Subgradient Methods

Wing-Kin (Ken) Ma

Department of Electronic Engineering,
The Chinese University Hong Kong, Hong Kong

Lesson 13, ELEG5481

Subgradient Methods

- Subgradient methods are a class of simple methods for solving convex problems, including those with nondifferentiable functions.

- developed in the Soviet Union in the 60's and 70's by Shor and others.

- can be slow (perhaps very slow) in convergence.

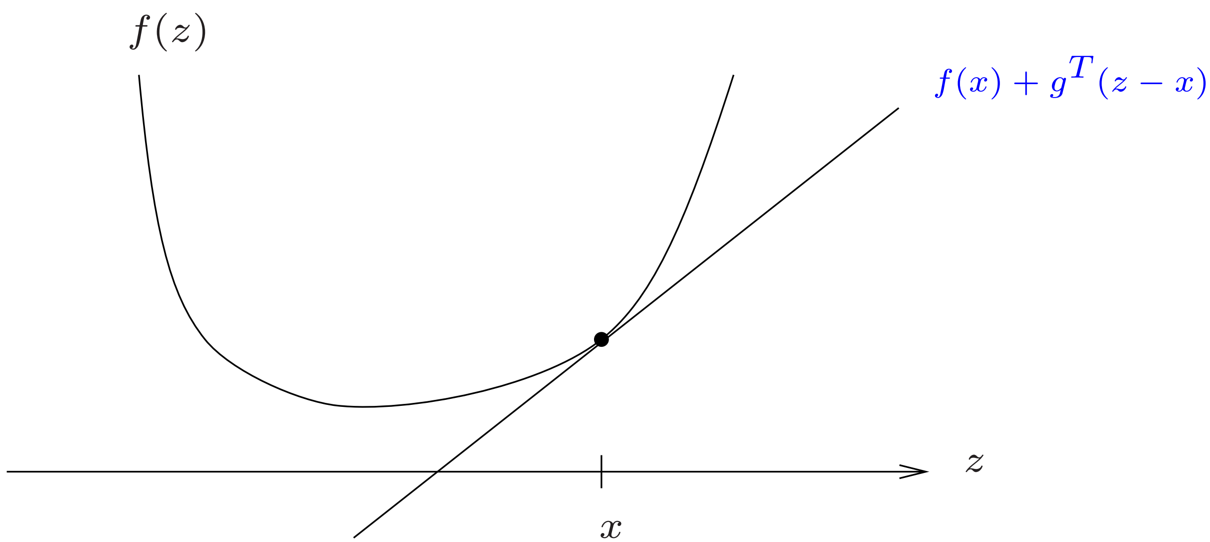
- can be applied to many different problems, including those where interior-point methods cannot be used.

- can used to decouple or decompose a large problem into many smaller ones. This has played a significant role in internet optimization, network utility max., and dynamic spectrum management in multiuser multicarrier systems.

Definition of Subgradient

- A vector $\boldsymbol{g} \in \mathbb{R}^n$ is said to be a *subgradient* of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ at $\boldsymbol{x} \in \mathrm{dom} f$ if, for all $\boldsymbol{z} \in \mathrm{dom} f$,

$$f(\boldsymbol{z}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{z} - \boldsymbol{x})$$



- If f is convex and differentiable, then its gradient $\nabla f(\boldsymbol{x})$ at $\boldsymbol{x}$ is a subgradient.

- A subgradient can exist even when f is nondifferentiable at $\boldsymbol{x}$.

Subdifferential

- A function f is called subdifferentiable at $\boldsymbol{x}$ if at least one subgradient of f exists at $\boldsymbol{x}$.

- The set of all subgradients at $\boldsymbol{x}$ is called the *subdifferential* of f at $\boldsymbol{x}$, and is denoted as
$$\partial f(\boldsymbol{x})$$

- A function f is called subdifferentiable if it is subdifferentiable at all $\boldsymbol{x} \in \mathrm{dom} f$.

Example: Absolute value

- Consider $f(x) = |x|$.

- A subgradient of f at x, denoted as g here, is

$$g = \begin{cases} 1, & x > 0 \\ -1, & x < 0 \\ \text{any value between } -1 \text{ and } 1, & x = 0 \end{cases}$$

- The subdifferential is

$$\partial f(x) = \begin{cases} \{1\}, & x > 0 \\ \{-1\}, & x < 0 \\ [-1, 1], & x = 0 \end{cases}$$

- Note that $|x|$ is not differentiable; the derivative does not exist at $x = 0$.

Basic Properties of Subgradients

- $\partial f(\boldsymbol{x})$ is a closed convex set, even for nonconvex f.

- If f is convex and $\boldsymbol{x} \in \text{int dom } f$, then $\partial f(\boldsymbol{x})$ is nonempty and bounded. (that means a convex f is *usually* subdifferentiable)

- If f is convex and differentiable, then

$$\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}.$$

- If f is convex and $\partial f(\boldsymbol{x}) = \{\boldsymbol{g}\}$, then f is differentiable at $\boldsymbol{x}$.

- $\boldsymbol{x}^\star$ is a minimizer of a convex f if and only if f is is subdifferentiable at $\boldsymbol{x}^\star$ and

$$\mathbf{0} \in \partial f(\boldsymbol{x}^\star).$$

Calculus of Subgradients

- nonnegative scaling: for $\alpha \geq 0$,

$$\partial(\alpha f)(\boldsymbol{x}) = \alpha \partial f(\boldsymbol{x})$$

- sum: Suppose $f = f_1 + \ldots + f_m$, f_i all being convex.

$$\partial f(\boldsymbol{x}) = \partial f_1(\boldsymbol{x}) + \ldots + \partial f_m(\boldsymbol{x})$$

  The same property applies to integrals.

- affine transformation of domain: Suppose f is convex, and let $h(\boldsymbol{x}) = f(\boldsymbol{A}\boldsymbol{x}+\boldsymbol{b})$.

$$\partial h(\boldsymbol{x}) = \boldsymbol{A}^T \partial f(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}).$$

- pointwise max.: Suppose $f_1, \ldots, f_m$ are convex, and let $f(\boldsymbol{x}) = \max_{i=1,\ldots,m} f_i(\boldsymbol{x})$.

$$\partial f(\boldsymbol{x}) = \mathrm{conv} \cup \{\partial f_i(\boldsymbol{x}) | f_i(\boldsymbol{x}) = f(\boldsymbol{x})\}$$

Example: Pointwise Linear Function

- Consider

$$f(\boldsymbol{x}) = \max_{i=1,\ldots,m} \boldsymbol{a}_i^T \boldsymbol{x} + b_i$$

- Let $f_i(\boldsymbol{x}) = \boldsymbol{a}_i^T \boldsymbol{x} + b_i$. We have $\partial f_i(\boldsymbol{x}) = \{\boldsymbol{a}_i\}$.

- Let $\mathcal{K}(\boldsymbol{x}) = \left\{ j \;\middle|\; \boldsymbol{a}_j^T \boldsymbol{x} + b_j = \max_{i=1,\ldots,m} \boldsymbol{a}_i^T \boldsymbol{x} + b_i \right\}$.

$$\partial f(\boldsymbol{x}) = \operatorname{conv} \bigcup_{j \in \mathcal{K}(\boldsymbol{x})} \{\boldsymbol{a}_j\}$$

- In particular, when $\mathcal{K}(\boldsymbol{x}) = \{k\}$, we have $\partial f(\boldsymbol{x}) = \{\boldsymbol{a}_k\}$.

Example: 1-norm

- Consider

$$f(\boldsymbol{x}) = \|\boldsymbol{x}\|_1 = \underbrace{|x_1|}_{f_1} + \ldots + \underbrace{|x_n|}_{f_n}$$

- Its subdifferential is

$$\begin{aligned}\partial f(\boldsymbol{x}) &= \partial f_1(\boldsymbol{x}) + \ldots + \partial f_m(\boldsymbol{x}) \\ &= \left\{\boldsymbol{g} \;\middle|\; g_i = 1 \text{ if } x_i > 0,\ g_i = -1 \text{ if } x_i < 0,\ g_i \in [-1,1] \text{ if } x_i = 0\right\}\end{aligned}$$

- Alternatively,

$$f(\boldsymbol{x}) = \max_{\boldsymbol{s}\in\{-1,1\}^n} \underbrace{\boldsymbol{s}^T\boldsymbol{x}}_{f_{\boldsymbol{s}}(\boldsymbol{x})}$$

and

$$\begin{aligned}\partial f(\boldsymbol{x}) &= \mathrm{conv} \bigcup \{\partial f_{\boldsymbol{s}}(\boldsymbol{x}) | \boldsymbol{s}^T\boldsymbol{x} = \|\boldsymbol{x}\|_1, \boldsymbol{s} \in \{-1,1\}^n\} \\ &= \{\boldsymbol{s} | \boldsymbol{s}^T\boldsymbol{x} = \|\boldsymbol{x}\|_1, \boldsymbol{s} \in [-1,1]^n\}\end{aligned}$$

- To put it simple, $\mathrm{sign}(\boldsymbol{x})$ is a subgradient of f at $\boldsymbol{x}$.

Supremum

- The pointwise max. result can be extended to supremum. Suppose

$$f(\boldsymbol{x}) = \sup_{\alpha \in \mathcal{A}} f_\alpha(\boldsymbol{x})$$

where f_α are subdifferentiable and $\mathcal{A}$ is compact.

$$\partial f(\boldsymbol{x}) = \mathrm{conv} \cup \{\partial f_\alpha(\boldsymbol{x}) \mid f_\alpha(\boldsymbol{x}) = f(\boldsymbol{x})\}$$

- **Example:** Consider $f(\boldsymbol{x}) = \lambda_{\sf max}(\boldsymbol{A}(\boldsymbol{x}))$, $\boldsymbol{A}(\boldsymbol{x}) = \boldsymbol{A}_0 + \sum_{i=1}^n x_i \boldsymbol{A}_i$. Since

$$\lambda_{\sf max}(\boldsymbol{A}(\boldsymbol{x})) = \sup_{\|\boldsymbol{y}\|_2 = 1} f_{\boldsymbol{y}}(\boldsymbol{x}), \qquad f_{\boldsymbol{y}}(\boldsymbol{x}) = \boldsymbol{y}^T \boldsymbol{A}(\boldsymbol{x}) \boldsymbol{y}$$

we have

$$\partial f(\boldsymbol{x}) = \mathrm{conv} \cup \left\{ (\boldsymbol{y}^T \boldsymbol{A}_1 \boldsymbol{y}, \ldots, \boldsymbol{y}^T \boldsymbol{A}_n \boldsymbol{y}) \mid \boldsymbol{y} \text{ a principal eigenvector of } \boldsymbol{A}(\boldsymbol{x}) \right\}$$

In particular, if the max. eigenvector of $\boldsymbol{A}(\boldsymbol{x})$, $\boldsymbol{y}$, is unique,

$$\partial f(\boldsymbol{x}) = \{(\boldsymbol{y}^T \boldsymbol{A}_1 \boldsymbol{y}, \ldots, \boldsymbol{y}^T \boldsymbol{A}_n \boldsymbol{y})\}.$$

The Subgradient Method for Unconstrained Opt.

- The goal is to solve

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x})$$

- A basic subgradient method:

> **given** $\{\alpha_k\}$, a step size sequence; & an initial point $\boldsymbol{x}^{(0)}$.
> $k := 0$; $i_{\text{best}} := 0$.
> **repeat**
> $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)}$, where $\boldsymbol{g}^{(k)}$ is any subgradient of f at $\boldsymbol{x}^{(k)}$.
> $k := k + 1$.
> $f_{\text{best}}^{(k)} = \min\{f_{\text{best}}^{(k-1)}, f(\boldsymbol{x}^{(k)})\}$. If $f(\boldsymbol{x}^{(k)}) = f_{\text{best}}^{(k)}$, then $i_{\text{best}} := k$.
> **until** a stopping criterion is satisfied.
> **output** $\boldsymbol{x}^{(i_{\text{best}})}$.

- Look similar to the gradient descent method (for differentiable f), but not the same.

- choose the best point among the generated sequence $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots$.

Step Size Rules

There are many different choices for the step sizes. Some typical rules are

- Constant step size: $\alpha_k = \alpha$.
- Constant step length: $\alpha_k = \gamma / \|\boldsymbol{g}^{(k)}\|_2$, where $\gamma > 0$.
- Square summable but not summable: the step sizes satisfy

$$\alpha_k \geq 0, \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty, \quad \sum_{k=1}^{\infty} \alpha_k = \infty$$

  An example is $\alpha_k = a/(b+k)$, where $a, b > 0$.

- Nonsummable diminishing: The step sizes satisfy

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

  An example is $\alpha_k = a/\sqrt{k}$, where $a > 0$.

Convergence

Let $f^\star = \inf_{\boldsymbol{x}} f(\boldsymbol{x})$, and G be such that $\|\boldsymbol{g}^{(k)}\|_2 \leq G$ for all k.

- Constant step size $\alpha_k = \alpha$:

$$\lim_{k\to\infty} f_{\text{best}}^{(k)} - f^\star \leq G^2\alpha/2$$

- Constant step length $\alpha_k = \gamma/\|\boldsymbol{g}^{(k)}\|_2$, $\gamma > 0$:

$$\lim_{k\to\infty} f_{\text{best}}^{(k)} - f^\star \leq G\gamma/2$$

- Square summable but not summable; and nonsummable diminishing:

$$\lim_{k\to\infty} f_{\text{best}}^{(k)} = f^\star$$

Given a solution precision ϵ, the number of iterates k for achieving $f_{\text{best}}^{(k)} - f^\star < \epsilon$ can also be proven.

Example: Minimizing a piece-wise linear function

- Consider

$$\min_{\boldsymbol{x}} \left(\max_{i=1,\ldots,m} \boldsymbol{a}_i^T \boldsymbol{x} + b_i \right)$$

- At the k iteration, find an (any) index for which

$$\boldsymbol{a}_j^T \boldsymbol{x}^{(k)} + b_j = \max_{i=1,\ldots,m} \boldsymbol{a}_i^T \boldsymbol{x}^{(k)} + b_i$$

  and we have

$$\boldsymbol{g}^{(k)} = \boldsymbol{a}_j.$$

Example: Solving SDPs

- The basic subgradient method may be used to solve SDPs (are you sure?)
- For simplicity, consider

$$
\begin{aligned}
\min_{\boldsymbol{X}} \ & \mathrm{Tr}(\boldsymbol{C}\boldsymbol{X}) \\
\text{s.t.} \ & X_{ii} = 1, \ i = 1, \ldots, n \\
& \boldsymbol{X} \succeq \mathbf{0}
\end{aligned}
$$

which has well-known applications in approximating MAXCUT & ML MIMO detection.

Example: Solving SDPs (cont'd)

- Let us add a redundant equality to the SDP

$$
\begin{aligned}
\min_{\boldsymbol{X}} \ & \mathrm{Tr}(\boldsymbol{C}\boldsymbol{X}) \\
\text{s.t.} \ & \boldsymbol{X} \succeq \boldsymbol{0}, \quad X_{ii} = 1, \ i = 1, \ldots, n \\
& \mathrm{Tr}(\boldsymbol{X}) = n
\end{aligned}
$$

  The dual of the SDP above is

$$
\begin{aligned}
\max_{\boldsymbol{\mu}, \nu} \ & -\boldsymbol{\mu}^T \mathbf{1} - n\nu \\
\text{s.t.} \ & \boldsymbol{C} + \boldsymbol{D}(\boldsymbol{\mu}) + \nu \boldsymbol{I} \succeq \boldsymbol{0}
\end{aligned}
$$

- Since

$$
\boldsymbol{C} + \boldsymbol{D}(\boldsymbol{\mu}) + \nu \boldsymbol{I} \succeq \boldsymbol{0} \Longleftrightarrow \lambda_{\min}(\boldsymbol{C} + \boldsymbol{D}(\boldsymbol{\mu})) \geq -\nu
$$

  we can rewrite the dual problem as an unconstrained problem

$$
\max_{\boldsymbol{\mu}} \ -\boldsymbol{\mu}^T \mathbf{1} + n\lambda_{\min}(\boldsymbol{C} + \boldsymbol{D}(\boldsymbol{\mu}))
$$

Example: Solving SDPs (cont'd)

- Now we deal with the dual problem

$$\max_{\boldsymbol{\mu}} \; d(\boldsymbol{\mu}) \triangleq -\boldsymbol{\mu}^T \mathbf{1} + n\lambda_{\min}(\boldsymbol{C} + \boldsymbol{D}(\boldsymbol{\mu}))$$

  by subgradient.

- A subgradient of $-d(\boldsymbol{\mu})$ at $\boldsymbol{\mu}$ is

$$\boldsymbol{g} = \mathbf{1} - n\boldsymbol{q}_{\min}^2$$

  where the superscript 2 denotes the elementwise square, and $\boldsymbol{q}_{\min}$ is a minimum eigenvector of $\boldsymbol{C} + \boldsymbol{D}(\boldsymbol{\mu})$.

Example: Solving SDPs (cont'd)

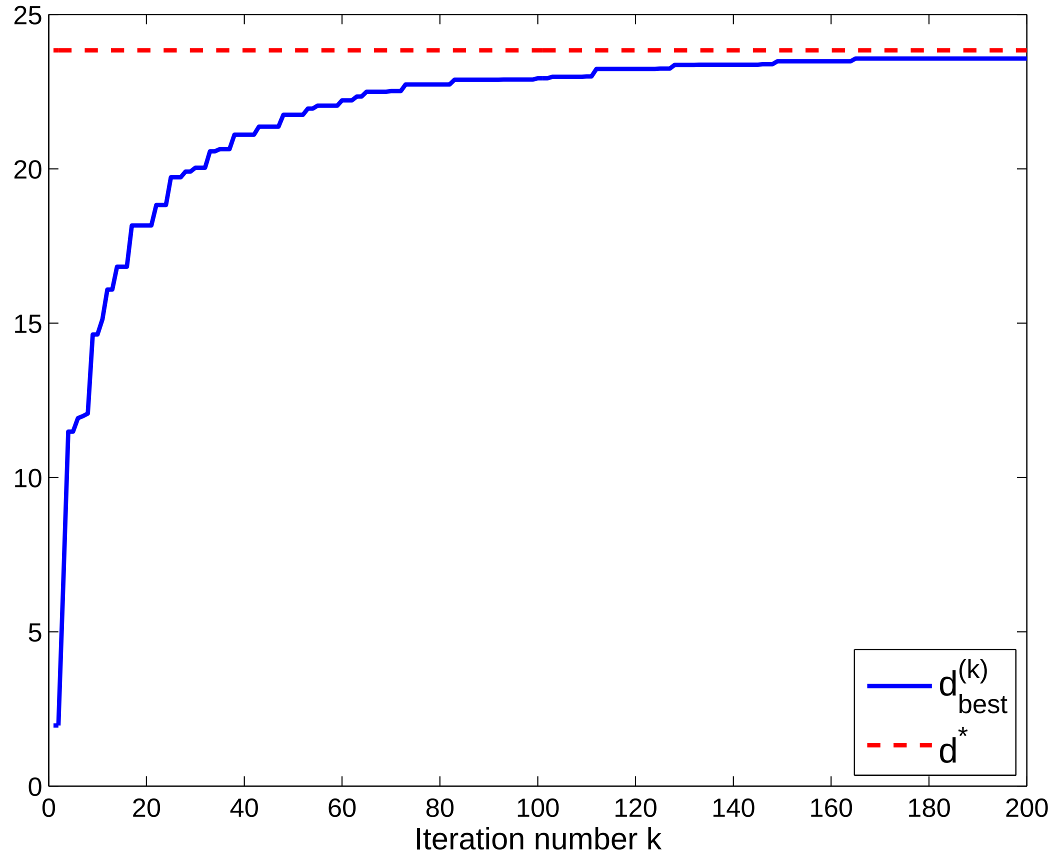


Figure 1: The value $d_{\text{best}}^{(k)}$ versus the iteration number k, for the subgradient method for SDP. The problem size is $n = 20$, and the step size rule is $\alpha_k = 1/\sqrt{k}$.

The Projected Subgradient Method

- The goal is to solve

$$\min_{\boldsymbol{x}\in\mathcal{C}} f(\boldsymbol{x})$$

  where $\mathcal{C}$ is a convex set.

- In the projected subgradient method, the iterates are obtained by

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{\mathcal{P}}_{\mathcal{C}}\left(\boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)}\right),$$

  where $\boldsymbol{\mathcal{P}}_{\mathcal{C}}$ is the Euclidean projection on $\mathcal{C}$; i.e.,

$$\boldsymbol{\mathcal{P}}_{\mathcal{C}}(\boldsymbol{x}) = \arg\min_{\boldsymbol{y}\in\mathcal{C}} \|\boldsymbol{y} - \boldsymbol{x}\|_2^2$$

- The convergence result is similar to that of the basic subgradient method.

Example: 1-norm minimization

- Consider

$$\begin{aligned} &\min \ \|\boldsymbol{x}\|_1 \\ &\text{s.t.} \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \end{aligned}$$

  where $\boldsymbol{A}$ is fat.

- We have $\text{sign}(\boldsymbol{x}) \in \partial f(\boldsymbol{x})$

- We have $\mathcal{C} = \{\boldsymbol{x} \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$, and

$$\boldsymbol{\mathcal{P}}(\boldsymbol{y}) = \boldsymbol{A}^\dagger \boldsymbol{b} + (\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^\dagger)\boldsymbol{y},$$

  where $\boldsymbol{A}^\dagger = \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{A}^T)^{-1}$.

- The corresponding projected gradient update is

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^\dagger)\text{sign}(\boldsymbol{x})$$

Example: 1-norm minimization (cont'd)

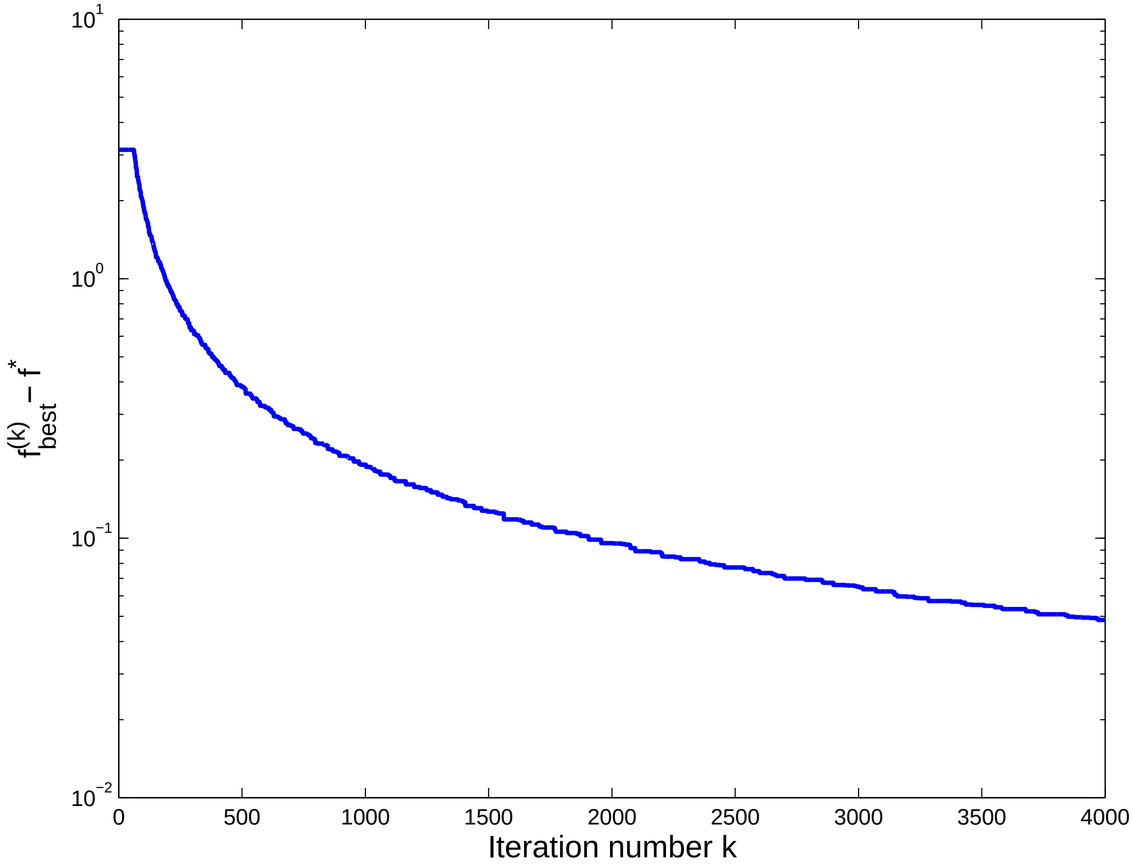


Figure 2: The gap $f^{(k)}_{\text{best}} - f^\star$ versus the iteration number k, for the projected subgradient method for 1-norm minimization. The problem size is $m = 50$, $n = 1000$, and the step size rule is $\alpha_k = 0.5/k$.

The projected subgradient method is efficient only when the projection on $\mathcal{C}$ can be easily computed; e.g.,

- An affine set: linear projection

- A halfspace: similar to affine sets

- The set of non-ve nos. $\mathcal{C} = \mathbb{R}^n_+$, a box $\mathcal{C} = \{\boldsymbol{x} \mid -1 \leq x_i \leq 1, i = 1, \ldots, n\}$: projection is truncation

- A 2-norm ball $\mathcal{C} = \{\boldsymbol{x} \mid \|\boldsymbol{x}\|_2 \leq 1\}$: projection is rescaling.

- An ellipsoid: no closed form, but can be easily computed.

- Simplex $\mathcal{C} = \{\boldsymbol{x} \succeq \mathbf{0} \mid \boldsymbol{x}^T\mathbf{1} \leq 1\}$: no closed form, but can be easily computed.

- The cone of PSD matrices: projection is to discard eigen-components that are -ve.

Projected Subgradient for Dual Problems

- We consider a constrained, not necessarily convex, problem

$$\begin{aligned} \min_{\boldsymbol{x}} \ & f_0(\boldsymbol{x}) \\ \text{s.t.} \ & f_i(\boldsymbol{x}) \leq 0, \quad i = 1, \ldots, m \end{aligned}$$

- We focus on dealing with its dual

$$\begin{aligned} \max_{\boldsymbol{\lambda}} \ & d(\boldsymbol{\lambda}) \\ \text{s.t.} \ & \boldsymbol{\lambda} \succeq \mathbf{0} \end{aligned}$$

where $d(\boldsymbol{\lambda}) = \inf_{\boldsymbol{x}} \left(f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) \right)$.

- Recall that $d(\boldsymbol{\lambda})$ is always concave, even when the primal problem is nonconvex.

- The projected subgradient method can be applied, if we can compute the subgradients of $d(\boldsymbol{\lambda})$.

- Let

$$\boldsymbol{x}^\star(\boldsymbol{\lambda}) = \arg\min_{\boldsymbol{x}} \left(f_0(\boldsymbol{x}) + \textstyle\sum_{i=1}^m \lambda_i f_i(\boldsymbol{x})\right)$$

denote a minimizer that attains $d(\boldsymbol{\lambda})$. We can write

$$d(\boldsymbol{\lambda}) = f_0(\boldsymbol{x}^\star(\boldsymbol{\lambda})) + \sum_{i=1}^m \lambda_i f_i(\boldsymbol{x}^\star(\boldsymbol{\lambda}))$$

- A subgradient of $-d$ at $\boldsymbol{\lambda}$ is then

$$-(f_1(\boldsymbol{x}^\star(\boldsymbol{\lambda})), \ldots, f_m(\boldsymbol{x}^\star(\boldsymbol{\lambda}))) \in \partial(-d)(\boldsymbol{\lambda})$$

- The updates of projected subgradient applied to dual max. is

$$
\begin{aligned}
&\text{solve for } \boldsymbol{x}^{(k)} = \arg\min_{\boldsymbol{x}} \left(f_0(\boldsymbol{x}) + \textstyle\sum_{i=1}^m \lambda_i^{(k)} f_i(\boldsymbol{x}) \right) \\
&\lambda_i^{(k+1)} = \left(\lambda_i^{(k)} + \alpha_k f_i(\boldsymbol{x}^{(k)}) \right)_+, \quad i = 1, \ldots, m
\end{aligned}
$$

  where $(\cdot)_+$ is the projection on $\mathbb{R}_+$. That is say, we are solving a sequence of unconstrained Lagrangian minimization.

- The projected subgradient method is a Lagrangian dual relaxation, in general. The generated points $\boldsymbol{x}^{(k)}$ may not be primal feasible.

- Suppose strong duality holds (e.g., convex problems with the Slater condition), and each $\boldsymbol{x}^{(k)}$ is a unique minimizer. Then the limit point of $\boldsymbol{x}^{(k)}$ is primal feasible (in fact, optimal).

- This dual max. approach, a.k.a. dual decomposition in some applications, plays a significant role.

Application: Dynamic Spectrum Management (DSM)

- Scenario: A multiuser subcarrier system, with K users and N subcarriers.

- Goal: joint power allocation for sum rate maximization

$$\begin{aligned} \max \ & \sum_{n=1}^{N} \sum_{k=1}^{K} \log\left(1 + \mathsf{SINR}_k^n(s_1^n, \ldots, s_K^n)\right) && \text{(rates of all subcarriers \& users)} \\ \text{s.t.} \ & \sum_{n=1}^{N} s_k^n \leq P_k, \quad k = 1, \ldots, K && \text{(per-user total power constraint)} \\ & 0 \leq s_k^n \leq S_{\max}, \quad \forall k, n && \text{(per-subcarrier power limit)} \end{aligned}$$

where the opt. variable s_k^n is the power of kth user at subcarrier n, and

$$\mathsf{SINR}_k^n(s_1^n, \ldots, s_K^n) = \frac{\alpha_{kk}^n s_k^n}{\sigma_k^n + \sum_{j \neq k} \alpha_{kj}^n s_j^n}$$

is the SINR of user k at subcarrier n.

Why DSM is hard?

- The DSM sum rate max. problem

$$
\begin{aligned}
\max \ & \sum_{n=1}^{N} \sum_{k=1}^{K} \log \left(1 + \mathsf{SINR}_k^n(s_1^n, \ldots, s_K^n)\right) \\
\text{s.t.} \ & \sum_{n=1}^{N} s_k^n \leq P_k, \quad k = 1, \ldots, K \\
& 0 \leq s_k^n \leq S_{\max}, \quad k = 1, \ldots, K, n = 1, \ldots, N
\end{aligned}
$$

  is nonconvex, even for $N = 1$.

- It is NP-hard in general.

- The no. of subcarriers N can be large; e.g., $N = 256$, $N = 1024$, ..., and per-user power constraints make the rate max. coupled w.r.t. subcarriers.

- The dual of the DSM problem is

$$\begin{aligned} \min\ & \boldsymbol{p}^T \boldsymbol{\lambda} + \varphi(\boldsymbol{\lambda}) \\ \text{s.t. } & \boldsymbol{\lambda} \succeq \mathbf{0} \end{aligned}$$

where $\boldsymbol{p} = (P_1, \ldots, P_K)$,

$$\begin{aligned} \varphi(\boldsymbol{\lambda}) = \max\ & \left(\sum_{n=1}^{N} \sum_{k=1}^{K} \log\left(1 + \mathsf{SINR}_k^n(s_1^n, \ldots, s_K^n)\right) - \lambda_k s_k^n \right) \\ \text{s.t. } & 0 \leq s_k^n \leq S_{\max}, k = 1, \ldots, K, n = 1, \ldots, N \end{aligned}$$

- An important result is that

$$\varphi(\boldsymbol{\lambda}) = \sum_{n=1}^{N} \max_{\substack{s_1^n, \ldots, s_K^n, \\ 0 \leq s_k^n \leq S_{\max}}} \left(\sum_{k=1}^{K} \log\left(1 + \mathsf{SINR}_k^n(s_1^n, \ldots, s_K^n)\right) - \lambda_k s_k^n \right)$$

i.e., $\varphi(\boldsymbol{\lambda})$ decomposes to many per-subcarrier power allocation problems.

- What remains is that we need to solve the per-subcarrier problems

$$\varphi_n(\boldsymbol{\lambda}) = \max_{\substack{s_1^n,\ldots,s_K^n,\\ 0\leq s_k^n\leq S_{\max}}} \left(\sum_{k=1}^{K} \log\left(1 + \mathsf{SINR}_k^n(s_1^n,\ldots,s_K^n)\right) - \lambda_k s_k^n \right)$$

  for $n = 1, \ldots, N$.

- The problem above is still nonconvex.

- For $K = 2$, exhaustive search was used (OSB **[Cendrillon *et al.*'06]**).

- For $K > 2$, some approximation methods should be used.

- For the OFDMA variation (one subcarrier can only be occupied by one user), there is a simple way of solving the per-subcarrier problem **[Luo-Zhang'09]**.

- (there are many more refs. & nice results in DSM that I have no time to mention here)

Optimal value of a convex opt. problem

- Consider the optimal value of a convex optimization problem

$$\begin{aligned} \phi(\boldsymbol{x},\boldsymbol{y}) = \min_{\boldsymbol{z}} \ & f_0(\boldsymbol{z}) \\ \text{s.t.} \ & f_i(\boldsymbol{z}) \le x_i, \ i=1,\ldots,m, \quad \boldsymbol{A}\boldsymbol{z}=\boldsymbol{y} \end{aligned}$$

  where $f_0, f_1, \ldots, f_m$ are convex. Its dual is

$$\begin{aligned} \phi(\boldsymbol{x},\boldsymbol{y}) = \max_{\boldsymbol{\lambda},\boldsymbol{\mu}} \ & g(\boldsymbol{\lambda},\boldsymbol{\mu}) - \boldsymbol{x}^T\boldsymbol{\lambda} - \boldsymbol{y}^T\boldsymbol{\mu} \\ \text{s.t.} \ & \boldsymbol{\lambda} \succeq \mathbf{0} \end{aligned}$$

  where $g(\boldsymbol{\lambda},\boldsymbol{\mu}) = \inf_{\boldsymbol{z}}(f_0(\boldsymbol{z}) + \sum_{i=1}^m \lambda_i f_i(\boldsymbol{z}) + \boldsymbol{\mu}^T\boldsymbol{A}\boldsymbol{z})$.

- Suppose that strong duality holds at $(\boldsymbol{x},\boldsymbol{y})$, & let $(\boldsymbol{\lambda}^\star,\boldsymbol{\mu}^\star)$ be a dual opt. solution fixing $(\boldsymbol{x},\boldsymbol{y})$.

$$-(\boldsymbol{\lambda}^\star,\boldsymbol{\mu}^\star) \in \partial\phi(\boldsymbol{x},\boldsymbol{y})$$

- This property is useful, e.g., in primal decomposition methods.

Application: MIMO BC Capacity

- Scenario: A multiuser MIMO broadcast channel (BC).
- Goal: Solve the MIMO BC capacity, which has been shown to be

$$
\begin{aligned}
\max_{\boldsymbol{Q}_1,\ldots,\boldsymbol{Q}_K} \quad & \log\det\left(\boldsymbol{I} + \sum_{k=1}^{K} \boldsymbol{H}_k \boldsymbol{Q}_k \boldsymbol{H}_k^H\right) \\
\text{s.t.} \quad & \sum_{k=1}^{K} \mathrm{Tr}(\boldsymbol{Q}_k) \leq P_{\text{total}} \\
& \boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_K \succeq \boldsymbol{0}
\end{aligned}
$$

  where $\boldsymbol{H}_k$ is MIMO channel from the basestation to user k.

- This problem is convex (CVX can do the job).
- Can we derive a simple algorithm by using the subgradient concepts?

A Related Problem: MIMO MAC Capacity

- To solve MIMO BC, let us look at a related problem— MIMO multiple access channel (MAC).

$$\begin{aligned} \max_{\boldsymbol{Q}_1,\ldots,\boldsymbol{Q}_K \succeq \boldsymbol{0}} \quad & \log\det\left(\boldsymbol{I} + \sum_{k=1}^{K} \boldsymbol{H}_k \boldsymbol{Q}_k \boldsymbol{H}_k^H\right) \\ \text{s.t.} \quad & \mathrm{Tr}(\boldsymbol{Q}_k) \leq P_k, \quad k = 1, \ldots, K \end{aligned}$$

  where P_k is the total power limit of user k.

- The MIMO MAC capacity is convex.

- A convenient way of solving the MIMO MAC capacity is to use the iterative water filling algorithm (IWFA):
  - at each iteration, maximize the objective fn. w.r.t. a $\boldsymbol{Q}_k$ while fixing the other $\{\boldsymbol{Q}_\ell\}_{\ell \neq k}$.
  - the maximization at each iteration is a single-user water filling problem.

Projected Subgradient for MIMO BC Capacity

- We can write the MIMO BC capacity as

$$\max_{P_1,\ldots,P_K} \underbrace{\begin{pmatrix} \max_{\boldsymbol{Q}_1,\ldots,\boldsymbol{Q}_K \succeq \boldsymbol{0}} & \log\det\left(\boldsymbol{I} + \sum_{k=1}^K \boldsymbol{H}_k \boldsymbol{Q}_k \boldsymbol{H}_k^H\right) \\ \text{s.t.} & \mathrm{Tr}(\boldsymbol{Q}_k) \leq P_k, \quad k = 1, \ldots, K \end{pmatrix}}_{\triangleq \phi(\boldsymbol{p})}$$

$$\text{s.t. } \boldsymbol{p}^T \boldsymbol{1} \leq P_{\text{total}}$$

that is, we use the subgradients of $-\phi(\boldsymbol{p})$ to solve the MIMO BC capacity.

- Specifically, the updates in projected subgradient are

find $\boldsymbol{\lambda}^{(k)}$ that is an optimal dual solution of $\phi(\boldsymbol{p}^{(k)})$, by IWFA.

$$\boldsymbol{p}^{(k+1)} = \left(\boldsymbol{p}^{(k)} + \alpha_k \boldsymbol{\lambda}^{(k)}\right)_{\mathcal{S}}$$

where $\mathcal{S}$ is the projection on the simplex $\mathcal{S} = \{\boldsymbol{p} \mid \boldsymbol{p}^T \boldsymbol{1} \leq P_{\text{total}}\}$ (no closed form, but can be easily computed [hint: it's like water filling]).

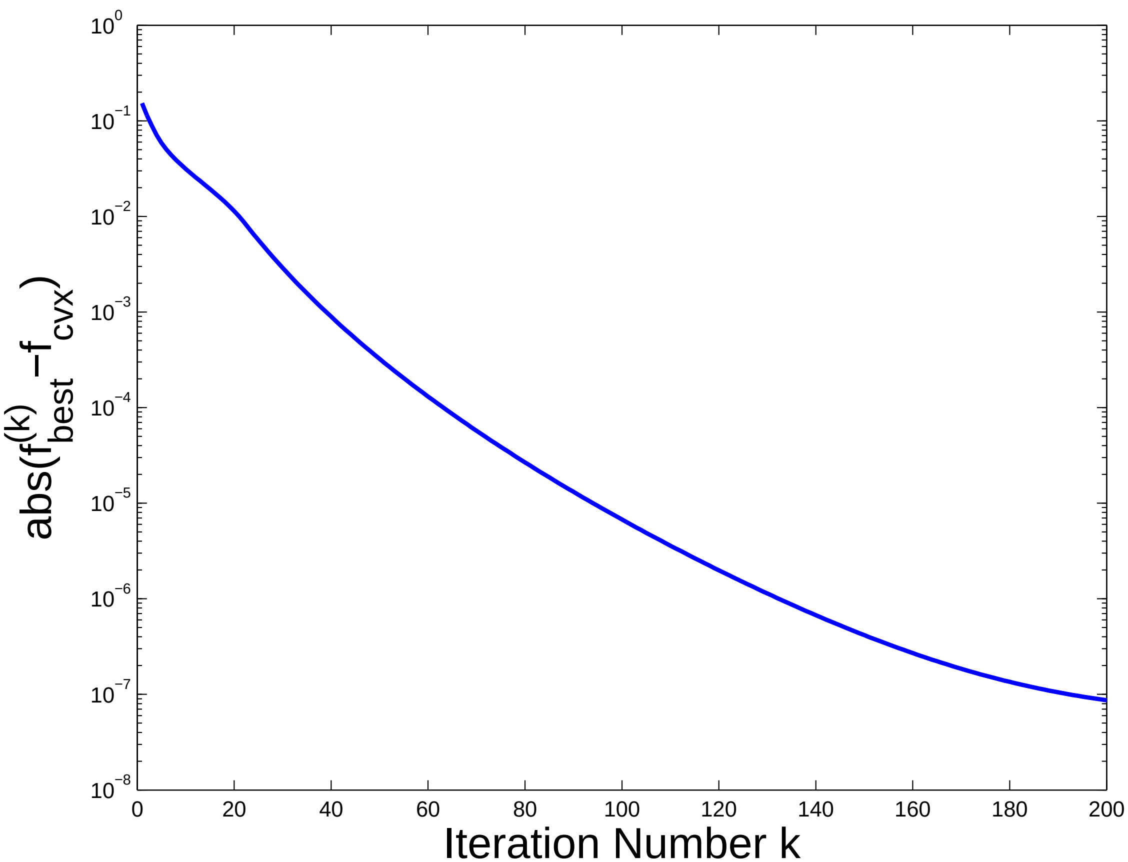


Figure 3: The gap $f_{\text{best}}^{(k)} - f^{\star}$ versus the iteration number k, for the projected subgradient method applied to MIMO BC capacity computations. $M_t = 12$, $M_r = 4$, $K = 3$, $P_{\text{total}} = 100$, and the step size rule is $\alpha_k = 9/\sqrt{k}$.

Subgradient Method for Constrained Optimization

- Consider a convex problem

$$
\begin{aligned}
&\min\ f_0(\boldsymbol{x}) \\
&\text{ s.t. } f_i(\boldsymbol{x}) \leq 0, i = 1, \ldots, m
\end{aligned}
$$

  We have seen that by applying the subgradient method to the dual, the problem can be solved.

- The subgradient method can also be applied directly to the primal.

- The method takes the same form

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

  where

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

Discussion

- Subgradient methods may provide low-complexity implementations to certain problems, but possibly with low accuracy.

- There are approaches that can speed up convergence; e.g., ellipsoid methods, cutting plane methods, bundle methods, . . . They are more complex requiring more computations to carry out the update.

References on Subgradient Methods

N. Shor, *Minimization Methods for Non-Differentiable Functions*, Springer Series in Computational Mathematics, Springer, 1985.

D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

S. Boyd and A. Mutapcic, *Subgradient Methods*, Notes for EE364b, Stanford University, 2006. Available online.

References on Applications

[Cendrillon *et al.*'06] R. Cendrillon, W. Yu, M. Moonen, J. Verlinden, and T. Bostoen, "Optimal multiuser spectrum balancing for digital subscriber lines," *IEEE Trans. Commun.*, vol. 54, no. 5, 2006.

[Luo-Zhang'09] Z.-Q. Luo and S. Zhang, "Duality gap estimation and polynomial time approximation for optimal spectrum management," *IEEE Trans. Signal Process.*, vol. 57, no. 7, 2009.

[Palomar'05] D. P. Palomar, "Convex Primal Decomposition for Multicarrier Linear MIMO Transceivers," *IEEE Trans. on Signal Processing*, vol. 53, no. 12, Dec. 2005.

[Yu'03] W. Yu, "A Dual Decomposition Approach to the Sum Power Gaussian Vector Multiple Access Channel Sum Capacity Problem," in *Proc. Conf. on Information and Systems (CISS)*, The Johns Hopkins Univ., March 12-14, 2003.