1. Introduction

- performance of numerical methods
- complexity bounds
- structural convex optimization
- course goals and topics
Some course info

Welcome to EE 546!

Instructor: Maryam Fazel, TA: Reza Eghbali

please see webpage for details:
http://www.ee.washington.edu/class/546/2016spr/

a few notes:

• pre-requisites: ee 578 or math 516 (if you have not taken these, consent of instructor is strictly needed)

• requirements: homeworks (3), course project (proposal, poster, mid-quarter+final reports)

• Maryam’s office hours: Wednesdays 10:30-11:45am; Reza’s TBA
(mathematical) optimization problem

\[
\begin{align*}
\text{minimize} & \quad f_0(x) \\
\text{subject to} & \quad f_j(x) \leq 0, \quad j = 1, \ldots, m \\
& \quad x \in S
\end{align*}
\]

- \(x = (x_1, \ldots, x_n)\): optimization variables
- \(f_0 : \mathbb{R}^n \rightarrow \mathbb{R}\): objective function
- \(f_j : \mathbb{R}^n \rightarrow \mathbb{R}, j = 1, \ldots, m\): constraint functions
- \(S\): “structural” constraints (like nonnegativity or boundedness)

**optimal solution** \(x^*\) satisfies \(f_0(x^*) \leq f_0(x)\) for all feasible \(x\)
Performance of a numerical method

numerical method $\mathcal{M}$  $\iff$ problem $\mathcal{P}$

**performance of $\mathcal{M}$ on $\mathcal{P}$**: total amount of *computational efforts* required by method $\mathcal{M}$ to *solve* the problem $\mathcal{P}$

- **to solve the problem** could mean
  - find the *exact* solution (impossible for most problems in finite time)
  - find an *approximate* solution with a small accuracy $\epsilon > 0$

- performance of $\mathcal{M}$ with respect to a *single* problem is meaningless

- need to define a model $(\mathcal{F}, \mathcal{O})$ consisting of
  - a *class* of problems $\mathcal{F}$, which have some common properties
  - an *oracle* $\mathcal{O}$, which provides $\mathcal{M}$ some information about $\mathcal{P}$ in $\mathcal{F}$

**performance of $\mathcal{M}$ on $(\mathcal{F}, \mathcal{O})$**: its performance on the *worst* problem from $\mathcal{F}$ (which may depend on $\mathcal{M}$)
General iterative scheme

**input:** a starting point $x^{(0)}$ and an accuracy $\epsilon > 0$

**initialization:** set $k = 0$, $I_{-1} = \emptyset$

- $k$ is iteration count
- $I_k$ is accumulated information set

**main loop:**

1. call oracle $O$ at $x^{(k)}$
2. update information set $I_k = I_{k-1} \cup \{x^{(k)}, O(x^{(k)})\}$
3. apply rules of method $M$ to $I_k$ and form new point $x^{(k+1)}$
4. check stopping criterion:
   - if yes then form an output $\bar{x}$
   - otherwise set $k = k + 1$ and go to 1
Measuring computational effort

- **analytical complexity:** number of calls of oracle required to solve problem $\mathcal{P}$ upto accuracy $\epsilon$ (also called *informational complexity*)

- **arithmetical complexity:** total number of arithmetic operations (including work of oracle and method itself) required to solve problem $\mathcal{P}$ upto accuracy $\epsilon$

relationships

- arithmetical complexity is more useful in practice; usually easily obtained from analytical complexity and complexity of oracle

we will mainly work with *upper/lower bounds* on analytical complexity
Black box oracle

local black box

- only information available for numerical method is answer of oracle
- oracle is *local*: small variation of problem far enough from query point $x$
  does not change answer at $x$

examples of oracle $\mathcal{O}(x)$

- *zero-order oracle*: returns function value $f(x)$
- *first-order oracle*: returns $f(x)$ and gradient $\nabla f(x)$
- *second-order oracle*: returns $f(x)$, $\nabla f(x)$ and Hessian $\nabla^2 f(x)$
Complexity bound for global optimization

**problem class** \( F \) (formulation and assumptions)

\[
\begin{align*}
\text{minimize}_{x \in B_n} & \quad f(x) \\
\end{align*}
\]

- \( B_n = \{ x \in \mathbb{R}^n \mid 0 \leq x_i \leq 1, \ i = 1, \ldots, n \} \)
- \( f(x) \) Lipschitz continuous on \( B_n \): there exist \( L > 0 \) such that
  \[
  |f(x) - f(y)| \leq L \|x - y\|_2, \quad \forall x, y \in B_n
  \]

**zero-order oracle:** \( \mathcal{O}(x) = f(x) \)

**goal:** find \( \bar{x} \in B_n \) such that \( f(\bar{x}) - f^* \leq \epsilon \)
Uniform grid method

method $G(\epsilon)$

1. let $p = \left\lfloor \frac{L\sqrt{n}}{2\epsilon} \right\rfloor + 1$ and form $(p + 1)^n$ points

   $$x^{(k_1,\ldots,k_n)} = \left( \frac{k_1}{p}, \ldots, \frac{k_n}{p} \right), \quad k_1 = 0,\ldots,p, \quad \ldots, \quad k_n = 0,\ldots,p$$

2. among all points $x^{(k_1,\ldots,k_n)}$, find $\bar{x}$ that has minimal objective value

3. return the pair $(\bar{x}, f(\bar{x}))$ as a result

(can be treated as an iterative process with $(p + 1)^n$ iterations)

**Theorem:** analytical complexity of $G$ on model $(\mathcal{F}, \mathcal{O})$ is $\left( \frac{L\sqrt{n}}{2\epsilon} + 2 \right)^n$
proof: let $x^*$ be a global solution, then there exist $(k_1, \ldots, k_n)$ such that

$$x^{(k_1, \ldots, k_n)} \leq x^* \leq x^{(k_1+1, \ldots, k_n+1)}$$

(element-wise inequality)

let $\hat{x} = \frac{1}{2}(x^{(k_1, \ldots, k_n)} + x^{(k_1+1, \ldots, k_n+1)})$ and

$$\tilde{x} = \begin{cases} 
  x_i^{(k_1+1, \ldots, k_n+1)}, & \text{if } x_i^* \geq \hat{x}_i \\
  x_i^{(k_1, \ldots, k_n)}, & \text{otherwise}
\end{cases}$$

then $|\tilde{x}_i - x_i^*| \leq \frac{1}{2p}$ for all $i$, therefore

$$\|\tilde{x} - x^*\|_2^2 = \sum_{i=1}^{n} (\tilde{x}_i - x_i^*)^2 \leq \frac{n}{4p^2}$$

since $\tilde{x}$ belongs to the grid, and $p = \left\lfloor \frac{L\sqrt{n}}{2\epsilon} \right\rfloor + 1 \geq \frac{L\sqrt{n}}{2\epsilon}$, we conclude

$$f(\bar{x}) - f^* \leq f(\tilde{x}) - f^* \leq L\|\tilde{x} - x^*\|_2 \leq \frac{L\sqrt{n}}{2p} \leq \epsilon$$
Lower complexity bound

questions:

• how good is this bound? (maybe our proof is too rough)
• how good is this method? (there may exist much better algorithms)

lower complexity bound

• based on black box concept
• valid for all reasonable iterative schemes working with the model \((\mathcal{F}, \mathcal{O})\)
• often use the idea of a resisting oracle
  – tries to create a worst problem for a particular method
  – starts from an “empty” function and tries to answer each call in worst possible way
  – however, must be compatible with previous answers and \(\mathcal{F}\)
    (after termination, it is possible to reconstruct the problem)
Lower bound for global optimization

**Problem class** $\mathcal{F}$ (formulation and assumptions)

\[
\text{minimize}_{x \in B_n} \ f(x)
\]

- $B_n = \{x \in \mathbb{R}^n \mid 0 \leq x_i \leq 1, \ i = 1, \ldots, n\}$
- $f(x)$ Lipschitz continuous on $B_n$: there exist $L > 0$ such that
  \[
  |f(x) - f(y)| \leq L\|x - y\|_2, \quad \forall \ x, y \in B_n
  \]

**Zero-order oracle:** $\mathcal{O}(x) = f(x)$

**Theorem:** Analytical complexity of this model $(\mathcal{F}, \mathcal{O})$ is at least $\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor \right)^n$
Proof of lower bound

define resisting oracle

$$\mathcal{O}(x) \text{ returns } f(x) = 0 \text{ at any test point } x$$

therefore any method can only return $$\bar{x}$$ with $$f(\bar{x}) = 0$$

construct worst function

• let $$p = \left\lfloor \frac{L}{2\epsilon} \right\rfloor \geq 1$$, then for any method that takes less than $$p^n$$ calls,

there exist $$x^* \in B_n$$ such that there is no test point in the box

$$B = \left\{ x \mid \|x - x^*\|_\infty \leq \frac{1}{2p} \right\}$$

• consider the function

$$\bar{f}(x) = \min\{0, \ L\|x - x^*\|_\infty - \epsilon\}$$

optimal value: $$\min_{x \in B_n} \bar{f}(x) = \bar{f}(x^*) = -\epsilon$$
check compatibility

\[
\bar{f}(x) = \min \{0, \, L \|x - x^*\|_\infty - \epsilon\}
\]

- \(\bar{f}(x)\) is Lipschitz continuous with parameter \(L\)
  \[
  |\bar{f}(x) - \bar{f}(y)| \leq L \|x - y\|_\infty \leq L \|x - y\|_2
  \]

- function \(\bar{f}(x)\) is non-zero only inside the box
  \[
  B' = \{x \mid \|x - x^*\|_\infty \leq \epsilon/L\}
  \]

- since \(p = \lfloor \frac{L}{2\epsilon} \rfloor \leq \frac{L}{2\epsilon}\), we conclude that
  \[
  B' \subseteq B
  \]
  therefore \(\bar{f}(x)\) equals zero at all test points

**Conclusion:** accuracy no less than \(\epsilon\) if number of oracle calls less than \(p^n\)
Complexity of global optimization

<table>
<thead>
<tr>
<th>uniform grid complexity</th>
<th>lower bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \left( \frac{L\sqrt{n}}{2\epsilon} \right)^n )</td>
<td>( \left( \frac{L}{2\epsilon} \right)^n )</td>
</tr>
</tbody>
</table>

- dependence on \( \epsilon \) is \textit{optimal}
- dependence on \( n \) is \textit{not optimal}

The conclusion depends on the problem class \( \mathcal{F} \): if we assume

\[
|f(x) - f(y)| \leq L \|x - y\|_{\infty}, \quad \forall x, y \in B_n
\]

then uniform grid method has complexity \( \left( \frac{L}{2\epsilon} \right)^n \), and it is \textit{optimal}

\textbf{question:} will higher-order oracles help improve complexity results?
Common classes and features

- **global optimization**
  - *goal*: find a global minimum
  - *problem class*: continuous functions
  - *oracle*: 0-1-2 order black box
  - *features*: no guarantee

- **nonlinear optimization**
  - *goal*: find a local minimum (not always acceptable)
  - *problem class*: differentiable functions
  - *oracle*: 1-2 order black box
  - *features*: variety of approaches, widespread software

- **convex optimization**
  - *goal*: find a global minimum
  - *problem class*: convex sets, convex functions (sometimes restrictive)
  - *oracle*: 1-2 order black box, and beyond
  - *features*: efficient practical methods, complete complexity theory
Complexities for convex optimization

minimize \( x \in Q \) \( f(x) \), where \( Q \subseteq \mathbb{R}^n \) is bounded, closed and convex

<table>
<thead>
<tr>
<th>problem class</th>
<th>lower bound</th>
<th>optimal methods?</th>
</tr>
</thead>
<tbody>
<tr>
<td>nonsmooth</td>
<td>( O \left( \frac{1}{\epsilon^2} \right) )</td>
<td>yes</td>
</tr>
<tr>
<td>smooth</td>
<td>( O \left( \frac{1}{\sqrt{\epsilon}} \right) )</td>
<td>yes</td>
</tr>
<tr>
<td>smooth and strongly convex</td>
<td>( O \left( \log(\frac{1}{\epsilon}) \right) )</td>
<td>yes</td>
</tr>
</tbody>
</table>

- based on **local black-box** first-order oracle
- independent of dimension (good for high-dimensional problems)

**big O notation:** \( a(\epsilon) = O(b(\epsilon)) \) means there exists \( M > 0 \) such that \( a(\epsilon) \leq M b(\epsilon) \) for all \( \epsilon \) sufficiently small
A conceptual contradiction

convexity is a global structure

• usually checked by inspection: e.g., composition of basic convex functions
• numerical verification of convexity is extremely difficult

but numerical methods use local black-box

beyond block box: structural convex optimization

• exploiting structure to improve performance of numerical methods
• recent developments:
  – interior-point methods (2nd-order oracle)
  – smoothing
  – minimization of composite objective
Minimization of composite objective

problem class:

\[
\minimize_{x \in \mathbb{R}^n} \left\{ \phi(x) \triangleq f(x) + \Psi(x) \right\}
\]

- \( f \) is convex and smooth (having Lipschitz-continuous gradient)
- \( \Psi \) is convex, but may be nondifferentiable
- using black-box first-order oracle, complexity is \( O(1/\epsilon^2) \)

structural convex optimization

- assume \( \Psi \) is simple, e.g., can solve explicitly the auxiliary problem

\[
\minimize_{x \in \text{dom} \Psi} \left\{ f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2} \|x - y\|^2 + \Psi(x) \right\}
\]

- accelerated gradient methods achieve reduced complexity \( O(1/\sqrt{\epsilon}) \)
Example: sparse least-squares

\[
\minimize_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1 \quad \text{(where } A \in \mathbb{R}^{m \times n})
\]

- important applications in signal processing, statistics, machine learning
- focus on problem class: \( m < n \) and \( x^* \) sparse (compressed sensing)

complexities of structural convex optimization

<table>
<thead>
<tr>
<th>numerical method</th>
<th>analytical complexity</th>
<th>oracle complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>subgradient method</td>
<td>( O(1/\epsilon^2) )</td>
<td>( O(mn) )</td>
</tr>
<tr>
<td>proximal gradient method</td>
<td>( O(1/\epsilon) )</td>
<td>( O(mn) )</td>
</tr>
<tr>
<td>accelerated gradient method</td>
<td>( O(1/\sqrt{\epsilon}) )</td>
<td>( O(mn) )</td>
</tr>
<tr>
<td>interior-point method</td>
<td>( O(\log(1/\epsilon)) )</td>
<td>( O(m^2n) )</td>
</tr>
<tr>
<td>prox gradient homotopy (under RIP)</td>
<td>( O(\log(1/\epsilon)) )</td>
<td>( O(mn) )</td>
</tr>
</tbody>
</table>
Applications of smoothing

- piecewise-linear approximation

$$\text{minimize}_{x \in \mathbb{R}^n} \max_{i=1,\ldots,m} (a_i^T x + b_i)$$

- one-norm approximation

$$\text{minimize}_x \|x\|_1 \text{ subject to } \|Ax - b\|_2 \leq \delta$$

- group regularization:

$$\text{minimize}_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1 + \rho \sum_{g \in G} w_g \|x_g\|_2$$
Example: low-rank matrix recovery

find a low-rank matrix given noisy linear constraints

$$\minimize_{X \in \mathbb{R}^{m \times n}} \frac{1}{2} \| A(X) - b \|_2^2 + \lambda \| X \|_*$$

where $A : \mathbb{R}^{m \times n} \to \mathbb{R}^p$ is a linear map, $b \in \mathbb{R}^p$. $\| X \|_* = \sum_i \sigma_i(X)$ is the nuclear norm or trace norm, sum of singular values

- special case: when $X = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}$, $\text{rank } X = \# \text{ of nonzero } x_i$

  reduces to sparse least squares, $\| X \|_*$ reduces to $\ell_1$ norm

- many applications in machine learning, controls, signal processing, e.g.,
  matrix completion problem (recommender systems, e.g. Netflix)

- more later...
Course goals and course work

- optimization algorithms along with their complexity analysis
- experience with implementations and applications
- methodologies of structural convex optimization
- exposure to research frontiers in convex optimization and applications

course work

- lectures focus on algorithms and complexity analysis
- 3 homeworks (lag implementation & theory)
- substantial project
Syllabus

tentative:

• **smooth optimization**: gradient method, quasi-Newton methods, Nesterov’s optimal methods

• **nonsmooth optimization**: subgradient calculus, subgradient methods

• **accelerated gradient methods**: proximal mapping, accelerated proximal gradient methods, smoothing

• **decomposition and coordinate descent**: dual decomposition, alternating direction multiplier method, randomized coordinate descent

• **stochastic and online optimization**: convergence and regret analysis, applications in large-scale machine learning

• **interior-point methods**: self-concordant barriers, path-following methods, efficient implementations
On the role of complexity analysis

complexity analysis plays an important role in convex optimization

- many ideas appeared early, but did not result in significant impact due to lack of convincing complexity analysis

<table>
<thead>
<tr>
<th>modern algorithms</th>
<th>early prototypes</th>
</tr>
</thead>
<tbody>
<tr>
<td>accelerated gradient methods</td>
<td>heavy ball method</td>
</tr>
<tr>
<td>polynomial-time IPMs</td>
<td>classical barrier methods</td>
</tr>
<tr>
<td>smoothing</td>
<td>smoothing</td>
</tr>
</tbody>
</table>

- quote from Yurii Nesterov (in 2004 book)

  ... more and more common that the new methods were provided with a complexity analysis, which is considered a better justification of their efficiency than computational experiments ...
References


(The global optimization example with Lipschitz continuous function in the Euclidean norm is from Nesterov’s lecture notes for INMA2460: Nonlinear Optimization, Catholic University of Louvain)