Mathematical Programming II	**Lecture 12**
ORIE 6310 Spring 2014	**March 6, 2014**
Scribe: Nanjing Jian	

We look at problem instances of the form (f, G), where f is the objective to be minimized and G is the feasible region. For now, let's define f on $C := [-1, 1]^n$ and $G = C$, so that the feasible region is the entire domain of f.

Assumption 1 *Assume f is continuously differentiable and satisfies*

$$\max_{x \in C} f(x) - \min_{x \in C} f(x) \leq 1. \tag{1}$$

We want to find $z \in C$ such that

$$\epsilon(z, f) := f(z) - \min_{x \in C} f(x) \leq \epsilon$$

for some $\epsilon \in (0, 1)$. For ease of notation, define $\min f(C) := \min_{x \in C} f(x)$ and $\max f(C) := \max_{x \in C} f(x)$.

The "access" to f is via an oracle: given $x \in C$, the oracle returns $y = (f(x), \nabla f(x))$. Any algorithm asks a series of questions $x_1, x_2, \ldots, x_k$ of the oracle, and gets answers $y_1, y_2, \ldots, y_k$, where each x_j is a function of $(x_1, y_1), (x_2, y_2), \ldots, (x_{j-1}, y_{j-1})$.

x_{k+1} can be a point in C and we then ask the oracle for information at x_{k+1}; otherwise, x_{k+1} = "STOP", and then the algorithm also provides a solution $z \in C$ that is a function of $(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)$. If x_{k+1} = "STOP", we say the algorithm takes $k+1$ steps, for the $k+1$ points $(x_1, x_2, \ldots, x_k, z)$.

Definition 1 *Given a class of functions $\mathscr{F}$, define $\mathcal{N}_{\mathscr{F}}(\epsilon) = \min\{N :$ there is some N step algorithm that always gives z with $\epsilon(z, f) \leq \epsilon$ for all $f \in \mathscr{F}\}$.*

In today's lecture, we will show that:

1. If we take for $\mathscr{F}$ all continuously differentiable functions on C satisfying (1) and (2):

$$||\nabla f(x)||_2 \leq 1 \quad \forall x \in C \tag{2}$$

   then an algorithm might need an *exponential* number of steps. (We need a condition like (2) to exclude a function with a very narrow hole of depth 1 dug at a point the algorithm hasn't looked at.)

2. If we add *convexity* to $\mathscr{F}$, but ask for

$$\hat{\epsilon}(z, f) := \min\{||z - x^*||_2 : x^* \text{ minimizes } f \text{ over } C\} \leq \epsilon,$$

   then such an algorithm is *impossible*!

Theorem 1 *Take $\mathscr{F}$ as above satisfying (1) and (2); then* $\mathcal{N}_{\mathscr{F}}(\epsilon) \geq 2^{-n}(\frac{1}{\epsilon})^n$.

Proof: Here is a sketch of the proof.
First, we produce a piecewise C^1 function f for the case $\frac{1}{\epsilon} \in \mathbb{Z}$. Suppose there were such an algorithm requiring fewer than $(\frac{1}{\epsilon})^n$ steps. Thus the steps are at most $(\frac{1}{\epsilon})^n - 2$ questions x_k and then z.
Then if the oracle always answers $(f(x), \nabla f(x)) = (0, 0)$, the algorithm will generate fewer than $(\frac{1}{\epsilon})^n$ x_k's and z.
Divide C into $(\frac{1}{\epsilon})^n$ small cubes, each of side $\frac{2}{\epsilon}$. Then the oracle misses the interior of one of these smaller cubes (shaded in blue), where we can make the function (zero otherwise) an inverted pyramid with minimum $-\epsilon$.

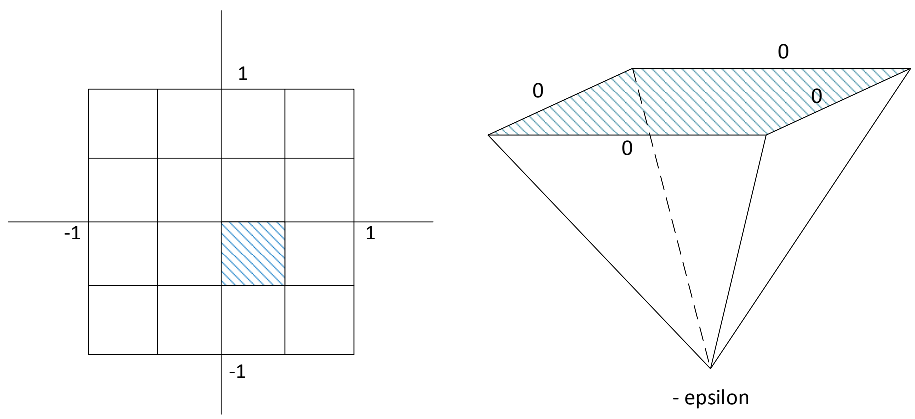

Figure 1: An example when $n = 2, \epsilon = 4$.

Then $f(z) = 0$ but $\min f(C) = -\epsilon$.
Now we need to fix three things:

1. We need error $> \epsilon$.
2. We need to handle $\frac{1}{\epsilon} \notin \mathbb{Z}$.
3. We need to make f a C^1 function.

First, in general, if $\epsilon \geq \frac{1}{2}$, the bound is at most 1 and there is nothing to do. If $\epsilon < \frac{1}{2}$, we find k with $2^{-k} > \epsilon \geq 2^{-k-1}$ and then divide the cube into 2^{kn} smaller cubes of side $2 \cdot 2^{-k}$. Note that $2^{kn} \geq 2^{-n} \cdot 2^{(k+1)n} \geq 2^{-n}(\frac{1}{\epsilon})^n$. So we can construct f as before with an inverted pyramid of depth $2^{-k} > \epsilon$, and then "smooth" the "pyramid" region as follows:

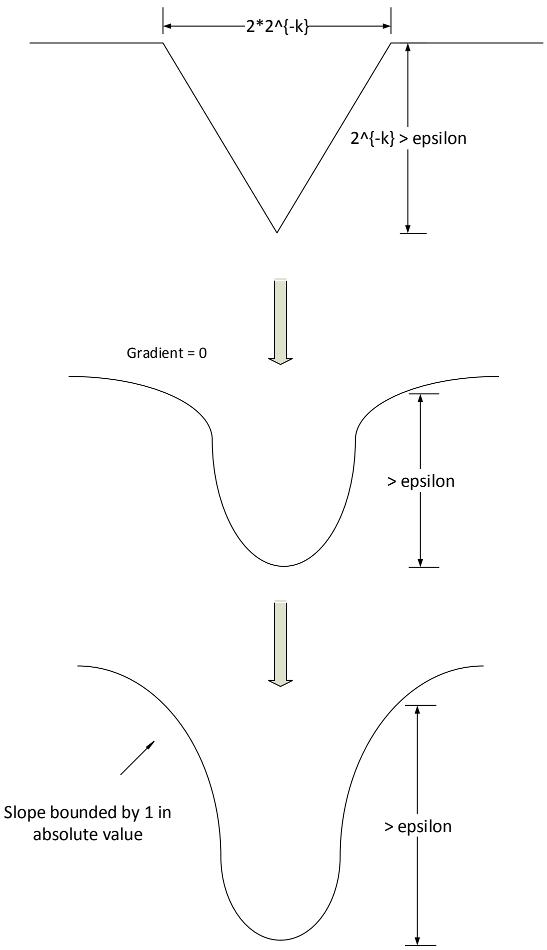

Figure 2: The "smoothing" of a one-dimensional function.

This can be extended into $\mathbb{R}^n$ similarly. $\square$

Note that the function we constructed is quasi-convex; also, we could smooth it to be C^k not just C^1. In fact, we get similar exponential behavior if we

- require f to be C^k for any k;
- or require f quasi-convex (which means all its level sets are convex);
- or allow stochastic algorithms.

Remark 1 *General references are [1] and [2].*

We will next show that if we ask for $\hat{\epsilon}(z, f) \leq \epsilon$, even with convexity, the task is impossible. Here we need to assume the (plausible but true) **fact** that for $n = 1$, if $\mathscr{F}$ is the set of convex functions satisfying (1) and (2), then $\mathcal{N}_{\mathscr{F}}(\epsilon) \to \infty$ as $\epsilon \downarrow 0$, i.e., for any N, there is some $\epsilon = \epsilon(N) > 0$ such that $\mathcal{N}_{\mathscr{F}}(\epsilon) > N$.

Theorem 2 *Let $\mathscr{F}$ be the set of convex C^1 functions on C satisfying (1) and (2). Then no algorithm can ensure $\hat{\epsilon}(z, f) < 1$ in any fixed finite number of steps for $n \geq 2$.*

Proof: Take $n = 2$ and for any C^1 function $f_1 : [-1, 1] \to \mathbb{R}$ satisfying (1), (2), define $f_2 : [-1, 1]^2 \to \mathbb{R}$ by $f_2(x^{(1)}, x^{(2)}) = f_1(x^{(1)})$ (i.e., f_2 is flat on the second dimension), where the superscripts here are components.
Suppose there is an algorithm for locating an approximate *minimizer* with $\hat{\epsilon}(z, f) < 1$ that takes N steps.
Use this algorithm to define another, $N+1$ step, algorithm to find an approximate *minimum* of a function f_1 by applying the first algorithm to the corresponding f_2 to get $x_1, x_2, \ldots, x_{N-1}, z$ and then call the oracle again at z. Stop and output $w^{(1)}$, where
$w = \arg\min\{f_2(x_1), f_2(x_2), \ldots, f_2(x_{N-1}), f_2(z)\}$.
By the fact above, there is some $\epsilon = \epsilon(N + 1) > 0$ so that $\mathcal{N}_{\mathscr{F}}(\epsilon) > N + 1$.
So there is some f_1 for which the algorithm does not give an ϵ-approximate minimum.
Hence, $f_2(x_j), 1 \leq j \leq N - 1$ and $f_2(z)$ are all greater than ϵ above $\min f_2(C)$.

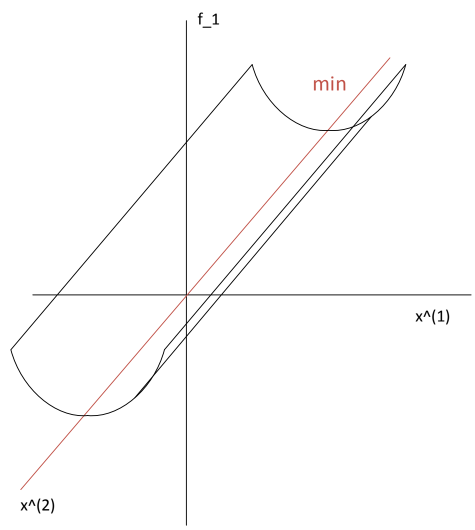

Figure 3: An example of f_2 with minimizers along $x^{(2)}$.

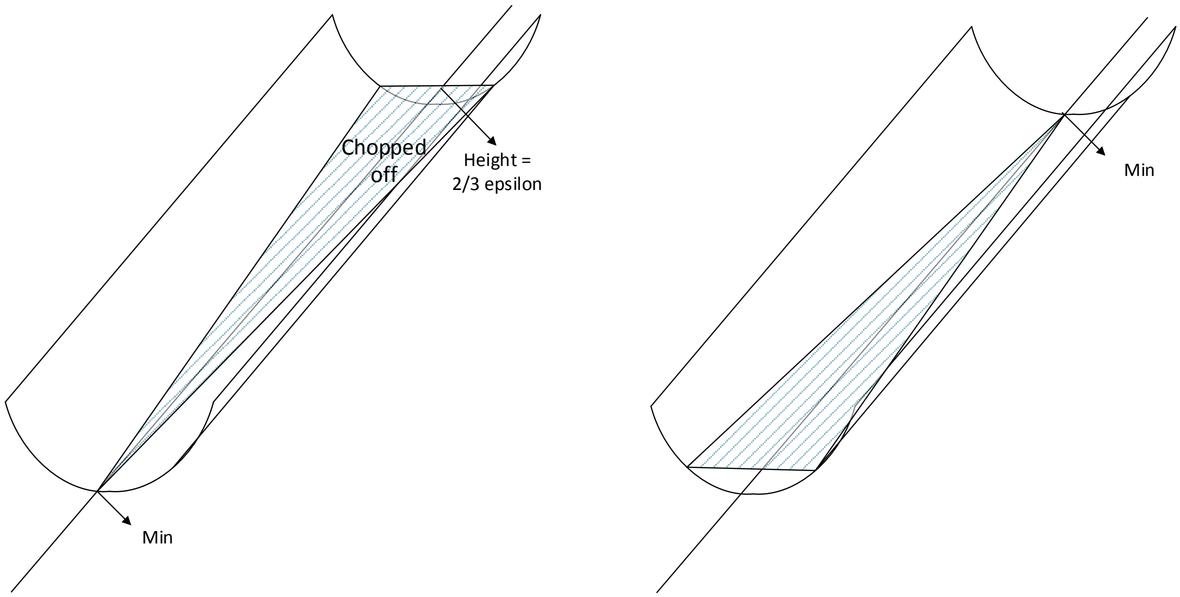

Figure 4: Corresponding possible counterexamples $\hat{f}_2$.

So, if $z^{(2)} \geq 0$ (the right-hand graph above), let

$$\hat{f}_2(x) = \max\{f_2(x), \min f_2(C) + \frac{\epsilon}{3}(x^{(2)} + 1))\} \tag{3}$$

Since it never sees any point for which the functions differ, the algorithm generates exactly the same iterates for $\hat{f}_2$ as for f_2, so the same z, and the only minimizers of $\hat{f}_2$ have $x^{(2)} = -1$, so $\hat{\epsilon} \geq 1$. Similarly, using the left-hand graph, if $z^{(2)} \leq 0$.
Then we smooth $\hat{f}_2$ at the boundaries of the chopped off region to make $\hat{f}_2$ become C^1. □

References

[1] A.S. Nemirovski and D.B. Yudin. *Problem complexity and method efficiency in optimization.* Wiley-Interscience series in discrete mathematics. Wiley, 1983.

[2] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course.* Applied Optimization. Springer, 2004.