorial optimization
  Alizadeh, Goemans, Williamson, Lovasz & Schrijver, Parrilo, . . .
- 1994: interior-point methods for nonlinear convex problems
  Nesterov & Nemirovsky, Overton, Todd, Ye, Sturm, . . .
- 1997–: robust optimization *Ben Tal, Nemirovsky, El Ghaoui, . . .*

New Standard Convex Problem Classes

Some new standard convex problem classes

- second-order cone program (SOCP)
- geometric program (GP) (and entropy problems)
- semidefinite program (SDP)
- generalized geometric program (GGP) (and circuit optimization problems)

for these new problem classes we have

- complete duality theory, similar to LP
- good algorithms, and robust, reliable software
- wide variety of new applications

Second-order cone program

second-order cone program (SOCP) has form

$$\begin{array}{ll} \text{minimize} & c_0^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m \end{array}$$

with variable $x \in \mathbf{R}^n$

- includes LP and QP as special cases
- nondifferentiable when $A_i x + b_i = 0$
- new IP methods can solve (almost) as fast as LPs

Example: robust linear program

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{Prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m \end{array}$$

where $a_i \sim \mathcal{N}(\bar{a}_i, \Sigma_i)$

equivalent to

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \bar{a}_i^T x + \Phi^{-1}(\eta) \|\Sigma_i^{1/2} x\|_2 \leq b_i, \quad i = 1, \ldots, m \end{array}$$

where Φ is (unit) normal CDF

robust LP is an SOCP for $\eta \geq 0.5$ ($\Phi(\eta) \geq 0$)

Entropy problems

unnormalized negative entropy is convex function

$$-\,\mathbf{entr}(x) = \sum_{i=1}^{n} x_i \log(x_i/\mathbf{1}^T x)$$

defined for $x_i \geq 0$, $\mathbf{1}^T x > 0$

entropy problem:

$$\begin{array}{ll} \text{minimize} & -\,\mathbf{entr}(A_0 x + b_0) \\ \text{subject to} & -\,\mathbf{entr}(A_i x + b_i) \leq 0, \quad i = 1, \ldots, m \end{array}$$

$A_i \in \mathbf{R}^{m_i \times n}$, $b_i \in \mathbf{R}^{m_i}$

Geometric program (GP)

log-sum-exp function:

$$\mathbf{lse}(x) = \log \left(e^{x_1} + \cdots + e^{x_n}\right)$$

. . . a smooth **convex** approximation of the max function

geometric program:

$$\begin{array}{ll} \text{minimize} & \mathbf{lse}(A_0 x + b_0) \\ \text{subject to} & \mathbf{lse}(A_i x + b_i) \leq 0, \quad i = 1, \ldots, m \end{array}$$

$A_i \in \mathbf{R}^{m_i \times n}$, $b_i \in \mathbf{R}^{m_i}$; variable $x \in \mathbf{R}^n$

Monomial functions

$x = (x_1, \ldots, x_n)$: vector of positive variables

function $f : \mathbf{R}^{+n} \rightarrow \mathbf{R}^{+}$ of form

$$f(x) = cx_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$$

with $c > 0$, $\alpha_i \in \mathbf{R}$, is called **monomial**

Posynomial functions

$x = (x_1, \ldots, x_n)$: vector of positive variables

function $f : \mathbf{R}^{+n} \to \mathbf{R}^+$ of form

$$f(x) = \sum_{k=1}^{t} c_k x_1^{\alpha_{1k}} x_2^{\alpha_{2k}} \cdots x_n^{\alpha_{nk}}$$

with $c_k > 0$, $\alpha_{ik} \in \mathbf{R}$, is called **posynomial**

like polynomial, but

- coefficients must be positive
- exponents can be fractional or negative

Posynomial form GP

posynomial form GP:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, \ldots, m \end{array}$$

f_i are posynomial; $x_i > 0$ are variables

to convert to (convex form) GP, express as

$$\begin{array}{ll} \text{minimize} & \log f_0(e^y) \\ \text{subject to} & \log f_i(e^y) \leq 0, \quad i = 1, \ldots, m \end{array}$$

objective and constraints have form $\mathbf{lse}(A_i y + b_i)$

Solving GPs (and entropy problems)

- GP and entropy problems are **duals** (if we solve one, we solve the other)
- new IP methods can solve large scale GPs (and entropy problems) almost as fast as LPs
- applications in many areas:
  - information theory, statistics
  - communications, wireless power control
  - digital and analog circuit design

Generalized geometric program (GGP)

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, \ldots, m \\ & g_i(x) = 1, \quad i = 1, \ldots, p \end{array}$$

f_i are **generalized posynomials**, g_i are monomials

- using tricks, can convert GGP to GP, then solve efficiently
- **conversion tricks can be automated**
  - parser scans problem description, forms GP
  - GP solver solves GP
  - solution transformed back (dummy variables eliminated)

What have we seen?

- convex optimization includes several classical optimization problems, and many others

- takes trained eye to find problems which fits into convex optimization

- very efficient (and fast) algorithms and software developed for (large) convex optimization problems

Statistical Digital Circuit Optimization

Statistical parameter variations in circuits

- statistical variations in process
  - random defects: random particles while chemical mechanical polishing (CMP), chemical vapor deposition (CVD), physical vapor deposition (PVD), *etc.*
  - systematic defects: optical proximity correction (OPC), *etc.*

- statistical variations in environment: supply voltage, temperature, *etc.*

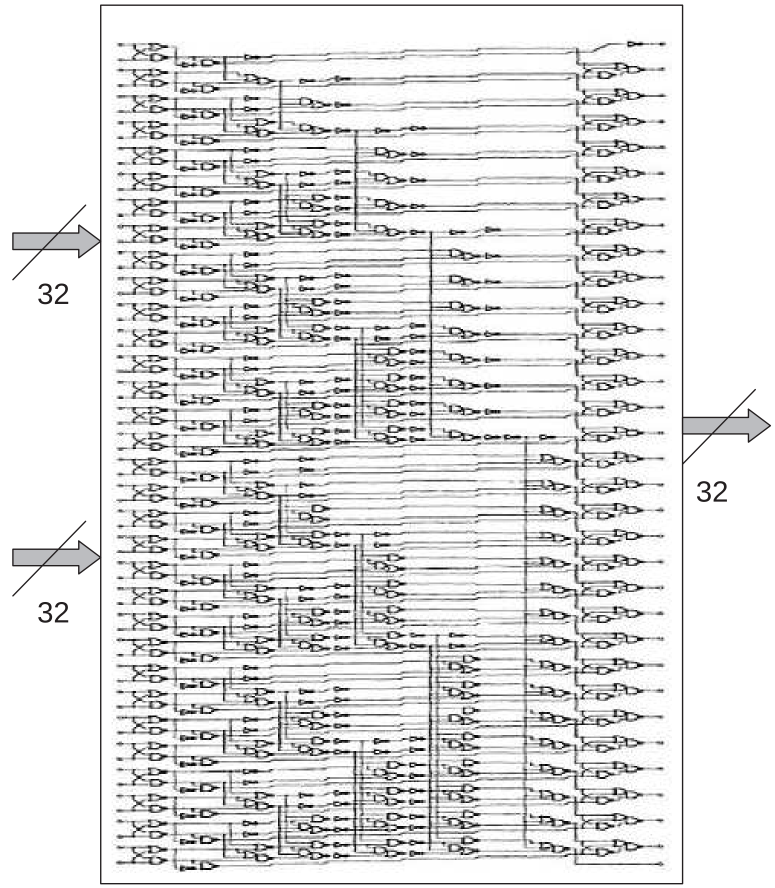
$\Rightarrow$ induces statistical variations in (physical) parameters, *e.g.*, effective length, width, oxide thickness, zero biased threshold voltage, mobility, *etc.*

Statistical parameter variations in circuits

- statistical variation significantly affects performance in deep submicron (DSM) regime
- statistical variation is very complex and extremely hard; modeling still open
- some efficient statistical circuit analysis methods
- merely start exploring statistical design methods; design for manufacturability (DFM), design for yield (DFY), design for testability (DFT), *etc.*

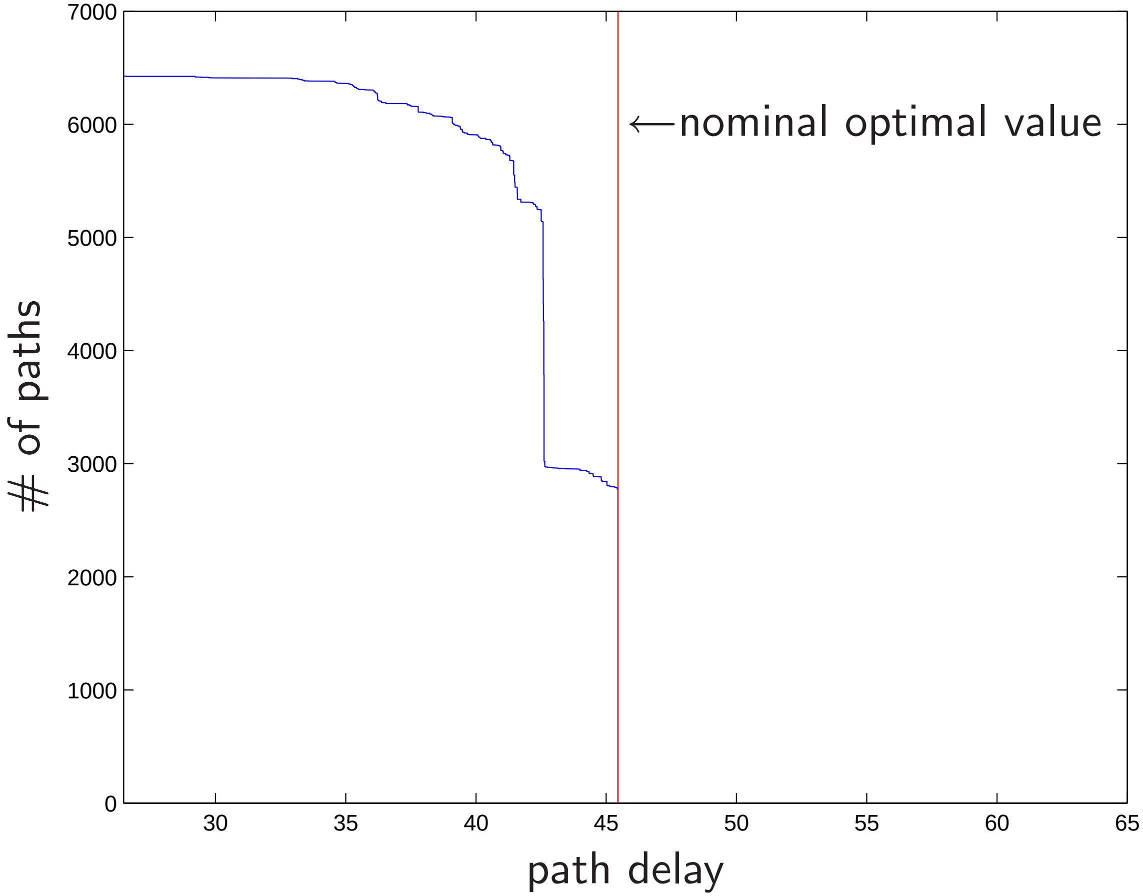
Ladner-Fisher 32-bit adder example

- minimize maximum delay with constraints
- simplified RC delay model
- Pelgrom variation model (15% σ/μ for min size devices)
- design variables:
  device widths for 451 gates . . .



Schematic of Ladner-Fisher 32-bit adder

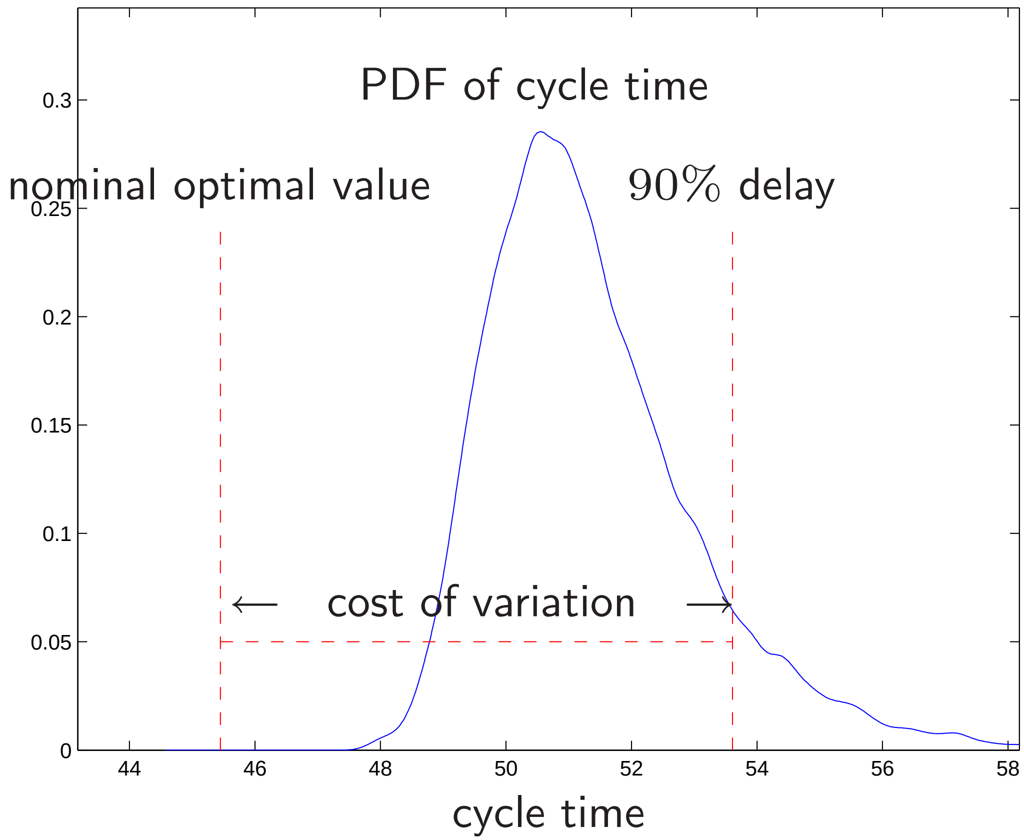
Optimization results (nominal: no uncertainty)



. . . around 2800 of 6400 total paths are critical

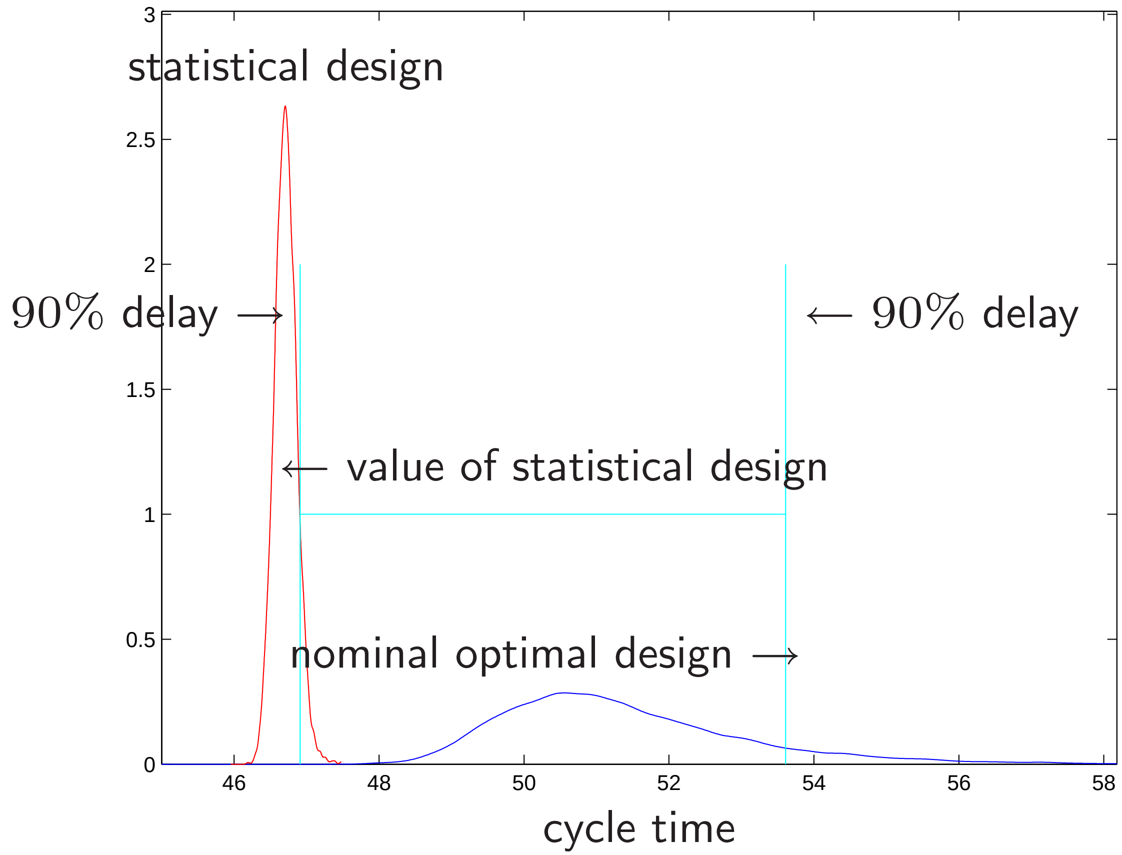
Cost of statistical variation

Monte Carlo SSTA analysis of nominal optimal design



Statistically robust design via new method

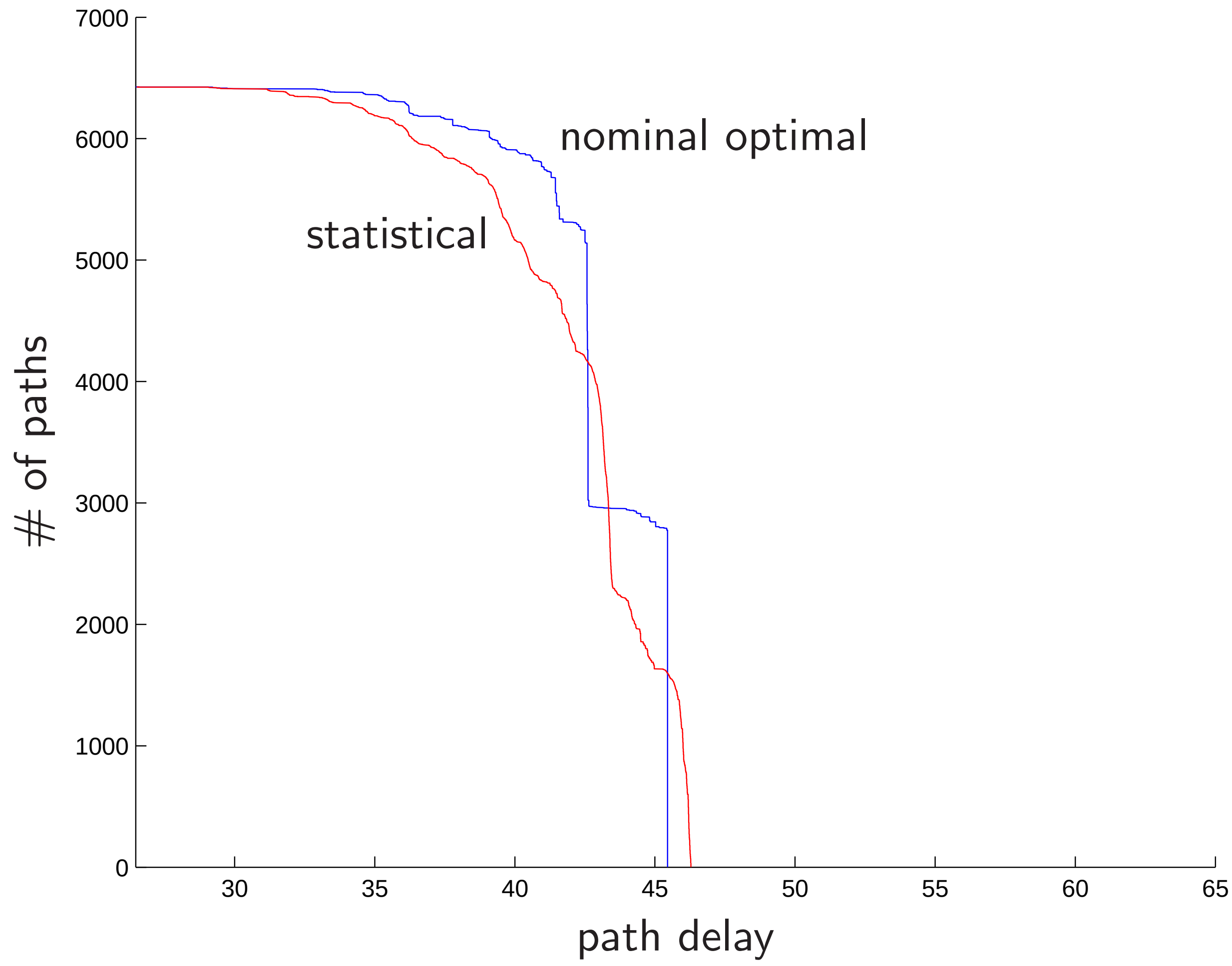
same circuit, uncertainty model, and constraints



Statistically robust design via new method

	Nominal delay	90% delay
Nominal design	**45.4**	53.6
Statistical design	46.3	**46.9**

Nominal optimal versus statistical design



In the second half of this talk

- introduce what generalized geometric program (GGP) is in detail

- show how we can
  - model gate delay very accurately using generalized posynomials
  - exactly cast nominal delay minimization problem into GGP
  - use heuristic for statistical circuit design problem using GGP

  when statistical static timing analysis (SSTA) is applied

Why convex optimization?

Trade-offs in optimization

- general trade-off between **generality** and **effectiveness**
- generality
  - number of problems that can be handled
  - accuracy of formulation
  - ease of formulation
- effectiveness
  - speed of solution, scale of problems that can be handled
  - global vs. local solutions
  - reliability, no baby-sitting, no starting point

Example: least-squares vs. simulated annealing

least-squares

- large problems reliably (globally) solved quickly
- no initial point, no algorithm parameter tuning
- solves very restricted problem form
- with tricks and extensions, basis of vast number of methods that work (control, filtering, regression, . . .)

simulated annealing

- can be applied to any problem (more or less)
- slow, needs tuning, babysitting; not global in practice
- method of choice for some problems you can't handle any other way

Where convex optimization fits in

somewhere in between, closer to least-squares . . .

- like least-squares, large problems can be solved reliably (globally), no starting point, tuning, . . .

- solves a class of problems broader than least-squares, less general than simulated annealing

- **formulation takes effort, but is fun and has high payoff**

Conclusions

- convex optimization includes several classical optimization problems, and many others
- takes trained eye to find problems which fits into convex optimization
- very efficient (and fast) algorithms and software developed for (large) convex optimization problems
- can cast circuit optimization problems into convex optimization problems, $e.g.$, GGP
- simple heuristic works very well for statistical digital circuit optimization problem