COMP 9602: Convex Optimization

Subgradient Methods

Dr. C. Wu

Department of Computer Science
The University of Hong Kong

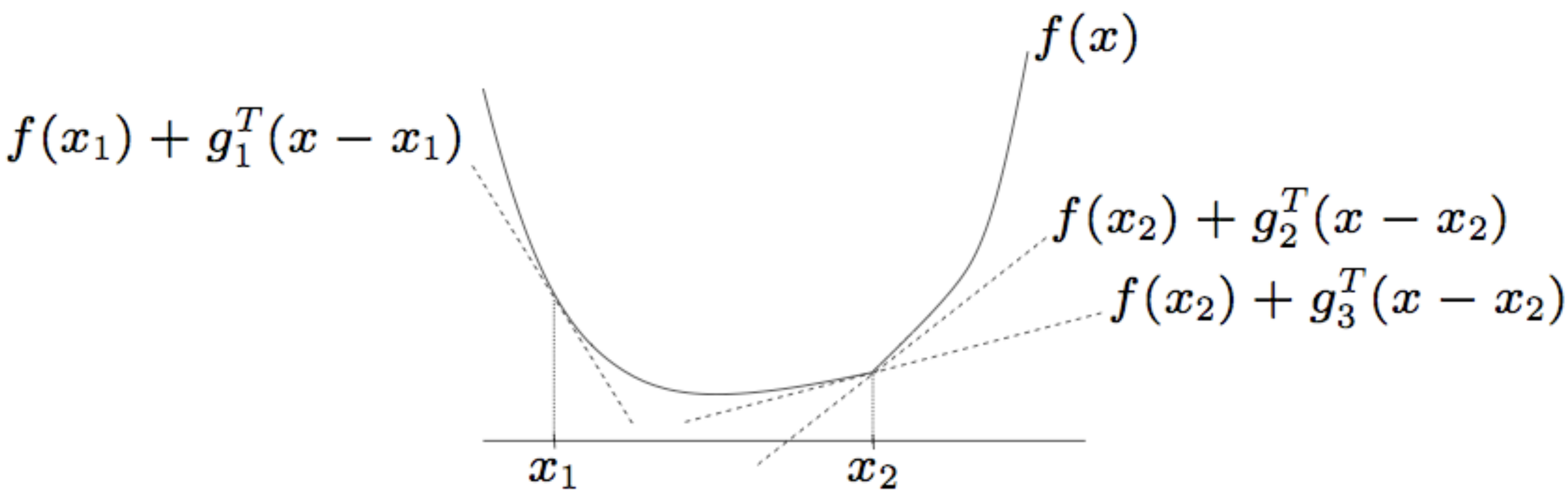
Roadmap

Theory	convex set convex function standard forms of optimization problems, quasi-convex optimization linear program, integer linear program quadratic program geometric program semidefinite program vector optimization duality
Algorithm	unconstrained optimization equality constrained optimization interior-point method subgradient method localization methods decomposition methods and more

Subgradient

g is a **subgradient** of f (not necessarily convex) at x if

$$f(y) \geq f(x) + g^T(y - x) \quad \text{for all } y$$



g_2, g_3 are subgradients at x_2; g_1 is a subgradient at x_1

(if $f(y) \leq f(x) + g^T(y - x)$ for all y, then g is a **supergradient**)

Subgradient (cont'd)

g is a **subgradient** of f (not necessarily convex) at x if

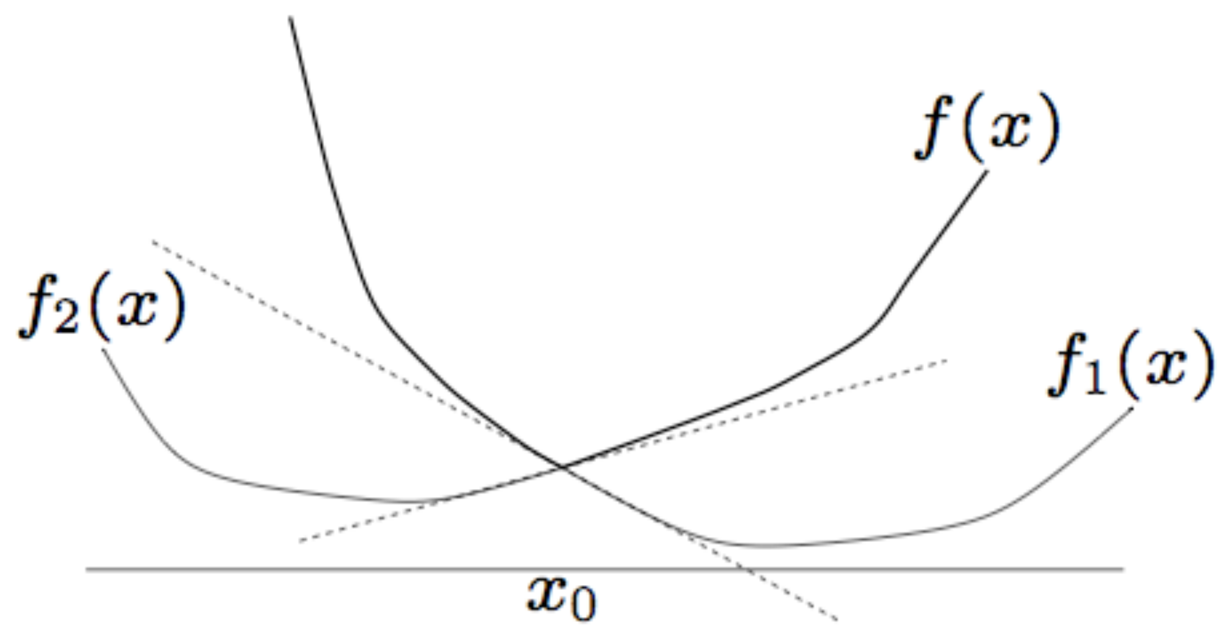
$$f(y) \geq f(x) + g^T(y - x) \quad \text{for all } y$$

- if f is convex and differentiable, $\nabla f(x)$ is the unique subgradient of f at x
- for non-differentiable functions, lots of choices for g are possible
- subgradient is useful in
  - algorithms for nondifferentiable convex optimization
  - convex analysis, *e.g.*, optimality conditions, duality for nondifferentiable problems

Subgradient (cont'd)

- Example

$f = \max\{f_1, f_2\}$, with f_1, f_2 convex and differentiable



- $f_1(x_0) > f_2(x_0)$: unique subgradient $g = \nabla f_1(x_0)$
- $f_2(x_0) > f_1(x_0)$: unique subgradient $g = \nabla f_2(x_0)$
- $f_1(x_0) = f_2(x_0)$: subgradients form a line segment $[\nabla f_1(x_0), \nabla f_2(x_0)]$

Subdifferential

- set of all subgradients of f at x is called the **subdifferential** of f at x, denoted $\partial f(x)$
- $\partial f(x)$ is a closed convex set (can be empty)

if f is convex,

- $\partial f(x)$ is nonempty, for $x \in$ **int dom** f
- $\partial f(x) = \{\nabla f(x)\}$, if f is differentiable at x
- if $\partial f(x) = \{g\}$, then f is differentiable at x and $g = \nabla f(x)$

Optimality condition for unconstrained problem

recall for f convex, differentiable,

$$f(x^\star) = \inf_x f(x) \iff 0 = \nabla f(x^\star)$$

generalization to nondifferentiable convex f:

$$f(x^\star) = \inf_x f(x) \iff 0 \in \partial f(x^\star)$$

Optimality condition for constrained problem

minimize $\quad f_0(x)$
subject to $\quad f_i(x) \leq 0,\ i = 1, \ldots, m$

we assume

- f_i convex, defined on $\mathbf{R}^n$ (hence subdifferentiable)
- strict feasibility (Slater's condition)

$x^\star$ is primal optimal ($\lambda^\star$ is dual optimal) iff

$$f_i(x^\star) \leq 0, \quad \lambda_i^\star \geq 0$$

$$\lambda_i^\star f_i(x^\star) = 0$$

$$0 \in \partial f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star \partial f_i(x^\star)$$

generalizes KKT for nondifferentiable f_i

Subgradient method

- A simple algorithm to minimize non-differentiable convex functions
- Similar to gradient method, but
  - step length not chosen using line search
  - not a descent method: function value can increase
- As compared to interior point and Newton's method
  - can be slower
  - can be applied to a much wider variety of problems
  - for large-scale problem: memory requirement much smaller
  - for distributed solution design by combining with primal or dual decomposition

Subgradient method for unconstrained optimization

Give a starting point $x^{(1)} \in \mathbf{dom} f$

Repeat

1. Find a subgradient $g^{(k)}$ of f at $x^{(k)}$
2. Choose a step size α_k
3. Update $x^{(k+1)} = x^{(k)} - \alpha_k g^{(k)}$

Until stopping criteria is satisfied

- $x^{(k)}$ is the kth iterate
- $g^{(k)}$ is **any** subgradient of f at $x^{(k)}$
- $\alpha_k > 0$ is the kth step size

not a descent method, so we keep track of best point so far

$$f_{\text{best}}^{(k)} = \min_{i=1,\ldots,k} f(x^{(i)})$$

Step size

- Step size rules (step sizes are fixed before algorithm execution)
  - *constant step size:* $\alpha_k = \alpha$ (constant)
  - *constant step length:* $\alpha_k = \gamma / \|g^{(k)}\|_2$ (so $\|x^{(k+1)} - x^{(k)}\|_2 = \gamma$)
  - *square summable but not summable:* step sizes satisfy

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

e.g. $\alpha_k = a/(b+k)$, where $a > 0$ and $b \geq 0$

  - *nonsummable diminishing:* step sizes satisfy

$$\lim_{k \to \infty} \alpha_k = 0, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

e.g. $\alpha_k = a/\sqrt{k}$, where $a > 0$

Convergence

- assumptions
  - $f^\star = \inf_x f(x) > -\infty$, with $f(x^\star) = f^\star$
  - $\|g\|_2 \leq G$ for all $g \in \partial f$
  - $R \geq \|x^{(1)} - x^\star\|_2$

$$\boxed{f_{\text{best}}^{(k)} - f^\star \leq \frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2 \sum_{i=1}^k \alpha_i}}$$

- convergence results: define $\bar{f} = \lim_{k\to\infty} f_{\text{best}}^{(k)}$
  - *constant step size:* $\bar{f} - f^\star \leq G^2\alpha/2$, *i.e.*, **converges to $G^2\alpha/2$-suboptimal** (converges to $f^\star$ if f differentiable, α small enough)
  - *constant step length:* $\bar{f} - f^\star \leq G\gamma/2$, *i.e.*, **converges to $G\gamma/2$-suboptimal**
  - *square summable but not summable* *step size rule:* $\bar{f} = f^\star$, *i.e.*, **converges** to optimality
  - *none summable diminishing* *step size rule:* $\bar{f} = f^\star$, *i.e.*, **converges** to optimality

Stopping criteria

- Stopping criteria

$$f_{\text{best}}^{(k)} - f^\star \le \frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2\sum_{i=1}^k \alpha_i} \quad \le \epsilon$$

- terminating when $\frac{R^2 + G^2 \sum_{i=1}^k \alpha_i^2}{2\sum_{i=1}^k \alpha_i} \le \epsilon$ can be very very slow

Example

- piece-wise linear minimization

minimize $\quad f(x) = \max_{i=1,\ldots,m}(a_i^T x + b_i)$

to find a subgradient of f: find index j for which $a_j^T x + b_j = \max\limits_{i=1,\ldots,m}(a_i^T x + b_i)$

and take $g = a_j$

subgradient method: $x^{(k+1)} = x^{(k)} - \alpha_k a_j$

Example

minimize $\quad f(x) = \max_{i=1,\ldots,m}(a_i^T x + b_i)$

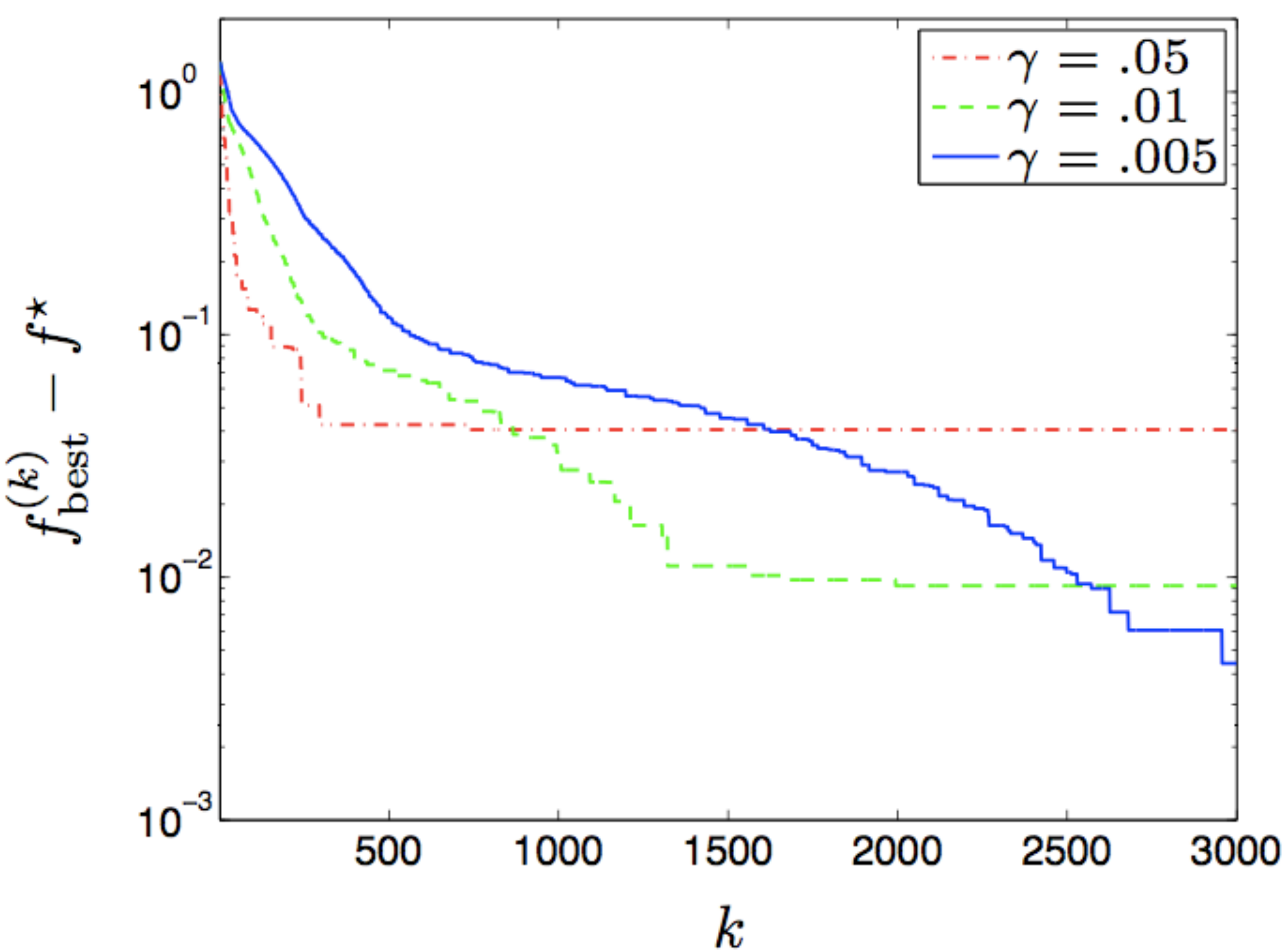
problem instance with $n = 20$ variables, $m = 100$ terms

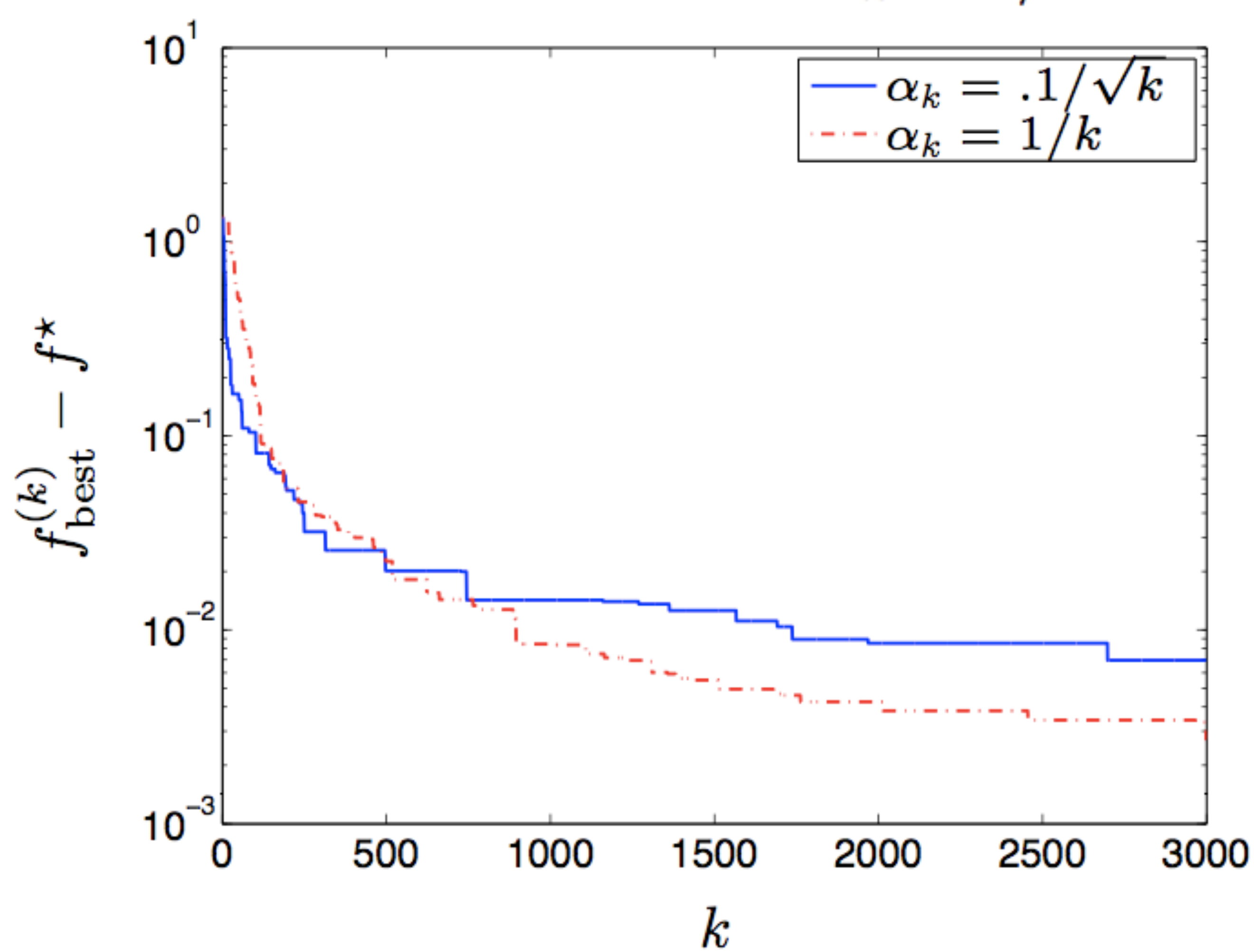
constant step length $\gamma = 0.05, 0.01, 0.005$

diminishing step rule $\alpha_k = 0.1/\sqrt{k}$

square summable step size rule

$\alpha_k = 1/k$





Subgradient method for constrained problems

constrained optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{C}, \end{array}$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $\mathcal{C} \subseteq \mathbf{R}^n$ are convex

Projected subgradient method

- projected subgradient method for primal problem
- projected subgradient method for dual problem

Subgradient method for constrained optimization

Projected subgradient method for primal problem

solves constrained optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{C}, \end{array}$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $\mathcal{C} \subseteq \mathbf{R}^n$ are convex

projected subgradient method is given by

$$x^{(k+1)} = P(x^{(k)} - \alpha_k g^{(k)}),$$

P is (Euclidean) projection on $\mathcal{C}$, and $g^{(k)} \in \partial f(x^{(k)})$

Projected subgradient method for primal problem

Give a starting point $x^{(1)} \in \mathbf{dom} f$

Repeat

1. Find a subgradient $g^{(k)}$ of f at $x^{(k)}$
2. Choose a step size α_k
3. Update $x^{(k+1)} = P(x^{(k)} - \alpha_k g^{(k)})$, where

P is (Euclidean) projection on C

Until stopping criteria is satisfied

Projection

- Projection: $s = argmin_{s \in C} \| x - s \|_2$
- Example: linear equality constrained problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & Ax = b \end{array}$$

projection of z onto $\{x \mid Ax = b\}$ is

$$\begin{aligned} P(z) &= z - A^T(AA^T)^{-1}(Az - b) \\ &= (I - A^T(AA^T)^{-1}A)z + A^T(AA^T)^{-1}b) \end{aligned}$$

projected subgradient update is

$$\begin{aligned} x^{(k+1)} &= P(x^{(k)} - \alpha_k g^{(k)}) \\ &= x^{(k)} - \alpha_k(I - A^T(AA^T)^{-1}A)g^{(k)} \end{aligned}$$

Convergence

same convergence results:

- for constant step size, converges to neighborhood of optimal (for f differentiable and α small enough, converges)
- for diminishing nonsummable step sizes, converges

key idea: projection does not increase distance to $x^\star$

Example

- Linear equality constrained problem

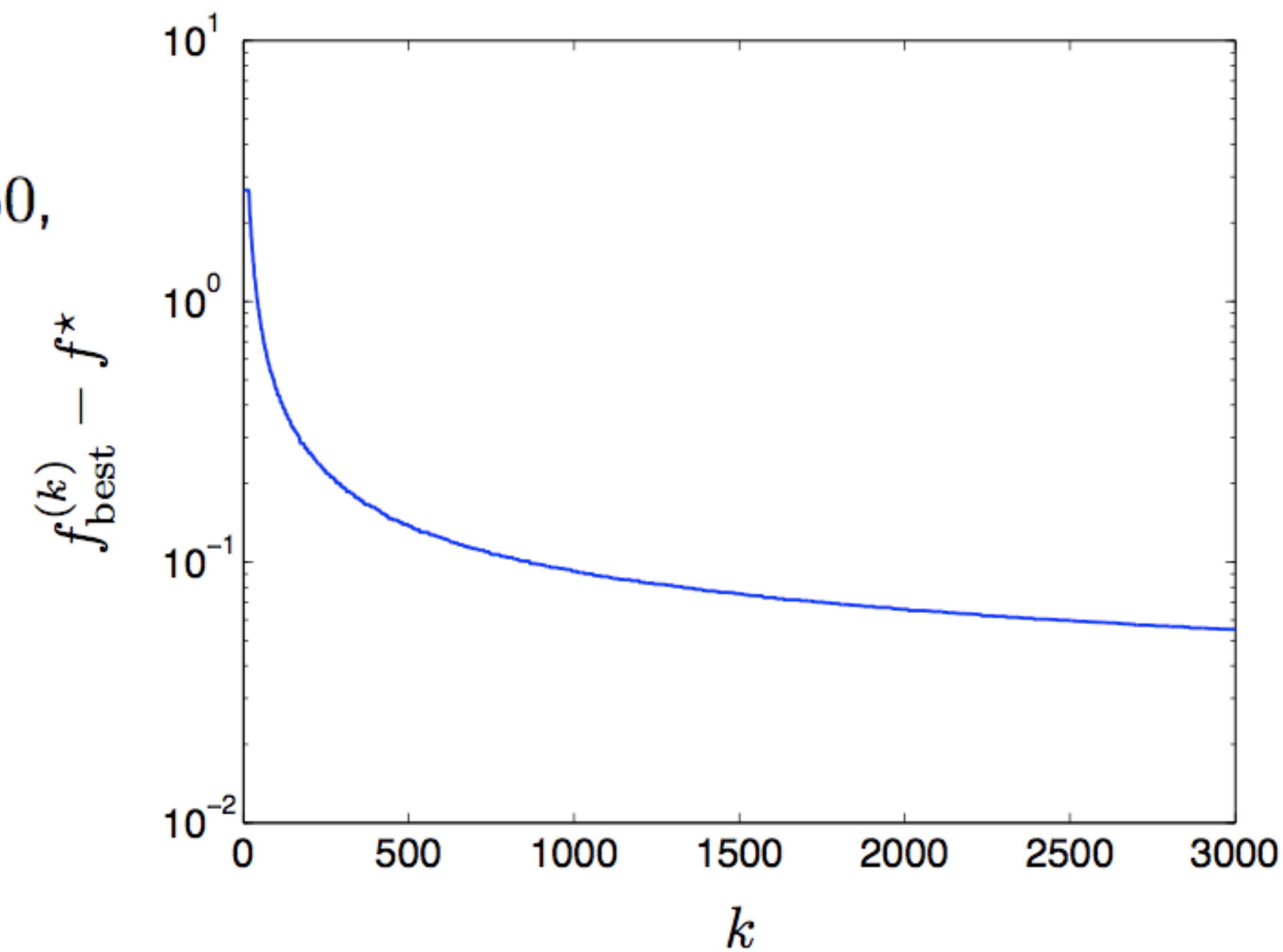
$$\begin{array}{ll} \text{minimize} & \|x\|_1 \\ \text{subject to} & Ax = b \end{array}$$

subgradient of objective is $g = \mathbf{sign}(x)$

projected subgradient update is $x^{(k+1)} = x^{(k)} - \alpha_k (I - A^T(AA^T)^{-1}A)\,\mathbf{sign}(x^{(k)})$

problem instance with $n = 1000$, $m = 50$,

step size $\alpha_k = 0.1/k$



Projected subgradient method for dual problem

(convex) primal:(Slater's condition holds)

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

solve dual problem

$$\begin{array}{ll} \text{maximize} & g(\lambda) \\ \text{subject to} & \lambda \succeq 0 \end{array}$$

via projected subgradient method:

$$\lambda^{(k+1)} = (\lambda^{(k)} - \alpha_k h^{(k)})_+ \qquad h^{(k)} \in \partial(-g)(\lambda^{(k)})$$

Projected subgradient method for dual problem

$$g(\lambda) = \inf_x (f_0(x) + \lambda_1 f_1(x) + \ldots + \lambda_m f_m(x))$$

$$-g(\lambda) = \sup_x (-f_0(x) - \lambda_1 f_1(x) - \ldots - \lambda_m f_m(x))$$

denote, for $\lambda \succeq 0$,

$$x^*(\lambda) = \operatorname*{argmin}_z \left(f_0(z) + \lambda_1 f_1(z) + \cdots + \lambda_m f_m(z)\right)$$

so $-g(\lambda) = -f_0(x^*(\lambda)) - \lambda_1 f_1(x^*(\lambda)) - \cdots - \lambda_m f_m(x^*(\lambda))$

a subgradient of $-g$ at λ is given by $h_i = -f_i(x^*(\lambda))$

projected subgradient method for dual:

$$x^{(k)} = x^*(\lambda^{(k)}), \qquad \lambda_i^{(k+1)} = \left(\lambda_i^{(k)} + \alpha_k f_i(x^{(k)})\right)_+$$

Projected subgradient method for dual problem

Give a starting point $\lambda^{(1)} \succeq 0$

Repeat

1. $\mathrm{x}^{(k)} = \mathrm{argmin}_z (f_0(z) + \lambda_1^{(k)} f_1(z) + \cdots + \lambda_m^{(k)} f_m(z))$

2. Choose a step size α_k

3. Update $\lambda_i^{(k+1)} = \left(\lambda_i^{(k)} + \alpha_k f_i(x^{(k)})\right)_+$

Until stopping criteria is satisfied

Projected subgradient method for dual problem

interpretation:

- λ_i is price for 'resource' $f_i(x)$
- price update $\lambda_i^{(k+1)} = \left(\lambda_i^{(k)} + \alpha_k f_i(x^{(k)})\right)_+$
  - increase price λ_i if resource i is over-utilized ($i.e.$, $f_i(x) > 0$)
  - decrease price λ_i if resource i is under-utilized ($i.e.$, $f_i(x) < 0$)
  - but never let prices get negative

convergence:

- primal iterates $x^{(k)}$ are not feasible, but become feasible in limit (sometimes can find feasible, suboptimal $\tilde{x}^{(k)}$ from $x^{(k)}$)
- dual function values $g(\lambda^{(k)})$ converge to $f^\star = f_0(x^\star)$

Example

minimize strictly convex quadratic ($P \succ 0$) over unit box:

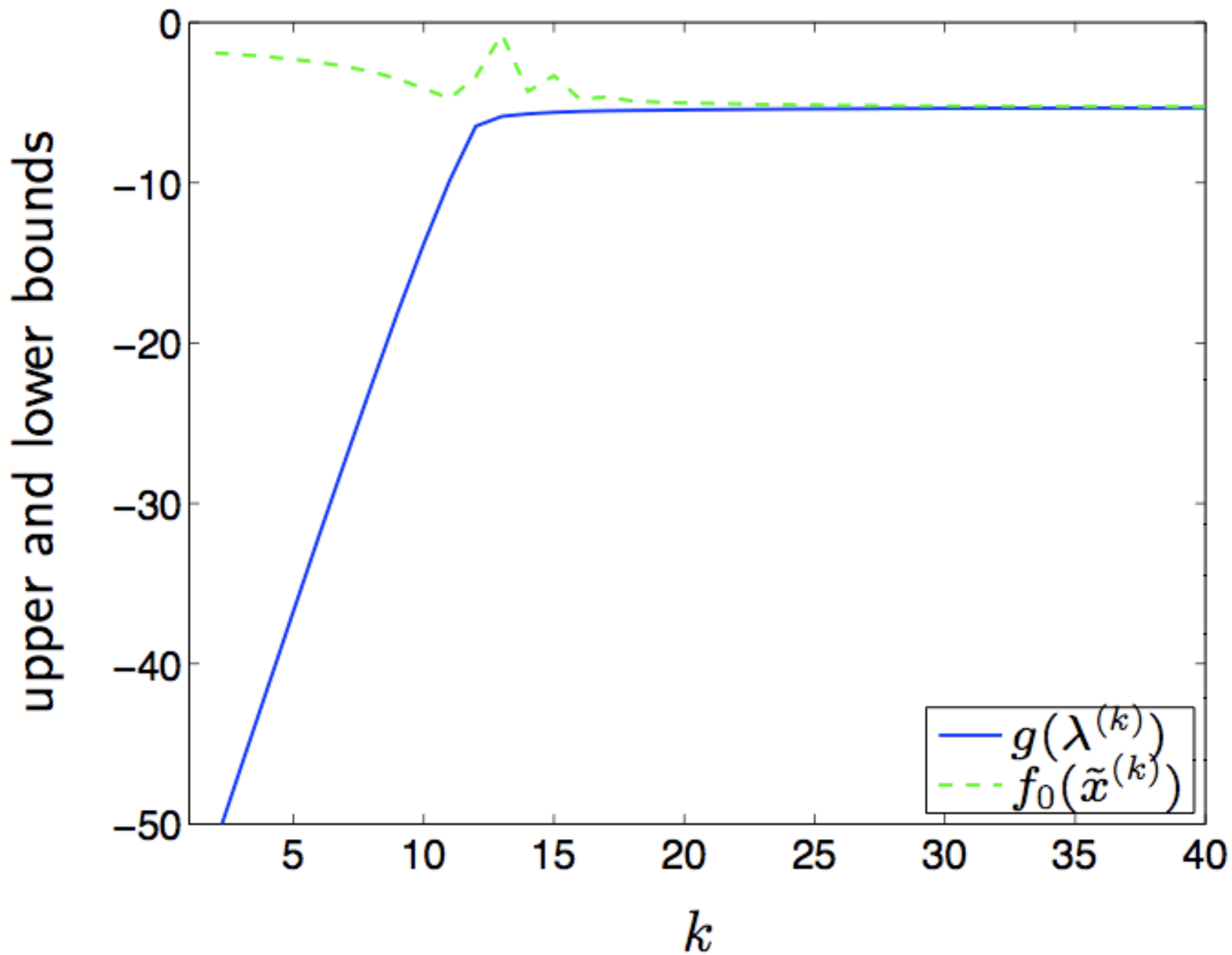
$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x - q^T x \\ \text{subject to} & x_i^2 \le 1, \quad i = 1, \ldots, n \end{array}$$

- $L(x, \lambda) = (1/2)x^T(P + \mathbf{diag}(2\lambda))x - q^T x - \mathbf{1}^T \lambda$
- $x^*(\lambda) = (P + \mathbf{diag}(2\lambda))^{-1} q$
- projected subgradient for dual:

$$x^{(k)} = (P + \mathbf{diag}(2\lambda^{(k)}))^{-1} q, \quad \lambda_i^{(k+1)} = \left(\lambda_i^{(k)} + \alpha_k((x_i^{(k)})^2 - 1)\right)_+$$

Example

problem instance with $n = 50$, fixed step size $\alpha = 0.1$, $f^\star \approx -5.3$;
$\tilde{x}^{(k)}$ is a nearby feasible point for $x^{(k)}$



Subgradient method for constrained optimization

constrained optimization problem

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

where $f_i : \mathbf{R}^n \to \mathbf{R}$ are convex

same update $x^{(k+1)} = x^{(k)} - \alpha_k g^{(k)}$, but we have

$$g^{(k)} \in \begin{cases} \partial f_0(x) & f_i(x) \leq 0, \quad i = 1, \ldots, m, \\ \partial f_j(x) & f_j(x) > 0 \end{cases}$$

- if the current point is feasible, use the subgradient of objective function
- otherwise, use the subgradient of any violated constraint

Convergence

assumptions:

- there exists an optimal $x^\star$; Slater's condition holds
- $\|g^{(k)}\|_2 \leq G$; $\|x^{(1)} - x^\star\|_2 \leq R$

typical result: for $\alpha_k > 0$, $\alpha_k \to 0$, $\sum_{i=1}^\infty \alpha_i = \infty$, we have $f_{\mathrm{best}}^{(k)} \to f^\star$

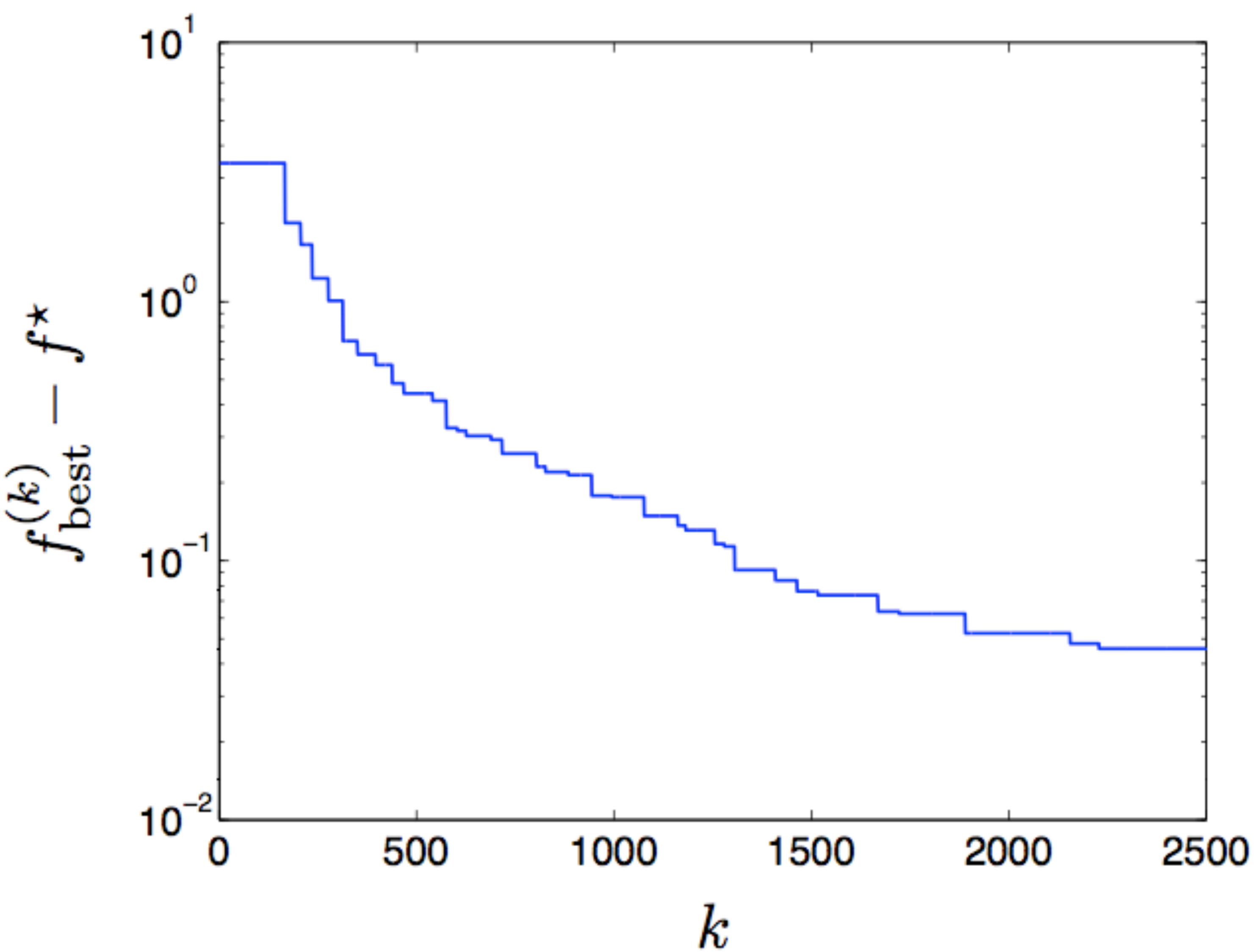
define $f_{\mathrm{best}}^{(k)} = \min\{f_0(x^{(i)}) \mid x^{(i)} \text{ feasible}, \ i = 1, \ldots, k\}$

Example

- Inequality form LP

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

LP with $n = 20$ variables, $m = 200$ inequalities, $f^\star \approx -3.4$;
$\alpha_k = 1/k$ for optimality step, Polyak's step size for feasibility step



Polyak's step size:

$$\alpha_k = \frac{f(x^{(k)}) - f^\star}{\|g^{(k)}\|_2^2}$$

can also use with estimated f*

- Reference
  - Subgradient method:
    subgradients_notes.pdf (reference 5 on Moodle)
    subgrad_method_notes.pdf (reference 6 on Moodle)
    N.Z. Shor, Minimization Methods for Non-differentiable Functions, Springer-Verlag, 1985
    Chapter 7.5, Dimitri P. Bertsekas, Nonlinear Programming (3rd edition), Athena Scientific, 2016

- Acknowledgement
  - Some materials are extracted from the slides created by Prof. Stephen Boyd for EE364b at Standard University