Structural Optimization: New perspectives for increasing efficiency of numerical schemes

Yurii Nesterov, CORE/INMA (UCL)

May 14, 2015 (Moscow)

Joint results with S. Stich (ETH Zurich)

Optimization and Applications in Control and Data Science

On the occasion of Boris Polyak's 80th birthday

Thirty years ago ...

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x),$

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Thirty years ago ...

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984):

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$,

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result:

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$.

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Thirty years ago ...

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare:

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare: Heavy ball method

Thirty years ago ...

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare: Heavy ball method (B.Polyak, 1964)

Thirty years ago ...

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare: Heavy ball method (B.Polyak, 1964)

$x_{k+1} = x_k + \alpha_k(x_k - x_{k-1}) - \beta_k \nabla f(x_k)$, $k \geq 0$.

Thirty years ago ...

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare: Heavy ball method (B.Polyak, 1964)

$x_{k+1} = x_k + \alpha_k(x_k - x_{k-1}) - \beta_k \nabla f(x_k)$, $k \geq 0$.
(Convergence analysis for QP)

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare: Heavy ball method (B.Polyak, 1964)

$x_{k+1} = x_k + \alpha_k(x_k - x_{k-1}) - \beta_k \nabla f(x_k)$, $k \geq 0$.
(Convergence analysis for QP)

Applications (1983-2003):

Thirty years ago ...

Problem: $\min_{x \in \mathbb{R}^n} f(x)$, f is convex and

$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, $x, y \in \mathbb{R}^n$.

Fast gradient method (N.1984): $x_0 \in \mathbb{R}^n$,

$y_k = x_k + \frac{k}{k+2}(x_k - x_{k-1})$, $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$. (Optimal)

Compare: Heavy ball method (B.Polyak, 1964)

$x_{k+1} = x_k + \alpha_k(x_k - x_{k-1}) - \beta_k \nabla f(x_k)$, $k \geq 0$.
(Convergence analysis for QP)

Applications (1983-2003): Empty set ...

Twenty years later ...

Twenty years later ...

Problem: $\min\limits_{x \in Q} f(x)$,

Twenty years later ...

Problem: $\min\limits_{x \in Q} f(x)$, f is convex,

Twenty years later ...

Problem: $\min\limits_{x \in Q} f(x)$, f is convex, Q is convex,

Twenty years later ...

Problem: $\min\limits_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function:

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03):

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

Twenty years later ...

Problem: $\min\limits_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

$$y_k = \tfrac{k}{k+3} v_k + \tfrac{2}{k+3} x_k,$$

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

$$y_k = \tfrac{k}{k+3} v_k + \tfrac{2}{k+3} x_k, \; x_{k+1} = y_k - \tfrac{1}{L} \nabla f(y_k), \; k \geq 0.$$

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

$$y_k = \tfrac{k}{k+3} v_k + \tfrac{2}{k+3} x_k, \; x_{k+1} = y_k - \tfrac{1}{L} \nabla f(y_k), \; k \geq 0.$$

Result:

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

$y_k = \frac{k}{k+3} v_k + \frac{2}{k+3} x_k$, $x_{k+1} = y_k - \frac{1}{L} \nabla f(y_k)$, $k \geq 0$.

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$.

Twenty years later ...

Problem: $\min_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

$$y_k = \tfrac{k}{k+3} v_k + \tfrac{2}{k+3} x_k, \; x_{k+1} = y_k - \tfrac{1}{L} \nabla f(y_k), \; k \geq 0.$$

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$.

Applications (2003 - 2015(?)):

Twenty years later ...

Problem: $\min\limits_{x \in Q} f(x)$, f is convex, Q is convex, and

$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$, $x, y \in Q$.

Prox-function: strongly convex $d(x)$, $x \in Q$.

Fast gradient method (N.03): $v_0 = x_0 \in \mathbb{R}^n$,

$$v_k = \arg\min_{x \in Q} \left\{ d(x) + \sum_{i=0}^{k-1} \tfrac{i+1}{2L} [f(y_k) + \langle \nabla f(y_i), x - y_i \rangle] \right\},$$

$$y_k = \tfrac{k}{k+3} v_k + \tfrac{2}{k+3} x_k, \; x_{k+1} = y_k - \tfrac{1}{L} \nabla f(y_k), \; k \geq 0.$$

Result: $f(x_k) - f^* \leq \frac{2LR^2}{k(k+1)}$, $k \geq 1$.

Applications (2003 - 2015(?)): Smooth approximations of nonsmooth functions.

Smoothing technique

Smoothing technique

Problem:

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u \in U} \{ \langle Ax, u \rangle - \phi(u) \}$.

Smoothing technique

Problem: $\min_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max_{u \in U} \{ \langle Ax, u \rangle - \phi(u) \}$.

Smoothing: $f_\mu(x) = \max_{u \in U} \{ \langle Ax, u \rangle - \phi(u) - \mu d_2(u) \}$,

Smoothing technique

Problem: $\min\limits_{x\in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u\in U}\{\langle Ax, u\rangle - \phi(u)\}$.

Smoothing: $f_\mu(x) = \max\limits_{u\in U}\{\langle Ax, u\rangle - \phi(u) - \mu d_2(u)\}$,
where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Smoothing technique

Problem: $\min_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max_{u \in U} \{ \langle Ax, u \rangle - \phi(u) \}$.

Smoothing: $f_\mu(x) = \max_{u \in U} \{ \langle Ax, u \rangle - \phi(u) - \mu d_2(u) \}$,
where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\| \nabla f_\mu(x) - \nabla f_\mu(y) \|_* \le \frac{1}{\mu} \|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u \in U} \{ \langle Ax, u \rangle - \phi(u) \}$.

Smoothing: $f_\mu(x) = \max\limits_{u \in U} \{ \langle Ax, u \rangle - \phi(u) - \mu d_2(u) \}$,

where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\| \nabla f_\mu(x) - \nabla f_\mu(y) \|_* \le \frac{1}{\mu} \|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,

where $\|A\| = \max\limits_{x,u} \{ \langle Ax, u \rangle : \ \|x\| \le 1, \ \|u\| \le 1 \}$.

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u \in U} \{\langle Ax, u \rangle - \phi(u)\}$.

Smoothing: $f_\mu(x) = \max\limits_{u \in U} \{\langle Ax, u \rangle - \phi(u) - \mu d_2(u)\}$,

where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\|\nabla f_\mu(x) - \nabla f_\mu(y)\|_* \leq \frac{1}{\mu} \|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,

where $\|A\| = \max\limits_{x,u} \{\langle Ax, u \rangle : \ \|x\| \leq 1, \ \|u\| \leq 1\}$.

Complexity:

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u \in U} \{\langle Ax, u\rangle - \phi(u)\}$.

Smoothing: $f_\mu(x) = \max\limits_{u \in U} \{\langle Ax, u\rangle - \phi(u) - \mu d_2(u)\}$,
where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\|\nabla f_\mu(x) - \nabla f_\mu(y)\|_* \leq \frac{1}{\mu}\|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,
where $\|A\| = \max\limits_{x,u} \{\langle Ax, u\rangle : \ \|x\| \leq 1, \ \|u\| \leq 1\}$.

Complexity: Choose $\mu = O(\epsilon)$.

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u \in U} \{\langle Ax, u\rangle - \phi(u)\}$.

Smoothing: $f_\mu(x) = \max\limits_{u \in U} \{\langle Ax, u\rangle - \phi(u) - \mu d_2(u)\}$,

where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\|\nabla f_\mu(x) - \nabla f_\mu(y)\|_* \le \frac{1}{\mu}\|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,

where $\|A\| = \max\limits_{x,u} \{\langle Ax, u\rangle : \ \|x\| \le 1, \ \|u\| \le 1\}$.

Complexity: Choose $\mu = O(\epsilon)$.

Then we get ϵ-solution in $O(\frac{1}{\epsilon}\|A\| D_1 D_2)$ iterations.

Smoothing technique

Problem: $\min_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max_{u \in U} \{ \langle Ax, u \rangle - \phi(u) \}$.

Smoothing: $f_\mu(x) = \max_{u \in U} \{ \langle Ax, u \rangle - \phi(u) - \mu d_2(u) \}$,

where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\| \nabla f_\mu(x) - \nabla f_\mu(y) \|_* \leq \frac{1}{\mu} \|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,

where $\|A\| = \max_{x,u} \{ \langle Ax, u \rangle : \ \|x\| \leq 1, \ \|u\| \leq 1 \}$.

Complexity: Choose $\mu = O(\epsilon)$.

Then we get ϵ-solution in $O(\frac{1}{\epsilon} \|A\| D_1 D_2)$ iterations.

Compare:

Smoothing technique

Problem: $\min\limits_{x \in Q} f(x)$, $\operatorname{diam} Q = D_1$.

Model: $f(x) = \max\limits_{u \in U} \{ \langle Ax, u \rangle - \phi(u) \}$.

Smoothing: $f_\mu(x) = \max\limits_{u \in U} \{ \langle Ax, u \rangle - \phi(u) - \mu d_2(u) \}$,

where d_2 is strongly convex on U, $\operatorname{diam} U = D_2$.

Then $\| \nabla f_\mu(x) - \nabla f_\mu(y) \|_* \le \frac{1}{\mu} \|A\|^2 \cdot \|x - y\|$, $x, y \in \mathbb{R}^n$, $\mu > 0$,

where $\|A\| = \max\limits_{x,u} \{ \langle Ax, u \rangle : \ \|x\| \le 1, \ \|u\| \le 1 \}$.

Complexity: Choose $\mu = O(\epsilon)$.

Then we get ϵ-solution in $O(\frac{1}{\epsilon} \|A\| D_1 D_2)$ iterations.

Compare: Subgradient schemes need $O(\frac{1}{\epsilon^2} \|A\|^2 D_1^2 D_2^2)$ iter.

Ten years later ...

Ten years later ...

Huge-scale problems

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Ten years later ...

Huge-scale problems $\quad\Rightarrow\quad$ *Coordinate descent methods*

Problem: $\quad \min\limits_{x \in \mathbb{R}^n} f(x),$

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n.$

Ten years later ...

Huge-scale problems $\quad\Rightarrow\quad$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n.$ (**NB:** $L_i \leq L_{\nabla f}$.)

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence:

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + h e_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2, \; i = 1 : n.$

Ten years later ...

Huge-scale problems $\quad \Rightarrow \quad$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n.$ (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2$, $i = 1 : n$.

Consequences [N.12]:

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n.$ (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2, \; i = 1 : n.$

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^{n} L_i^\alpha, \; \alpha \in [0, 1]$.

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + h e_i) - \nabla_i f(x)| \leq L_i |h|, \ \forall x, h \in \mathbb{R}^n.$ (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2, \ i = 1 : n.$

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^{n} L_i^\alpha, \ \alpha \in [0, 1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$.

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n.$ (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2, \; i = 1 : n.$

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^{n} L_i^\alpha, \; \alpha \in [0, 1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$. Then
  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2S_1} \|\nabla f(x)\|^2_{[0]}$

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i}\nabla_i f(x) e_i\right) \geq \frac{1}{2L_i}\left(\nabla_i f(x)\right)^2, \; i = 1:n.$

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^{n} L_i^\alpha$, $\alpha \in [0,1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$. Then
  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2S_1}\|\nabla f(x)\|^2_{[0]} \Rightarrow \mathcal{E}\left(f(x_k)\right) \leq \frac{S_1 R_0^2}{k}$.

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|$, $\forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2$, $i = 1 : n$.

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^n L_i^\alpha$, $\alpha \in [0, 1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$. Then
  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2S_1} \|\nabla f(x)\|^2_{[0]} \Rightarrow \mathcal{E}\left(f(x_k)\right) \leq \frac{S_1 R_0^2}{k}$.
  $\left(\|x\|^2_{[\alpha]} = \sum\limits_{i=1}^n L_i^\alpha \left(x^{(i)}\right)^2, \quad \|g\|^2_{[\alpha]} = \sum\limits_{i=1}^n L_i^{-\alpha} \left(g^{(i)}\right)^2 . \right)$

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i}\nabla_i f(x) e_i\right) \geq \frac{1}{2L_i}\left(\nabla_i f(x)\right)^2, \; i = 1 : n.$

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^{n} L_i^\alpha, \; \alpha \in [0,1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$. Then
  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2S_1}\|\nabla f(x)\|^2_{[0]} \;\Rightarrow\; \mathcal{E}\left(f(x_k)\right) \leq \frac{S_1 R_0^2}{k}$.
  $\left(\|x\|^2_{[\alpha]} = \sum\limits_{i=1}^{n} L_i^\alpha \left(x^{(i)}\right)^2, \quad \|g\|^2_{[\alpha]} = \sum\limits_{i=1}^{n} L_i^{-\alpha}\left(g^{(i)}\right)^2. \right)$
- Choose i with probability $\pi_i = \frac{1}{n}$.

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \; \forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i} \nabla_i f(x) e_i\right) \geq \frac{1}{2L_i} \left(\nabla_i f(x)\right)^2, \; i = 1 : n.$

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^n L_i^\alpha, \; \alpha \in [0,1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$. Then

  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2S_1} \|\nabla f(x)\|^2_{[0]} \;\Rightarrow\; \mathcal{E}\left(f(x_k)\right) \leq \frac{S_1 R_0^2}{k}$.

  $\left(\|x\|^2_{[\alpha]} = \sum\limits_{i=1}^n L_i^\alpha \left(x^{(i)}\right)^2, \quad \|g\|^2_{[\alpha]} = \sum\limits_{i=1}^n L_i^{-\alpha} \left(g^{(i)}\right)^2. \right)$

- Choose i with probability $\pi_i = \frac{1}{n}$. Then

  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2n} \|\nabla f(x)\|^2_{[1]}$

Ten years later ...

Huge-scale problems $\Rightarrow$ *Coordinate descent methods*

Problem: $\min\limits_{x \in \mathbb{R}^n} f(x)$, with objective satisfying conditions

$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq L_i |h|, \forall x, h \in \mathbb{R}^n$. (**NB:** $L_i \leq L_{\nabla f}$.)

Hence: $f(x) - f\left(x - \frac{1}{L_i}\nabla_i f(x) e_i\right) \geq \frac{1}{2L_i}\left(\nabla_i f(x)\right)^2$, $i = 1 : n$.

Consequences [N.12]: Denote $S_\alpha = \sum\limits_{i=1}^n L_i^\alpha$, $\alpha \in [0,1]$.

- Choose i with probability $\pi_i = \frac{1}{S_1} L_i$. Then

  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2S_1}\|\nabla f(x)\|^2_{[0]} \Rightarrow \mathcal{E}\left(f(x_k)\right) \leq \frac{S_1 R_0^2}{k}$.

  $$\left(\|x\|^2_{[\alpha]} = \sum_{i=1}^n L_i^\alpha \left(x^{(i)}\right)^2, \quad \|g\|^2_{[\alpha]} = \sum_{i=1}^n L_i^{-\alpha} \left(g^{(i)}\right)^2. \right)$$

- Choose i with probability $\pi_i = \frac{1}{n}$. Then

  $\mathcal{E}\left(f(x_+)\right) \leq f(x) - \frac{1}{2n}\|\nabla f(x)\|^2_{[1]} \Rightarrow \mathcal{E}\left(f(x_k)\right) \leq \frac{nR_1^2}{k}$.

Fast Coordinate Descent

Fast Coordinate Descent

Random generator:

Random generator: For $\beta \in [0, 1]$, get $j = \mathcal{R}_\beta(L) \in \{1 : n\}$

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \overset{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \le 2(\frac{n}{k+1})^2 L_{\max} R^2_{[0]}$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \le 2\frac{nS_1}{(k+1)^2} R^2_{[0]}$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \le 2\left(\frac{S_{1/2}}{k+1}\right)^2 R^2_{[0]}$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \leq 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD:

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \le 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \le 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \le 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \equiv \text{Prob}(j=i) \stackrel{\text{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \leq 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$

1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \le 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \le 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \le 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$
1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.
2) Solve $a_{t+1}^2 S_{1/2}^2 = A_t + a_{t+1}$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \overset{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \leq 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$

1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.

2) Solve $a_{t+1}^2 S_{1/2}^2 = A_t + a_{t+1}$. Set $A_{t+1} = A_t + a_{t+1}$, $\alpha_t = \frac{a_{t+1}}{A_{t+1}}$.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \leq 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$

1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.

2) Solve $a_{t+1}^2 S_{1/2}^2 = A_t + a_{t+1}$. Set $A_{t+1} = A_t + a_{t+1}$, $\alpha_t = \frac{a_{t+1}}{A_{t+1}}$.

3) Set $y_t = (1-\alpha_t)x_t + \alpha_t v_t$,

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \leq 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$

1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.

2) Solve $a_{t+1}^2 S_{1/2}^2 = A_t + a_{t+1}$. Set $A_{t+1} = A_t + a_{t+1}$, $\alpha_t = \frac{a_{t+1}}{A_{t+1}}$.

3) Set $y_t = (1-\alpha_t)x_t + \alpha_t v_t$, $x_{t+1} = y_t - \frac{1}{L_{i_t}} \nabla_{i_t} f(y_t) e_{i_t}$,

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \equiv \text{Prob}(j=i) \overset{\text{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \le 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \le 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \le 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$

1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.

2) Solve $a_{t+1}^2 S_{1/2}^2 = A_t + a_{t+1}$. Set $A_{t+1} = A_t + a_{t+1}$, $\alpha_t = \frac{a_{t+1}}{A_{t+1}}$.

3) Set $y_t = (1-\alpha_t)x_t + \alpha_t v_t$, $x_{t+1} = y_t - \frac{1}{L_{i_t}} \nabla_{i_t} f(y_t) e_{i_t}$,

$$v_{t+1} = v_t - \frac{a_{t+1} S_{1/2}}{L_{i_t}^{1/2}} \nabla_{i_t} f(y_t) e_{i_t}.$$

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \le 2(\frac{n}{k+1})^2 L_{\max} R^2_{[0]}$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \le 2\frac{nS_1}{(k+1)^2} R^2_{[0]}$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \le 2\left(\frac{S_{1/2}}{k+1}\right)^2 R^2_{[0]}$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$
1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.
2) Solve $a^2_{t+1} S^2_{1/2} = A_t + a_{t+1}$. Set $A_{t+1} = A_t + a_{t+1}$, $\alpha_t = \frac{a_{t+1}}{A_{t+1}}$.
3) Set $y_t = (1-\alpha_t)x_t + \alpha_t v_t$, $x_{t+1} = y_t - \frac{1}{L_{i_t}} \nabla_{i_t} f(y_t) e_{i_t}$,

$$v_{t+1} = v_t - \frac{a_{t+1} S_{1/2}}{L_{i_t}^{1/2}} \nabla_{i_t} f(y_t) e_{i_t}.$$

NB: Each iteration needs $O(n)$ a.o.

Fast Coordinate Descent

Random generator: For $\beta \in [0,1]$, get $j = \mathcal{R}_\beta(L) \in \{1:n\}$ with probabilities $\pi_\beta[i] \;\equiv\; \mathrm{Prob}(j=i) \stackrel{\mathrm{def}}{=} \frac{1}{S_\beta} L_i^\beta, \quad i \in \{1:n\}$.

- [N.12] $\beta = 0$: $\mathcal{E}(f(x_k)) - f^* \leq 2(\frac{n}{k+1})^2 L_{\max} R_{[0]}^2$.
- [Lee, Sidford 13] $\beta = 1$: $\mathcal{E}(f(x_k)) - f^* \leq 2\frac{nS_1}{(k+1)^2} R_{[0]}^2$.
- [N., Stich 15] $\beta = \frac{1}{2}$: $\mathcal{E}(f(x_k)) - f^* \leq 2\left(\frac{S_{1/2}}{k+1}\right)^2 R_{[0]}^2$.

Fast CD: Choose $v_0 = x_0 \in \mathbb{R}^n$. Set $A_0 = 0$
1) Choose active coordinate $i_t = \mathcal{R}_{1/2}(L)$.
2) Solve $a_{t+1}^2 S_{1/2}^2 = A_t + a_{t+1}$. Set $A_{t+1} = A_t + a_{t+1}$, $\alpha_t = \frac{a_{t+1}}{A_{t+1}}$.
3) Set $y_t = (1-\alpha_t)x_t + \alpha_t v_t$, $x_{t+1} = y_t - \frac{1}{L_{i_t}} \nabla_{i_t} f(y_t) e_{i_t}$,

$$v_{t+1} = v_t - \frac{a_{t+1} S_{1/2}}{L_{i_t}^{1/2}} \nabla_{i_t} f(y_t) e_{i_t}.$$

NB: Each iteration needs $O(n)$ a.o. $\Rightarrow$ Not for Huge Scale.

Complexity

Complexity

For getting ϵ-accuracy we need

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2}R_{[0]}}{\epsilon^{1/2}}$ iterations

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2}R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{nL_{\nabla f}^{1/2}R_{[0]}}{\epsilon^{1/2}}$.

Main question:

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB:

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often unbalanced.

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often <u>unbalanced</u>.

Model: $f(x) = F(Ax, x)$,

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often unbalanced.

Model: $f(x) = F(Ax, x)$, where $F(s,x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often unbalanced.

Model: $f(x) = F(Ax, x)$, where $F(s, x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Main assumption:

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often <u>unbalanced</u>.

Model: $f(x) = F(Ax, x)$, where $F(s,x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Main assumption: $F(s,x)$ can be computed in $O(m+n)$ a.o.

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often <u>unbalanced</u>.

Model: $f(x) = F(Ax, x)$, where $F(s,x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Main assumption: $F(s,x)$ can be computed in $O(m+n)$ a.o.

Consequences:

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often <u>unbalanced</u>.

Model: $f(x) = F(Ax, x)$, where $F(s,x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Main assumption: $F(s,x)$ can be computed in $O(m+n)$ a.o.

Consequences:

- After coordinate move, product Ax can be computed in $O(m)$ a.o.

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often <u>unbalanced</u>.

Model: $f(x) = F(Ax, x)$, where $F(s,x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Main assumption: $F(s,x)$ can be computed in $O(m+n)$ a.o.

Consequences:

- After coordinate move, product Ax can be computed in $O(m)$ a.o.
- If Ax is known, $\nabla_i f(x)$ can be computed in $O(m)$ a.o.

Complexity

For getting ϵ-accuracy we need $\frac{S_{1/2} R_{[0]}}{\epsilon^{1/2}}$ iterations $\leq \frac{n L_{\nabla f}^{1/2} R_{[0]}}{\epsilon^{1/2}}$.

Main question: When CD-oracle is n times cheaper?

NB: For FGM, complexity oracle/method is often unbalanced.

Model: $f(x) = F(Ax, x)$, where $F(s,x) : \mathbb{R}^{m+n} \to \mathbb{R}$.

Main assumption: $F(s,x)$ can be computed in $O(m+n)$ a.o.

Consequences:

- After coordinate move, product Ax can be computed in $O(m)$ a.o.
- If Ax is known, $\nabla_i f(x)$ can be computed in $O(m)$ a.o.
- If Ax and Ay are known, $\alpha Ax + \beta A_y$ can be computed in $O(m)$ a.o.

Example 1: Unconstrained quadratic minimization

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n\times n}$ is dense.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s, x) = \frac{1}{2}\langle s, x \rangle - \langle b, x \rangle$.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s,x) = \frac{1}{2}\langle s, x \rangle - \langle b, x \rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x \rangle - \langle b, x \rangle$.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s,x) = \frac{1}{2}\langle s, x\rangle - \langle b, x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right)$$

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s,x) = \frac{1}{2}\langle s, x\rangle - \langle b, x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2 \lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s,x) = \frac{1}{2}\langle s, x\rangle - \langle b, x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2 \lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

NB: We use inequality $L_i \leq \lambda_{\max}^{1/2}(A)$, $i \in \{1:n\}$.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s,x) = \frac{1}{2}\langle s, x\rangle - \langle b, x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2 \lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

NB: We use inequality $L_i \leq \lambda_{\max}^{1/2}(A)$, $i \in \{1:n\}$.

For some cases it is too weak.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n\times n}$ is <u>dense</u>.

Define $F(s,x) = \frac{1}{2}\langle s,x\rangle - \langle b,x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax,x\rangle - \langle b,x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2 \lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

NB: We use inequality $L_i \leq \lambda_{\max}^{1/2}(A)$, $i \in \{1:n\}$.

For some cases it is too weak.

Example: $0 < \gamma_1 \leq A^{(i,j)} \leq \gamma_2$, $i,j \in \{1:n\}$.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is dense.

Define $F(s,x) = \frac{1}{2}\langle s, x\rangle - \langle b, x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2 \lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

NB: We use inequality $L_i \leq \lambda_{\max}^{1/2}(A)$, $i \in \{1:n\}$.

For some cases it is too weak.

Example: $0 < \gamma_1 \leq A^{(i,j)} \leq \gamma_2$, $i,j \in \{1:n\}$.

Then $S_{1/2} \leq n\gamma_2^{1/2}$

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n \times n}$ is <u>dense</u>.

Define $F(s,x) = \frac{1}{2}\langle s, x\rangle - \langle b, x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2 \lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

NB: We use inequality $L_i \leq \lambda_{\max}^{1/2}(A)$, $i \in \{1:n\}$.

For some cases it is too weak.

Example: $0 < \gamma_1 \leq A^{(i,j)} \leq \gamma_2$, $i,j \in \{1:n\}$.

Then $S_{1/2} \leq n\gamma_2^{1/2}$ and $\lambda_{\max}(A) \;\geq\; \gamma_1 \lambda_{\max}(1_n 1_n^T) \;=\; n\gamma_2$.

Example 1: Unconstrained quadratic minimization

Let $A = A^T \succ 0 \in \mathbb{R}^{n\times n}$ is <u>dense</u>.

Define $F(s,x) = \frac{1}{2}\langle s,x\rangle - \langle b,x\rangle$.

Then $f(x) = \frac{1}{2}\langle Ax,x\rangle - \langle b,x\rangle$. We have

$$T_{CD} = O\left(\frac{nS_{1/2}}{\epsilon^{1/2}} R_{[0]}\right) \quad \leq \quad T_{FGM} = O\left(\frac{n^2\lambda_{\max}^{1/2}(A)}{\epsilon^{1/2}} R_{[0]}\right).$$

NB: We use inequality $L_i \leq \lambda_{\max}^{1/2}(A)$, $i \in \{1:n\}$.

For some cases it is too weak.

Example: $0 < \gamma_1 \leq A^{(i,j)} \leq \gamma_2$, $i,j \in \{1:n\}$.

Then $S_{1/2} \leq n\gamma_2^{1/2}$ and $\lambda_{\max}(A) \geq \gamma_1 \lambda_{\max}(1_n 1_n^T) = n\gamma_2$.

We gain $O(n^{1/2})$ in the total number of operations.

Example 2: Smoothing technique

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,
where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Example 2: Smoothing technique

Consider function $f(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption:

Example 2: Smoothing technique

Consider function $f(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations ($m \geq n$).

Example 2: Smoothing technique

Consider function $f(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{m R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \sum_{i=1}^{n} \| A e_i \| \right)$$

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max\limits_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{m R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \sum_{i=1}^{n} \| A e_i \| \right) \quad \leq \quad T_{FGM} = O\left(\frac{mn R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \| A \| \right).$$

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max\limits_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{m R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \sum_{i=1}^{n} \| A e_i \| \right) \quad \leq \quad T_{FGM} = O\left(\frac{mn R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \| A \| \right).$$

It can be that $\| A e_i \| << \| A \|$, $i \in \{1 : n\}$.

Example 2: Smoothing technique

Consider function $f(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{mR_{[0]}}{\mu^{1/2}\epsilon^{1/2}} \sum_{i=1}^{n} \|Ae_i\| \right) \quad \leq \quad T_{FGM} = O\left(\frac{mnR_{[0]}}{\mu^{1/2}\epsilon^{1/2}} \|A\| \right).$$

It can be that $\|Ae_i\| << \|A\|$, $i \in \{1 : n\}$.

Example: $0 \ < \ \gamma_1 \leq A^{(i,j)} \ \leq \ \gamma_2, \quad i \in \{1 : m\}, \ j \in \{1 : n\}$.

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max\limits_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{m R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \sum_{i=1}^{n} \|Ae_i\| \right) \quad \leq \quad T_{FGM} = O\left(\frac{mn R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \|A\| \right).$$

It can be that $\|Ae_i\| << \|A\|$, $i \in \{1:n\}$.

Example: $0 \; < \; \gamma_1 \leq A^{(i,j)} \; \leq \; \gamma_2, \quad i \in \{1:m\}, \; j \in \{1:n\}$.

Then $\|Ae_i\| \leq \gamma_2 \sqrt{m}$,

Example 2: Smoothing technique

Consider function $f(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max\limits_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max\limits_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{m R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \sum_{i=1}^{n} \|Ae_i\| \right) \quad \leq \quad T_{FGM} = O\left(\frac{mn R_{[0]}}{\mu^{1/2} \epsilon^{1/2}} \|A\| \right).$$

It can be that $\|Ae_i\| << \|A\|$, $i \in \{1 : n\}$.

Example: $0 \ < \ \gamma_1 \leq A^{(i,j)} \ \leq \ \gamma_2, \quad i \in \{1:m\}, \ j \in \{1:n\}$.

Then $\|Ae_i\| \leq \gamma_2 \sqrt{m}$, and $\|A\| \geq \gamma_1 \| 1_m 1_n^T \| = \gamma_1 \sqrt{mn}$.

Example 2: Smoothing technique

Consider function $f(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) \}$,

where $Q \subset \mathbb{R}^m$ is closed convex and bounded.

Define $f_\mu(x) = \max_{u \in Q} \{ \langle Ax, u \rangle - \phi(u) - \mu d(u) \}$, $\mu > 0$.

Main assumption: $F(s) = \max_{u \in Q} \{ \langle s, u \rangle - \phi(u) - \mu d(u) \}$ is computable in $O(m)$ operations $(m \geq n)$. Then

$$T_{CD} = O\left(\frac{mR_{[0]}}{\mu^{1/2}\epsilon^{1/2}} \sum_{i=1}^{n} \|Ae_i\| \right) \quad \leq \quad T_{FGM} = O\left(\frac{mnR_{[0]}}{\mu^{1/2}\epsilon^{1/2}} \|A\| \right).$$

It can be that $\|Ae_i\| << \|A\|$, $i \in \{1:n\}$.

Example: $0 \ < \ \gamma_1 \leq A^{(i,j)} \ \leq \ \gamma_2, \quad i \in \{1:m\}$, $j \in \{1:n\}$.

Then $\|Ae_i\| \leq \gamma_2 \sqrt{m}$, and $\|A\| \geq \gamma_1 \| 1_m 1_n^T \| = \gamma_1 \sqrt{mn}$.

We gain $O(\sqrt{n})$ in complexity.

Conclusion

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

3. We get better results on a problem class, which covers most of applications of Smoothing Technique with dense matrices.

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

3. We get better results on a problem class, which covers most of applications of Smoothing Technique with dense matrices.

4. Our method is oracle/iteration balanced ($O(n + m)$ operations for both).

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

3. We get better results on a problem class, which covers most of applications of Smoothing Technique with dense matrices.

4. Our method is oracle/iteration balanced ($O(n+m)$ operations for both).

But:

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

3. We get better results on a problem class, which covers most of applications of Smoothing Technique with dense matrices.

4. Our method is oracle/iteration balanced ($O(n+m)$ operations for both).

But:

5. Constrained minimization is not covered.

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n}T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

3. We get better results on a problem class, which covers most of applications of Smoothing Technique with dense matrices.

4. Our method is oracle/iteration balanced ($O(n+m)$ operations for both).

But:

5. Constrained minimization is not covered.

6. Numerical testing is needed.

Conclusion

1. Provided that $T(\nabla_i f(x)) = \frac{1}{n} T(f)$, we get methods, which are always more efficient than usual FGM.

2. Sometimes the gain reaches $O(\sqrt{n})$.

3. We get better results on a problem class, which covers most of applications of Smoothing Technique with dense matrices.

4. Our method is oracle/iteration balanced ($O(n+m)$ operations for both).

But:

5. Constrained minimization is not covered.

6. Numerical testing is needed.

CONGRATULATIONS TO BT!