Bundle methods for stochastic programs

Oracles with on-demand accuracy for two-stage programming problems

Welington de Oliveira

BAS Lecture 27, June 16, 2016, IMPA

Level bundle method: convergence analysis

Level bundle method: oracle with on-demand accuracy

Algorithm

$$f_k^M(x) = \max_{j \in \mathcal{B}_k} \{f(x_j) + \langle g_j, x - x_j \rangle\}, \quad x^{k+1} = \arg\min \left\{ \frac{1}{2}\|x - \hat{x}_k\|^2 \; : \; f_k^M(x) \le f_k^{\text{lev}}, \; x \in X \right\}$$

Step 0. Choose $\gamma \in (0,1)$ and tol > 0. Given x_1 and $f_0^{\text{low}} \le f^*$, call the oracle to compute $(f(x_1), g_1)$. Define $f_1^{\text{up}} \leftarrow f(x_1)$, $\hat{x}_1 \leftarrow x_1$, $k \leftarrow 1$, $\mathcal{B}_1 \leftarrow \{1\}$, $l = 1$, $k(l) = 1$

Step 1. Set $\Delta_k \leftarrow f_k^{\text{up}} - f_k^{\text{low}}$. If $\Delta_k \le$ tol, stop.

Step 2. If $\Delta_k \le (1-\gamma)\Delta_{k(l)}$, set $k(l) \leftarrow k$, $l \leftarrow l+1$ and set $\hat{x}_k$ as the best candidate x_j. Otherwise set $\hat{x}_k \leftarrow \hat{x}_{k-1}$

Step 3. Set $f_k^{\text{lev}} = f_k^{\text{low}} + \gamma\Delta_k$ and try to obtain x^{k+1}. If the QP is infeasible, set $f_k^{\text{low}} \leftarrow f_k^{\text{lev}}$ and go back to Step 1.

Step 4. Call the oracle to compute $(f(x^{k+1}), g_{k+1})$. Set $f_k^{\text{up}} \leftarrow \min\{f_k^{\text{up}}, f(x^{k+1})\}$

Step 5. Choose $\mathcal{B}_{k+1} \supset \{k+1, k^a\}$, set $f_{k+1}^{\text{low}} \leftarrow f_k^{\text{low}}$, $k \leftarrow k+1$ and go back to Step 1.

Convergence analysis

Notice that:

- stability center are fixed along cycles $k(l) \leq j < k(l+1)$, $l = 1, 2, \ldots,$
- $\{f_j^{\text{lev}}\}_{l=k(l)}^{k(l+1)-1}$ is non increasing for any fixed l
- $\Delta_{k(l+1)} \leq (1-\gamma)\Delta_{k(l)}$ for all l
- $\Delta_{k(l+1)} \leq (1-\gamma)^l \Delta_{k(1)}$ for all l

Furthermore, $\lim_{l\to\infty} \Delta_{k(l)} = 0$.

Convergence amounts to showing that $l \uparrow \infty$!

Lemma

Let $\Lambda > 0$ the Lipschitz constant of f. Given $l \geq 0$ fixed and $k \geq k(l)$, we have that

$$\begin{aligned} \|x_{j+1} - \hat{x}_{k(l)}\|^2 &\geq \|x_j - \hat{x}_{k(l)}\|^2 + \|x_{j+1} - x_j\|^2 \\ &\geq \|x_j - \hat{x}_{k(l)}\|^2 + \left(\frac{(1-\gamma)}{\Lambda}\Delta_j\right)^2 \quad \forall\, j = k(l), \ldots, k\,. \end{aligned}$$

Propositon

Suppose that X is a compact set and let D be its diameter. Given $l \geq 0$ fixed, $k \geq k(l)$ and $\Delta_k > \mathrm{tol}$, the number of iterations between $k(l)$ and k is bounded by

$$\left(\frac{D\Lambda}{(1-\gamma)\Delta_k}\right)^2$$

Theorem

Suppose that X is a compact set and let D be its diameter. Then, to obtain $\Delta_k \leq \mathrm{tol}$ the algorithm performs at most

$$\left(\frac{D\Lambda}{(1-\gamma)\,\mathrm{tol}}\right)^2 \left(\frac{1}{1-(1-\gamma)^2}\right)$$

iterations.

Two-stage stochastic linear programming

In two-stage stochastic linear programming problems with finitely many scenarios $\xi^i = (q^i, T^i, W^i, h^i)$ we wish to solve the high dimensional LP

$$\begin{cases} \min & c^\top x + \sum_{i=1}^N p_i [{q^i}^\top y^i] \\ \text{s.t.} & Ax = b,\ x \geq 0 \\ & T^i x + W^i y^i = h^i,\ y^i \geq 0, \quad i = 1, \ldots, N \end{cases}$$

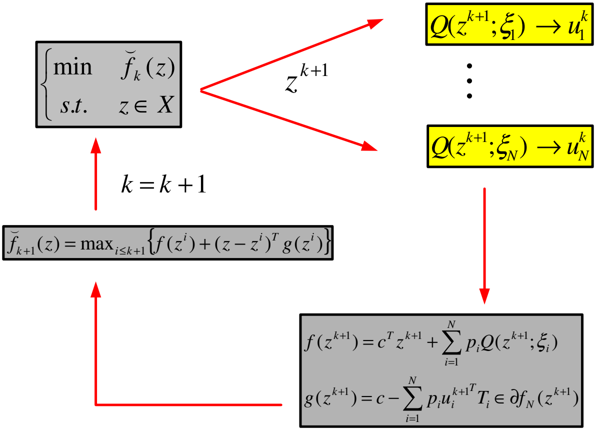
Two-stage decomposition

$$\min\ f(x) \quad \text{s.t.} \quad x \in X\,, \qquad \text{with} \qquad f(x) := c^\top x + \sum_{i=1}^N p_i Q(x, \xi^i)\,,$$

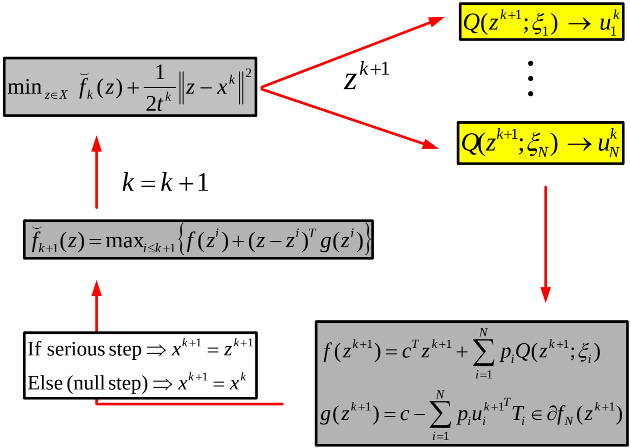
$$Q(x,\xi) = \begin{cases} \min & q^\top y \\ \text{s.t.} & Wy = h - Tx \\ & y \geq 0\,. \end{cases} \qquad \text{and} \qquad X := \{x \in \Re^n_+ : Ax = b\}$$

We know that $g = c - \sum_{i=1}^N p_i {T^i}^\top \pi^i \in \partial f(x)$, where π^i is a dual solution of $Q(x, \xi^i)$

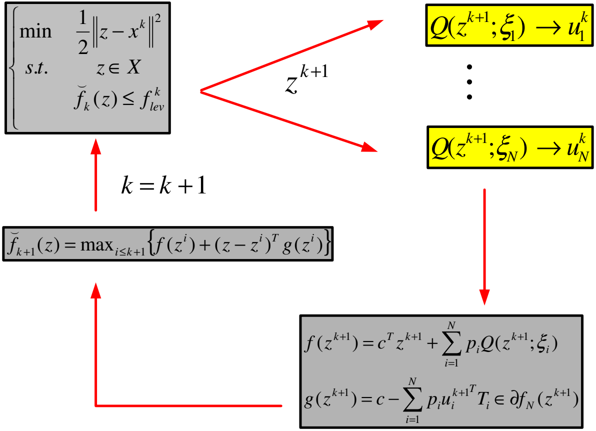
Cutting-plane method: L-Shaped method



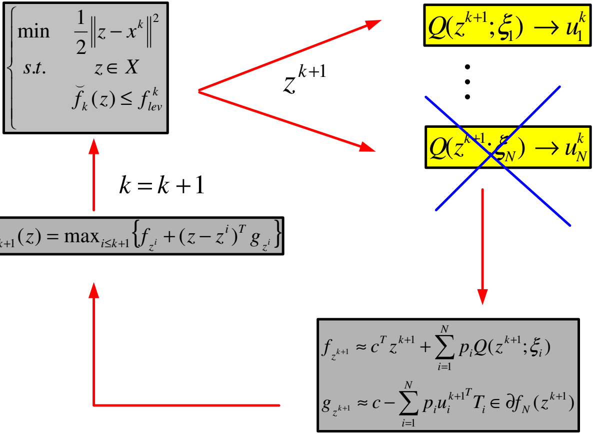
Proximal bundle method: regularized decomposition



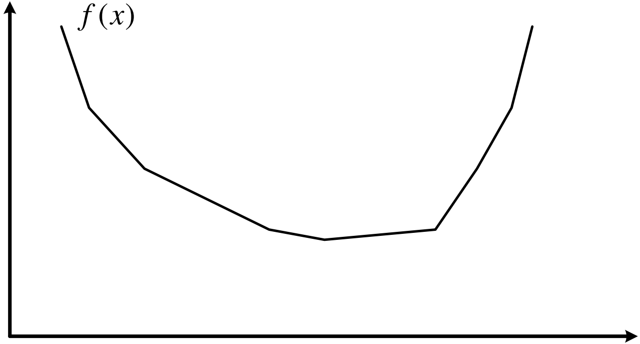
Level bundle method: Level decomposition



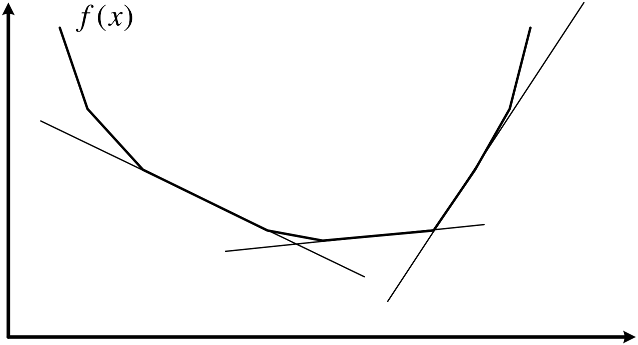
Inexact level decomposition



Exact linearizations

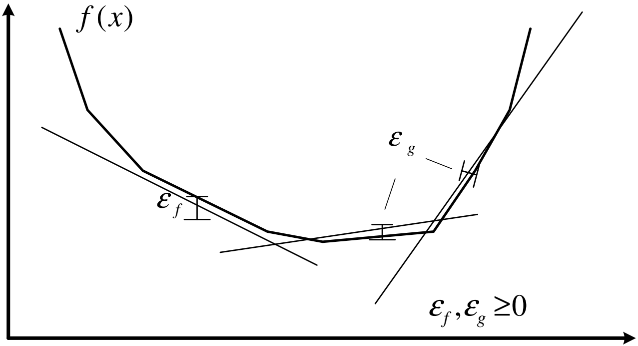


Exact linearizations



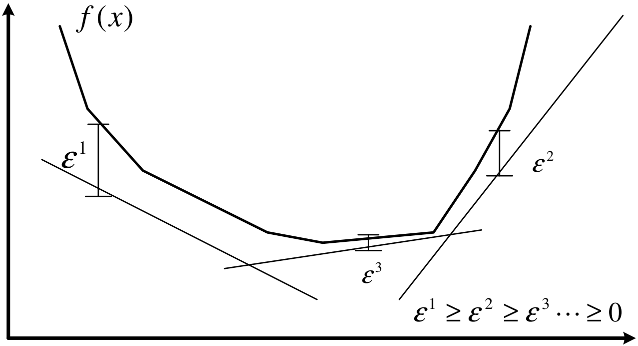
Oracle error: $\epsilon = 0$

Inexact linearizations



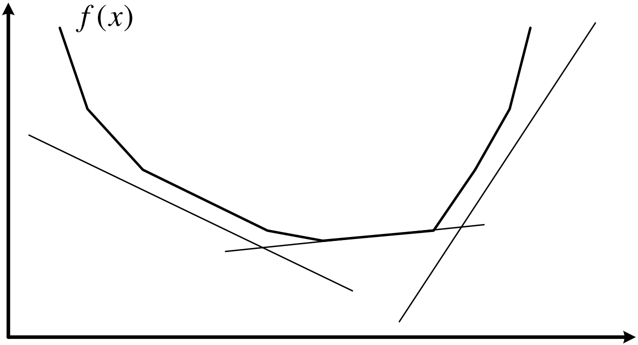
Inexact oracle of the upper case: $\epsilon > 0$ and $\epsilon^g > 0$

Asymptotic exact linearizations



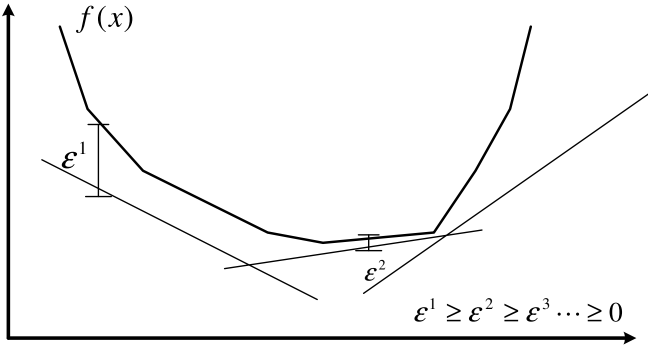
Asymptotic exact oracle: $\epsilon \downarrow 0$ and $\epsilon^g = 0$

Partly exact linearizations



Partly exact oracle: $\epsilon^g = 0$ (always) and $\epsilon = 0$ only serious steps

On-demand accuracy



Oracle with on-demand accuracy

combines asymptotically and partly exact oracles

Oracles with on-demand accuracy

Exact oracle

$$x \longrightarrow \boxed{\textbf{oracle}} \longrightarrow \begin{cases} f(x) \\ g \in \partial f(x) \end{cases}$$

We will consider a more general oracle

Oracles with on-demand accuracy

Exact oracle

$$x \longrightarrow \boxed{\textbf{oracle}} \longrightarrow \begin{cases} f(x) \\ g \in \partial f(x) \end{cases}$$

We will consider a more general oracle

Oracle with on-demand accuracy

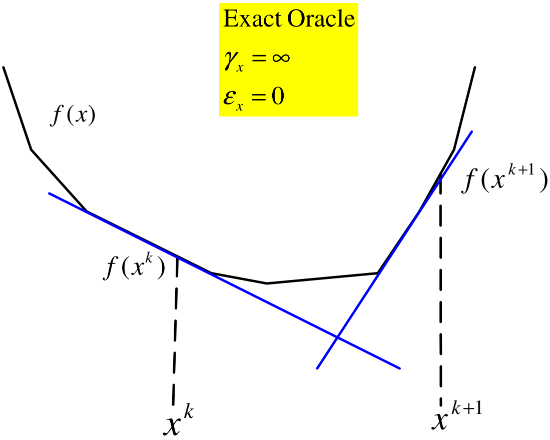
$$\begin{pmatrix} x \\ \gamma_x \\ \epsilon_x \end{pmatrix} \longrightarrow \boxed{\textbf{oracle on-d}} \longrightarrow \begin{cases} f_x = f(x) - \eta_x \\ g_x \in \partial_{\eta_x} f(x) \end{cases}$$

We assume that $\eta_x \geq 0$ is bounded from above by the informed tolerance $\epsilon_x \geq 0$, whenever $f_x \leq \gamma_x$. As a result,

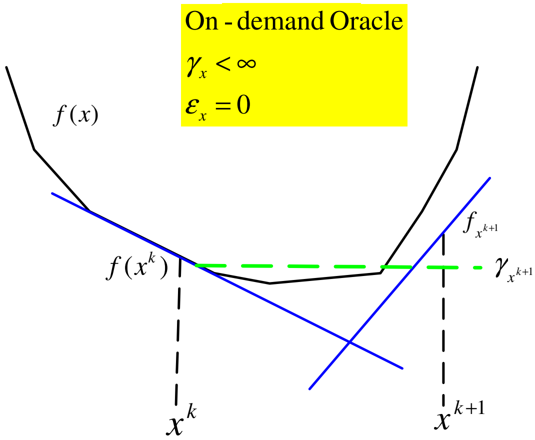
$$f_x \in [f(x) - \epsilon_x, f(x)] \quad \text{if} \quad f_x \leq \gamma_x \,.$$

If $\gamma_x := \infty$ and $\epsilon_x := 0$ for all $x \in X$, then the resulting oracle is of the exact type.

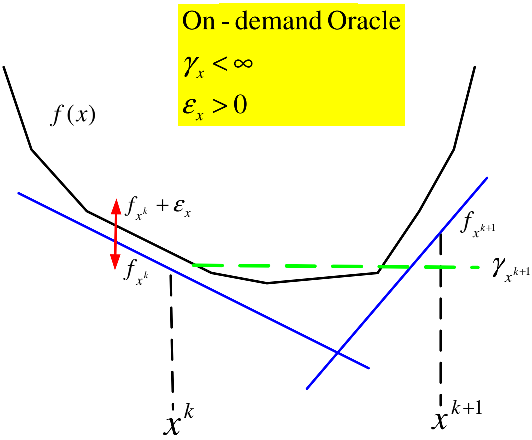
Exact linearizations



On-demand accuracy: exact and inexact Linearizations



On-demand accuracy: asymp. and inexact linearizations



Two-stage stochastic linear programming

Primal-dual formulations

- $\min_{x \in X} \; f(x), \quad \text{with} \quad X = \{x \in \Re^n_+ : \; Ax = b\}$
- $f(x) = c^\top x + \sum_{i=1}^N p_i Q(x, \xi^i)\,,$

$$Q(x, \xi^i) = \left\{ \begin{array}{ll} \min_y & q^\top y \\ \text{s.t.} & Wy = h^i - T^i x \\ & y \geq 0. \end{array} \right. \quad \equiv \quad \left\{ \begin{array}{ll} \max_u & (h^i - T^i x)^\top \pi \\ \text{s.t.} & W^\top \pi \leq q \end{array} \right.$$

$$= (h^i - T^i x)^\top \pi_i,$$

where $\Pi = \{\pi : \; W^\top \pi \leq q\}$ is a fixed set.

Suppose that a finite set $\mathcal{D} \subset \{\pi : W^\top \pi \leq q\}$ is given. Set $j = 0$.

Given $x^k \in X$, $\epsilon_x^k \geq 0$ and $\gamma_x^k \in \Re$

1. Fast approximation:

$$\forall i \;\; \text{find} \;\; \pi_i := \arg\max \; \pi^\top(\xi^i - Tx) \;\; \text{s.t.} \;\; \pi \in \mathcal{D}.$$

2. Approximating the oracle information:

$$f_x^k = c^\top x^k + \sum_{i=1}^N p_i \pi_i^\top (h^i - T^i x^k) \quad \text{and} \quad g_x^k = c - \sum_{i=1}^N p_i T^\top \pi_i$$

3. If $f_x^k > \gamma_x^k$ or $j = N$ stop: return (f_x^k, g_x^k)
4. Set $j = j + 1$ and (inexactly) solve the jth LP

$$\text{find } \pi_j \in \{\pi : W^\top \pi \leq q\} \text{ such that } Q(x^k, \xi^j) - \pi_j^\top (h^j - T^j x^k) \leq \epsilon_x^k$$

5. Add π_j to $\mathcal{D}$ and go back to Step 2

Level bundle methods for oracle with on-demand accuracy

The method generates a sequence of points $\{x^k\}_k \subset X$. At iteration k, $\mathcal{B}_k \subset \{1, 2, \ldots, k\}$ is an index set, and

$$\{x_j, f_x^j, g_x^j\}_{j\in\mathcal{B}_k} \quad \text{is the bundle of information}$$

that provides the cutting-planing model for f

$$f_k^M(x) := \max_{j\in\mathcal{B}_k}\{f_x^j + \langle g_x^j, x - x^j\rangle\} \leq f(x)\,.$$

$$x^{k+1} = \arg\min_{x\in X^k} \frac{1}{2}\|x - \hat{x}_k\|^2$$

$X^k := \{x \in X : f_k^M(x) \leq f_k^{\text{lev}}\}$

Ingredients

$f_k^M(x)$	$:= \max_{j \in \mathcal{B}_k} \{ f_x^j + \langle g_x^j, x - x^j \rangle \}$	model
f_k^{up}	$:= \min_{1 \le j \le k} \; f_x^j + \epsilon_x^j$	upper bound
f_k^{low}	$:= \min_{x \in X} f_k^M(x)$	lower bound
f_k^{lev}	$:= \gamma f_k^{\text{up}} + (1-\gamma) f_k^{\text{low}}$	level parameter
$\mathbb{X}_k$	$:= \{ x \in X : \; \check{f}_k(x) \le f_k^{\text{lev}} \}$	level set
Δ_k	$:= f_k^{\text{up}} - f_k^{\text{low}}$	optimality gap
γ_x^{k+1}	$:= \frac{1}{2} f_k^{\text{up}} + \frac{1}{2} f_k^{\text{lev}}$	descent target (possible choice)
ϵ_x^{k+1}	$:= \frac{1}{2} \Delta_k$	oracle error
x^{k+1}	$:= \arg\min_{x \in X^k} \frac{1}{2} \lVert x - \hat{x}^k \rVert^2$	next iterate.

Level bundle method with on-demand accuracy

Step 0. Choose $\gamma \in (0,1)$ and tol > 0. Given x_1 and $f_0^{\text{low}} \leq f^*$, call the oracle with $\epsilon_x^1 \geq 0$ and $\gamma_x^1 = \infty$ to compute (f_x^1, g_x^1). Define $f_1^{\text{up}} \leftarrow f_x^1 + \epsilon_x^1$, $\hat{x}_1 \leftarrow x_1$, $k \leftarrow 1$, $\mathcal{B}_1 \leftarrow \{1\}$, $l = 1$, $k(l) = 1$

Step 1. Set $\Delta_k \leftarrow f_k^{\text{up}} - f_k^{\text{low}}$. If $\Delta_k \leq \text{tol}$, stop.

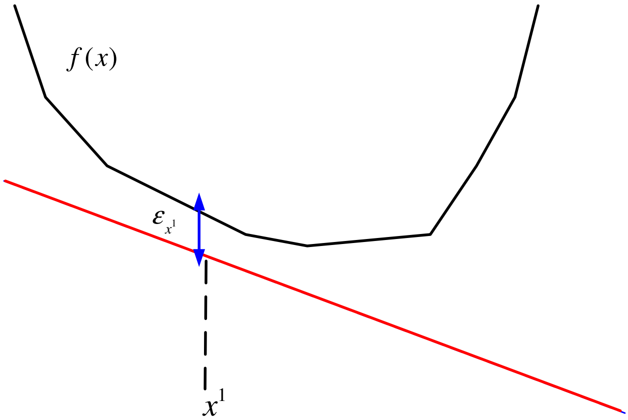
Step 2. If $\Delta_k \leq (1-\gamma)\Delta_{k(l)}$, set $k(l) \leftarrow k$, $l \leftarrow l+1$ and set $\hat{x}_k$ as the best candidate x_j. Otherwise set $\hat{x}_k \leftarrow \hat{x}_{k-1}$

Step 3. Set $f_k^{\text{lev}} = f_k^{\text{low}} + \gamma\Delta_k$ and try to obtain x^{k+1}. If the QP is infeasible, set $f_k^{\text{low}} \leftarrow f_k^{\text{lev}}$ and go back to Step 1.

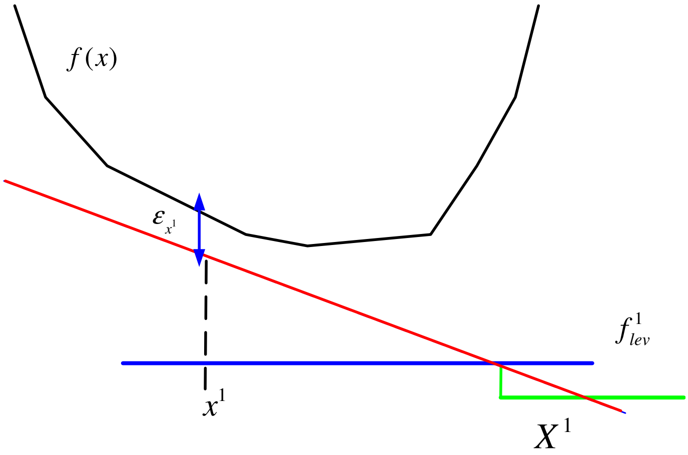
Step 4. Set $\epsilon_x^{k+1} = \Delta_k/2$ and $\gamma_x^{k+1} = (f_k^{\text{up}} + f_k^{\text{lev}})/2$. Call the oracle to compute (f_x^{k+1}, g_x^{k+1}). If $f_x^{k+1} \leq \gamma_x^{k+1}$, then set $f_k^{\text{up}} \leftarrow \min\{f_k^{\text{up}}, f_x^{k+1} + \epsilon_x^{k+1}\}$

Step 5. Choose $\mathcal{B}_{k+1} \supset \{k+1, k^a\}$, set $f_{k+1}^{\text{low}} \leftarrow f_k^{\text{low}}$, $k \leftarrow k+1$ and go back to Step 1.

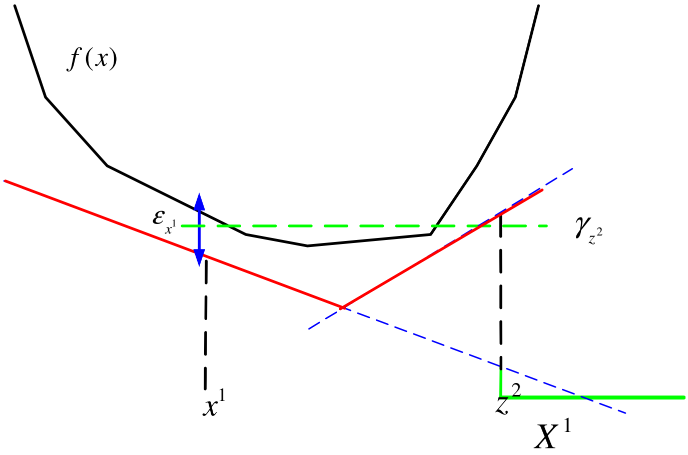
Level bundle method for oracle with on-demand accuracy



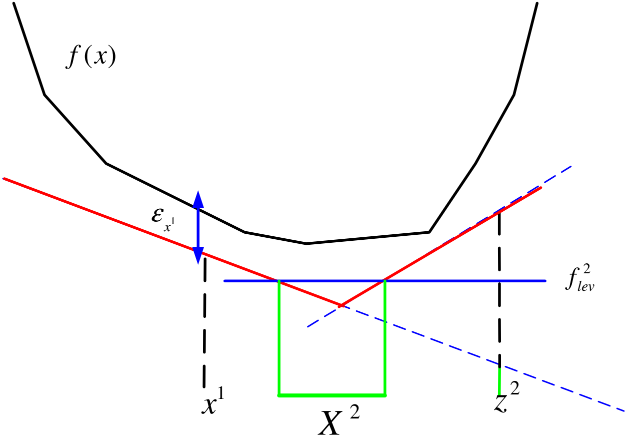
Level bundle method for oracle with on-demand accuracy



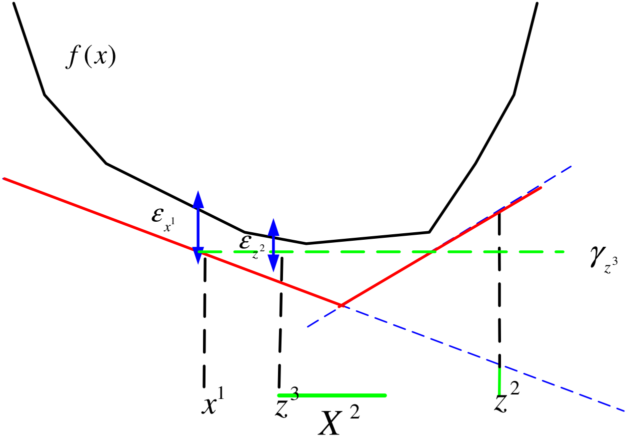
Level bundle method for oracle with on-demand accuracy



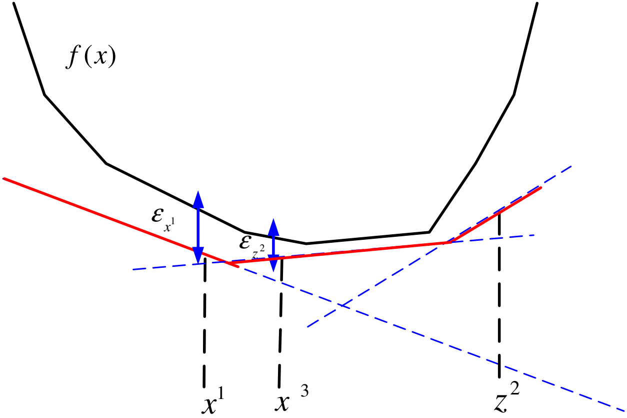
Level bundle method for oracle with on-demand accuracy



Level bundle method for oracle with on-demand accuracy



Level bundle method for oracle with on-demand accuracy



Numerical experiments

Two-stage stochastic linear programming

- $\min_{x \in X} \; f(x), \quad \text{with} \quad X = \{x \in \Re^n_+ : \; Ax = b\}$
- $f(x) = c^\top x + \sum_{i=1}^N p_i Q(x, \xi^i)\,,$

$$Q(x, \xi^i) = \begin{cases} \min_y & q^\top y \\ \text{s.t.} & Wy = h^i - T^i x \\ & y \geq 0. \end{cases}$$

For 10 different problems, 15 instances corresponding to

$$N \in \{10, 20, 30, 40, 50, 80, 100, 150, 200, 250, 300, 350, 400, 450, 500\}$$

scenarios were considered.

150 different instances

Numerical experiments

All solvers provided tol-solutions

- CP= Cutting-plane method
- μ = Proximal bundle method
- ℓ = Level bundle method

Tabela: Total CPU times in hours

CP Ex	CP AE	μ Ex	ℓ Ex	ℓ PAE	ℓ AE	ℓ PI1	ℓ PI2	ℓ PAE
7.4	6.3	5.6	5.2	4.4	4.4	2.6	2.5	2.1

Tabela: Percentile CPU time reduction ($CPU\%$)

CP AE	μ Ex	ℓ Ex	ℓ PAE	ℓ AE	ℓ PI1	ℓ PI2	ℓ PAE
15	24	30	41	41	65	66	72

Obrigado!

Gracias!

Grazie!

Thank you!