Exploiting uncontrolled information in nonsmooth optimization

Sofia Zaourar

Inria Grenoble

PGMO, October 2013

Joint work with Jérôme Malick (CNRS, Grenoble)
and Welington Oliveira (IMPA, Rio, Brazil)

Outline

1. Nonsmooth optimization and fine oracles
2. Additional information: uncontrolled coarse oracles
3. A way to exploit uncontrolled information
4. Numerical illustration

Outline

General framework

The problem we want to solve

$$\boxed{f_* = \min_{x \in X} \; f(x)}$$

- $f : \mathbb{R}^n \to \mathbb{R}$ **convex**, **nonsmooth** and **expensive to evaluate**
- $X \subseteq \mathbb{R}^n$ polyhedral set

Typical situation

$$\boxed{f(x) := \max_{u \in U} \; h(u, x)}$$

with $h(u, \cdot)$ convex for all $u \in U$

Applications

- Lagrangian relaxation
- Benders decomposition
- Stochastic optimization

Example 1: Unit-commitment problem

Data and variables

- n generation plants, T time periods
- $d \in \mathbb{R}^T$ demand forecast
- $p_i \in \mathbb{R}^T$ generation schedule for plant i

The problem (simplified form)

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{i=1}^{n} c_i(p_i) && \textit{// total cost} \\
\text{subject to} \quad & \sum_{i=1}^{n} p_i = d && \textit{// balance constraint} \\
& p_i \in \mathcal{P}_i && \textit{// technical constraints}
\end{aligned}
$$

Example 1: Unit-commitment problem

Resolution by Lagrangian relaxation

Dual problem (dualization of the balance constraints)

$$\max_{x \in \mathbb{R}^T} \overbrace{\sum_{i=1}^{n} \underbrace{\min_{p_i \in \mathcal{P}_i} \{c_i(p_i) - x^\top p_i\}}_{\text{Lagrangian subproblems}} + x^\top d}^{\text{dual function}}$$

Example 1: Unit-commitment problem

Resolution by Lagrangian relaxation

Dual problem (dualization of the balance constraints)

$$\max_{x \in \mathbb{R}^T} \overbrace{\sum_{i=1}^{n} \underbrace{\min_{p_i \in \mathcal{P}_i} \{c_i(p_i) - x^\top p_i\}}_{\text{Lagrangian subproblems}} + x^\top d}^{\text{dual function}}$$

Equivalent formulation

$$\min_{x \in \mathbb{R}^T} f(x) \quad \text{with} \quad f(x) := \sum_{i=1}^{n} \max_{p_i \in \mathcal{P}_i} \{-c_i(p_i) + x^\top p_i\} - x^\top d$$

$\rightsquigarrow$ $f(\cdot)$ convex, nonsmooth and expensive to evaluate

Example 2: Stochastic programming

[Shapiro et al 2009]

Two-stage stochastic linear problem

- x here-and-now decision variables
- $c^\top x$ present cost
- $Q(x; D)$ future cost depending on some random parameter D

$$\begin{cases} \min_x & c^\top x + \mathbb{E}[\,Q(x; D)\,] \\ \text{s.t.} & x \in X = \{x \in \mathbb{R}^n_+ : Ax = b\} \end{cases}$$

Discrete probability distribution

N scenarios: $d_1, \ldots, d_N$, with probability $p_1, \ldots, p_N$ (N large)

$$\mathbb{E}[\,Q(x; D)\,] := \sum_{i=1}^{N} p_i Q(x, d_i)$$

Example 2: Stochastic programming

[Shapiro et al 2009]

Future cost

$$Q(x, d_i) := \begin{cases} \min\limits_{y \geq 0} & q^\top y \\ \text{s.t.} & Tx + Wy = d_i \end{cases}$$

Example 2: Stochastic programming

[Shapiro et al 2009]

Future cost

$$Q(x, d_i) := \begin{cases} \min\limits_{y \geq 0} & q^\top y \\ \text{s.t.} & Tx + Wy = d_i \end{cases}$$

$$= \begin{cases} \max\limits_{u} & (d_i - Tx)^\top u \\ \text{s.t.} & W^\top u \leq q \end{cases}$$

Example 2: Stochastic programming

[Shapiro et al 2009]

Future cost

$$Q(x, d_i) := \begin{cases} \min\limits_{y \geq 0} & q^\top y \\ \text{s.t.} & Tx + Wy = d_i \end{cases}$$

$$= \begin{cases} \max\limits_{u} & (d_i - Tx)^\top u \\ \text{s.t.} & W^\top u \leq q \end{cases}$$

Equivalent formulation

$$\min_{x \in X} f(x) \quad \text{with} \quad f(x) := c^\top x + \sum_{i=1}^{N} p_i \max_{W^\top u \leq q} (d_i - Tx)^\top u$$

$\rightsquigarrow$ $f(\cdot)$ convex, nonsmooth and expensive to evaluate

In general: Evaluation of $f(\cdot)$ via an oracle

Exact evaluation

$x \longrightarrow$ **Exact oracle** $\longrightarrow f(x) \in \mathbb{R}$

$\longrightarrow g \in \mathbb{R}^n \text{ s.t. } f(x) + g^\top(\cdot - x) \leq f(\cdot)$

Fine evaluation - for a given accuracy $\eta \geq 0$

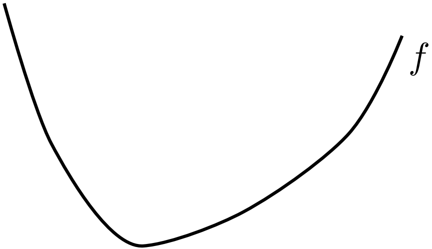
$x \longrightarrow$ **Fine oracle** $\longrightarrow f_x \text{ s.t. } f(x) - \eta \leq f_x \leq f(x)$

$\longrightarrow g_x \text{ s.t. } f_x + g_x^\top(\cdot - x) \leq f(\cdot)$

Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

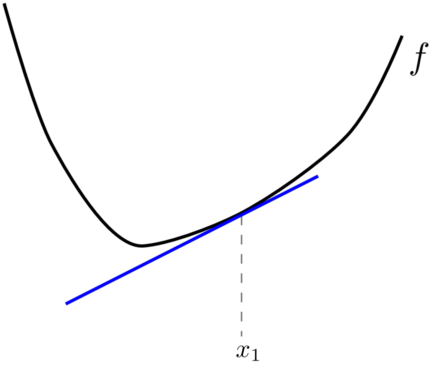
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\ldots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

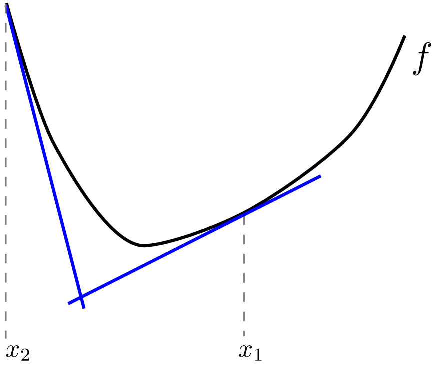
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\ldots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

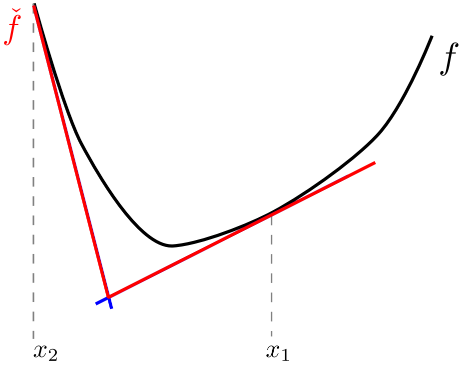
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\ldots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

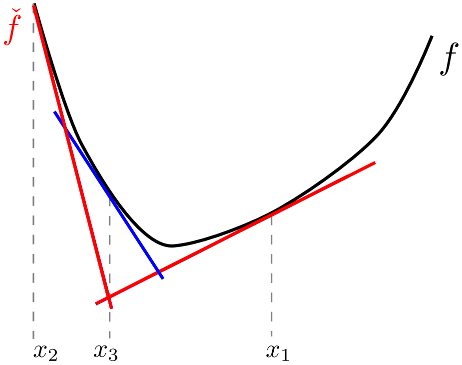
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\dots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

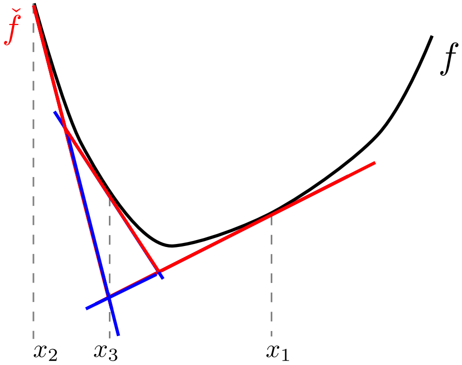
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\dots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

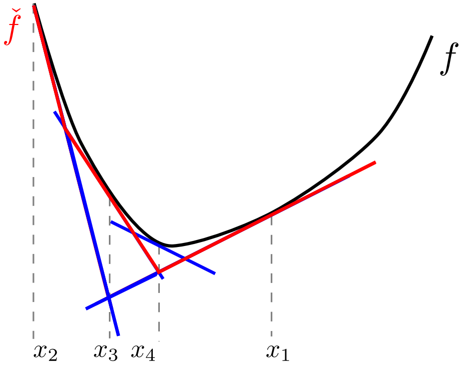
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\ldots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

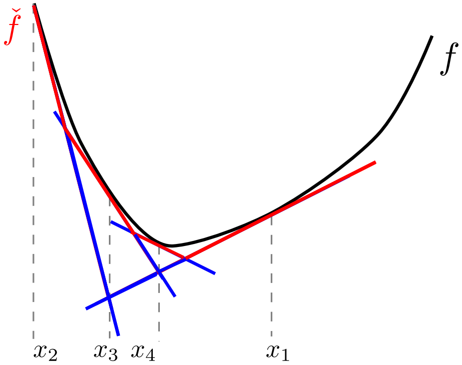
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\ldots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

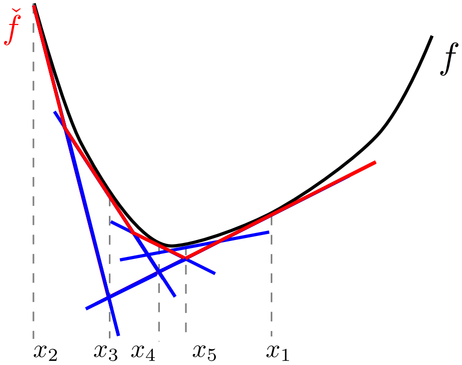
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\ldots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

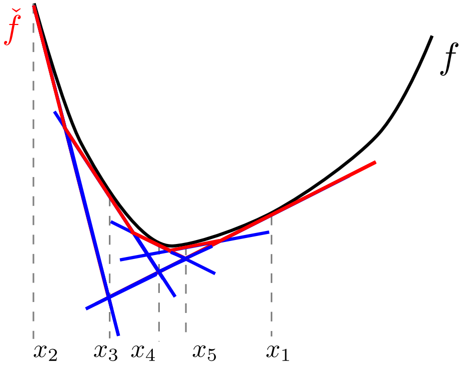
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\dots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

Bundle methods [Lemarechal 1974] are methods of choice

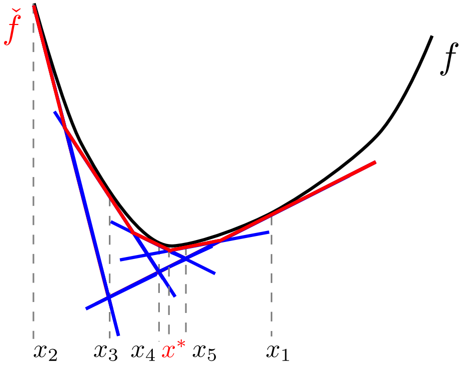
- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\dots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



Minimizing a nonsmooth convex function

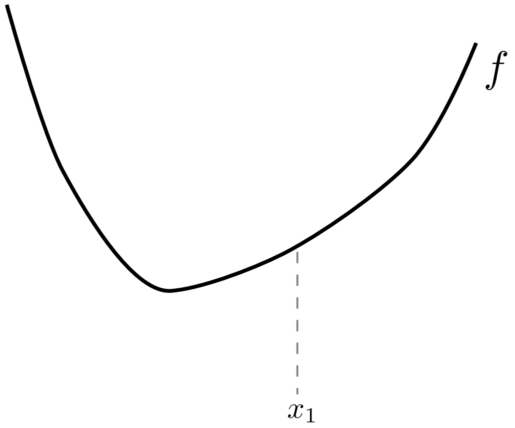
Bundle methods [Lemarechal 1974] are methods of choice

- Use cutting plane model $\check{f}_k(x) := \max_{i=1,\dots,k} \{ f_i + g_i^\top (x - x_i) \}$
- Basic example: Kelley method [Kelley 1960] $x_{k+1} \in \arg\min_{x \in X} \check{f}_k(x)$



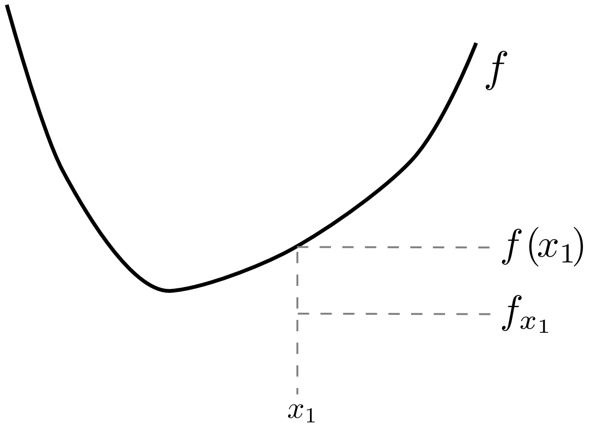
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



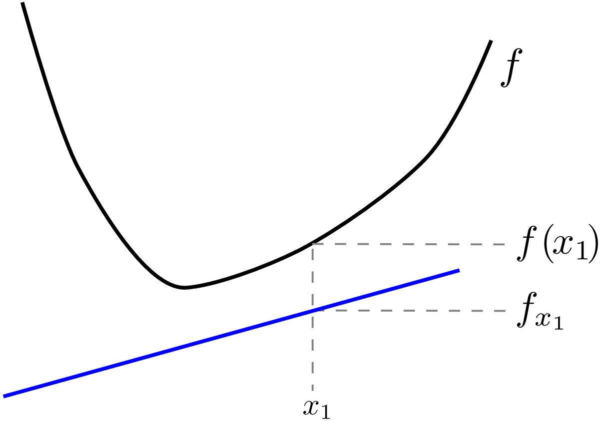
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



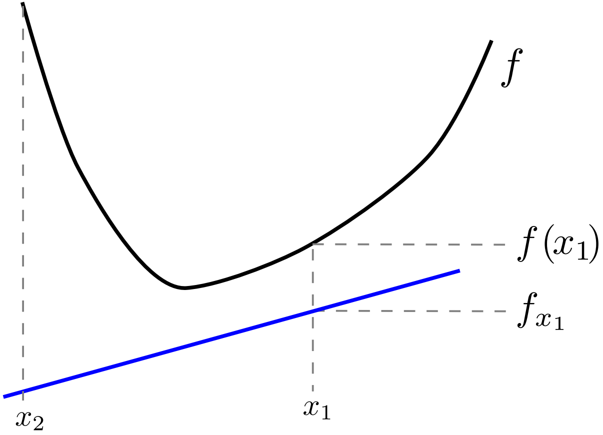
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



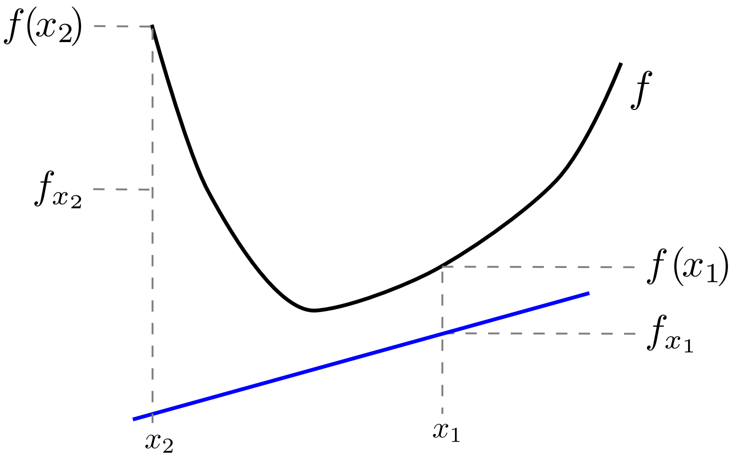
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



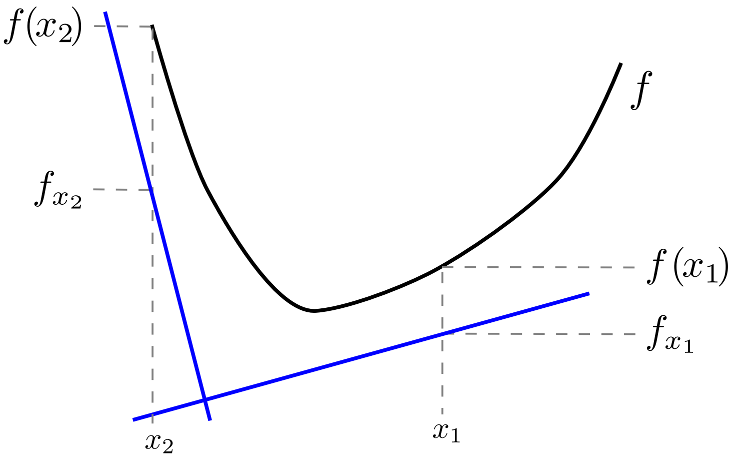
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



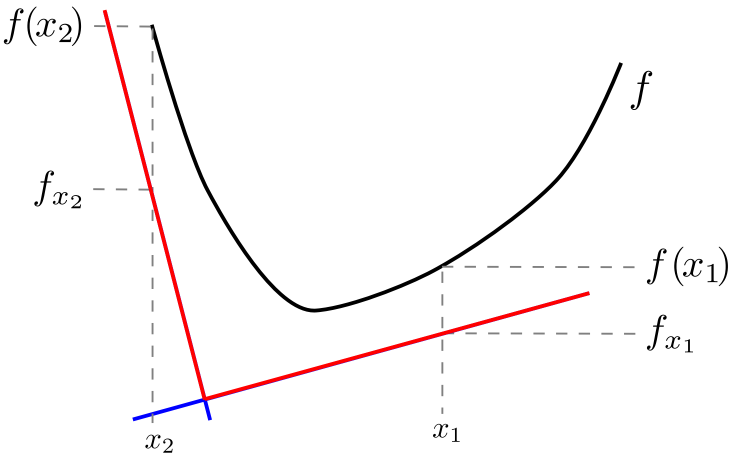
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



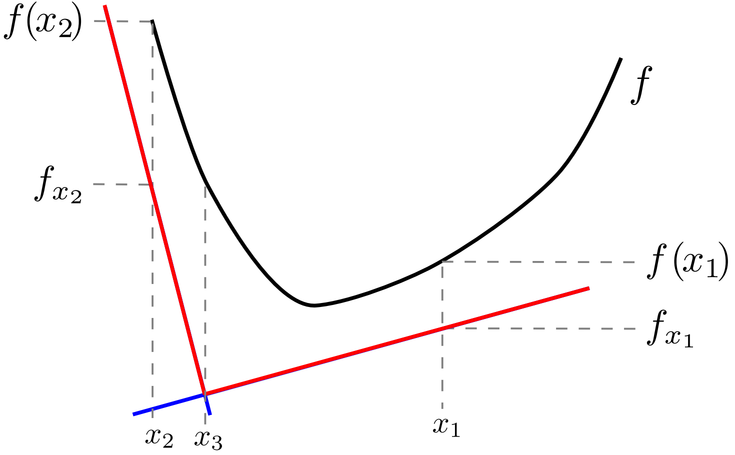
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



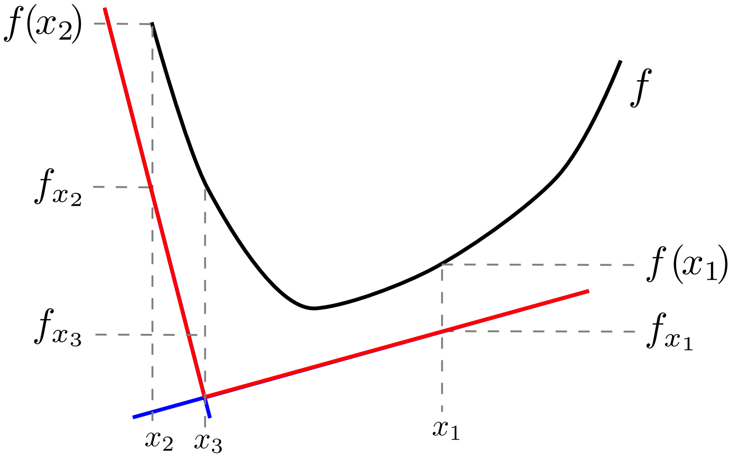
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



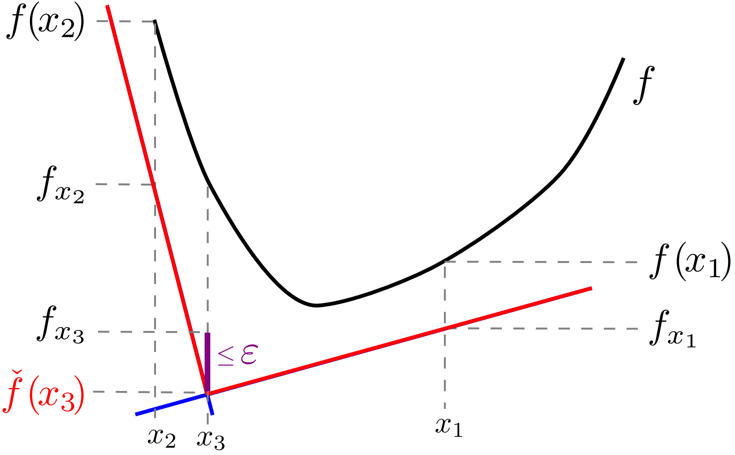
When the function is known through a fine oracle

For example: Kelley method + fine η-oracle



When the function is known through a fine oracle

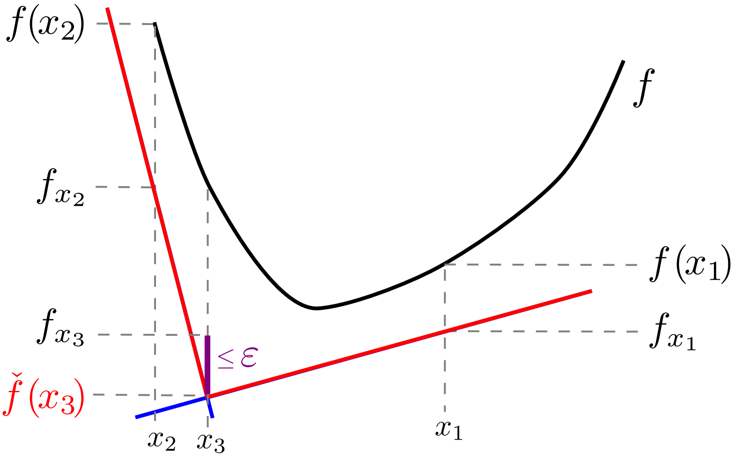
For example: Kelley method + fine η-oracle



When the function is known through a fine oracle

For example: Kelley method + fine η-oracle

Convergence to an $(\eta + \varepsilon)$-solution



Bundle methods & fine oracles

Stabilized Kelley method

Proximal bundle method

$$\min_{x \in X} \check{f}_k(x) + \frac{1}{2t_k} \|x - \hat{x}_k\|^2$$

Level bundle method

$$\min_{x \in X} \frac{1}{2} \|x - \hat{x}_k\|^2 \text{ s.t. } \check{f}_k(x) \leq \text{lev}_k$$

Convergence to ...

- **an approximate solution**: for a fixed upper bound on the oracle error η, the iterates are an η-minimizing sequence:

$$f^* \leq \liminf f(x_k) \leq f^* + \eta$$

  [Hintermuller 2001, Kiwiel 2006, Oliveira et al 2011]

- **an exact solution**: for vanishing oracle error $\eta_k \to 0$
  [Zakeri et al 2000, Oliveira et al 2011]

Outline

Another type of inexact oracles

Observation

Suppose

$$f(x) := \max_{u \in U} \; h(u, x)$$

For any $\bar{u} \in U$

- $h(\bar{u}, x) \leq f(x)$

Another type of inexact oracles

Observation

Suppose

$$f(x) := \max_{u \in U} \; h(u, x)$$

For any $\bar{u} \in U$

- $h(\bar{u}, x) \leq f(x) \quad \rightsquigarrow f_x := h(\bar{u}, x)$ under-approximation of $f(x)$

Another type of inexact oracles

Observation

Suppose

$$f(x) := \max_{u \in U} \; h(u, x)$$

For any $\bar{u} \in U$

- $h(\bar{u}, x) \leq f(x) \quad \rightsquigarrow f_x := h(\bar{u}, x)$ under-approximation of $f(x)$
- $g_x \in \partial_x h(\bar{u}, x)$: $h(\bar{u}, x) + g_x^\top(\cdot - x) \leq h(\bar{u}, \cdot) \leq f(\cdot)$

Another type of inexact oracles

Observation

Suppose

$$f(x) := \max_{u \in U} \; h(u, x)$$

For any $\bar{u} \in U$

- $h(\bar{u}, x) \leq f(x) \quad \rightsquigarrow f_x := h(\bar{u}, x)$ under-approximation of $f(x)$
- $g_x \in \partial_x h(\bar{u}, x) : \; h(\bar{u}, x) + g_x^\top (\cdot - x) \leq h(\bar{u}, \cdot) \leq f(\cdot)$

$\rightsquigarrow g_x$ approximate subgradient of f at x

Another type of inexact oracles

Observation

Suppose

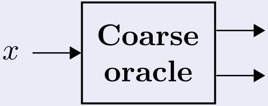
$$f(x) := \max_{u \in U} \, h(u,x)$$

For any $\bar{u} \in U$

- $h(\bar{u},x) \leq f(x) \quad \rightsquigarrow f_x := h(\bar{u},x)$ under-approximation of $f(x)$
- $g_x \in \partial_x h(\bar{u},x)$: $h(\bar{u},x) + g_x^\top(\cdot - x) \leq h(\bar{u},\cdot) \leq f(\cdot)$

$\rightsquigarrow g_x$ approximate subgradient of f at x

Coarse oracle: inexact oracle with uncontrolled accuracy



f_x s.t. $\quad f_x \leq f(x)$

g_x s.t. $\quad f_x + g_x^\top(\cdot - x) \leq f(\cdot)$

Coarse oracle for unit-commitment

$$f(x) = \sum_{i=1}^{n} \underbrace{\max_{p_i \in \mathcal{P}_i} h_i(p_i, x)}_{(L_i)} = \sum_{i=1}^{n} \max_{p_i \in \mathcal{P}_i} \{-c_i(p_i) + x^\top p_i\} - x^\top d$$

Some subpbroblems may be difficult to solve

EDF example:

- Some (L_i) are large Mixed Integer Programs, solved by a branch-and-bound algorithm (e.g. CPLEX).

The idea

Any feasible solution found during the algorithm gives coarse information on f (for free!)

Coarse oracle for stochastic programming

$$f(x) = \sum_{i=1}^{N} \underbrace{\max_{u \in U} h_i(u,x)}_{(S_i)} = \sum_{i=1}^{N} p_i \max_{W^\top u \leq q} (d_i - Tx)^\top u + c^\top x$$

Dealing with the huge number of scenarios

1. Solving approximately each scenario [Zakeri et al 2000]
   $\rightsquigarrow$ fine possibly expensive oracle
2. Solving accurately a small well-chosen subset of scenarios [Oliveira et al 2011]
   - u_i^* solution of (S_i) is an approximate solution for (S_j)

   $\rightsquigarrow$ cheap, possibly good, but uncontrolled oracle

What about using cheap uncontrolled oracles?

Interest

More cutting planes $\Longrightarrow$ **better model** $\Longrightarrow$ potentially better iterates

$$\check{f}(x) = \max\{f_{x_i} + g_{x_i}^\top (x - x_i) : i \in I^{\text{fine}} \cup I^{\text{coarse}}\}$$

Challenges

- Practical: **Larger quadratic problems**

$$\min_{x \in X} \check{f}_k(x) + \frac{1}{2t_k} \|x - \hat{x}_k\|^2 \quad \text{or} \quad \min_{x \in X : \check{f}_k(x) \leq \text{lev}_k} \frac{1}{2} \|x - \hat{x}_k\|^2$$

  - Constant advances in quadratic programming solvers
  - Additional cost negligible compared to fine oracle costs
- Theoretical: **Convergence proofs do not extend** directly in this case

Outline

Exploiting coarse uncontrolled oracles

Our settings: two oracles

Fine oracle	Coarse oracle
expensive	cheap / for free
error bounded by $\eta \geq 0$	unknown accuracy

Exploiting coarse uncontrolled oracles

Our settings: two oracles

	Fine oracle	Coarse oracle
	expensive	cheap / for free
	error bounded by $\eta \geq 0$	unknown accuracy

General scheme

Loop between 3 steps:

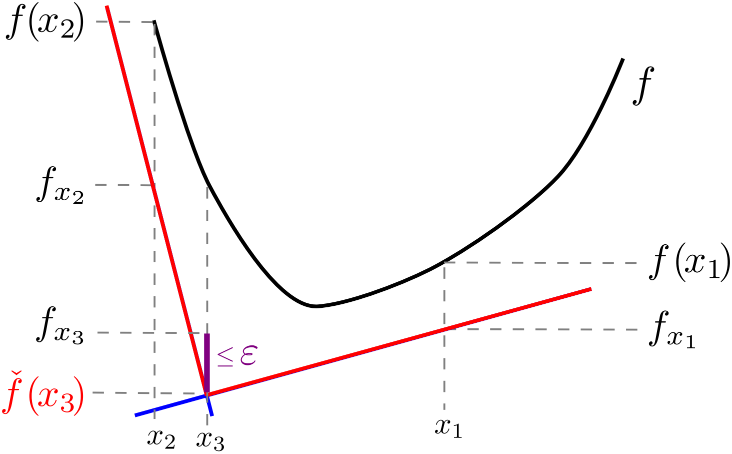
1. **Construct a cutting-plane model**
   - using only the coarse oracle
   - update the lower bound
2. **Test termination**
   - using upper and lower bounds
3. **Compute next iterate**
   - call the fine oracle
   - update the upper bound and stability center $\hat{x}_k$
   - add the exact cut to the bundle

For example: application to Kelley method (the algorithm)

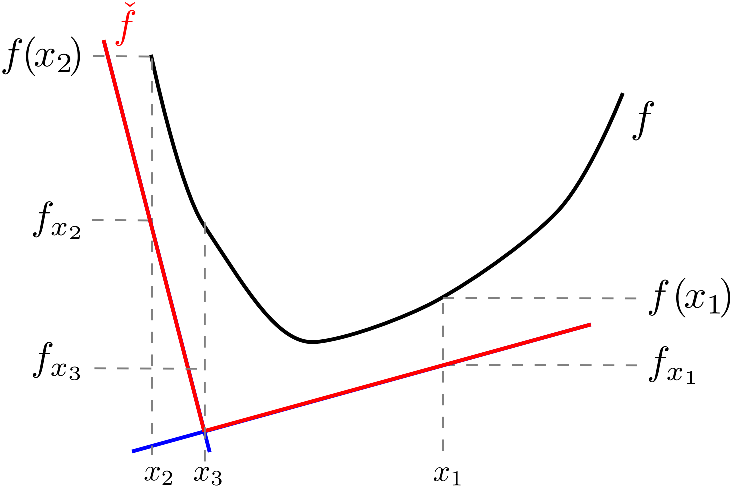
Algorithm scheme

1. **Run inexact Kelley method**
   - x_{k+1} minimum of $\check{f}_k(\cdot)$
   - Update the lower bound f^{low}
2. **Stopping test**
   - $f^{\text{up}} - f^{\text{low}} \leq \varepsilon$?
3. **Call the fine oracle**
   - $(f_{x_{k+1}}, g_{x_{k+1}}) \leftarrow \text{fine\_oracle}(x_{k+1})$
   - If $f_{x_{k+1}} < f_{\hat{x}_k}$ then $\hat{x}_{k+1} \leftarrow x_{k+1}$ else $\hat{x}_{k+1} \leftarrow \hat{x}_k$
   - Update the upper bound f^{up}
   - Add the fine plan $(f_{x_{k+1}}, g_{x_{k+1}})$ to the bundle

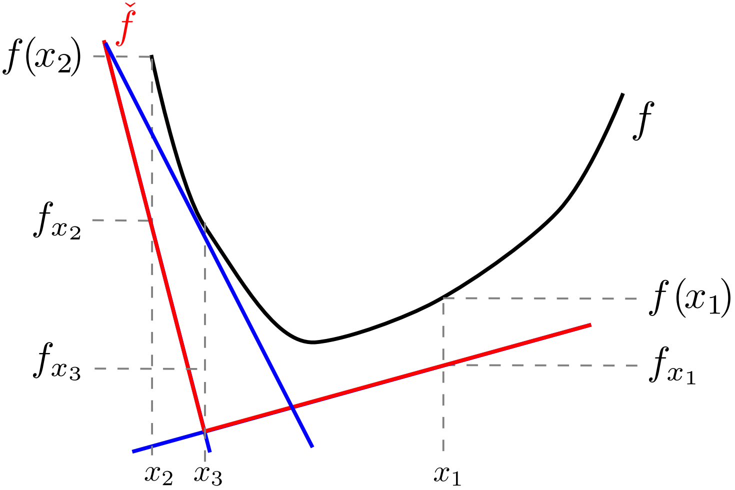
For example: application to Kelley method (illustration)



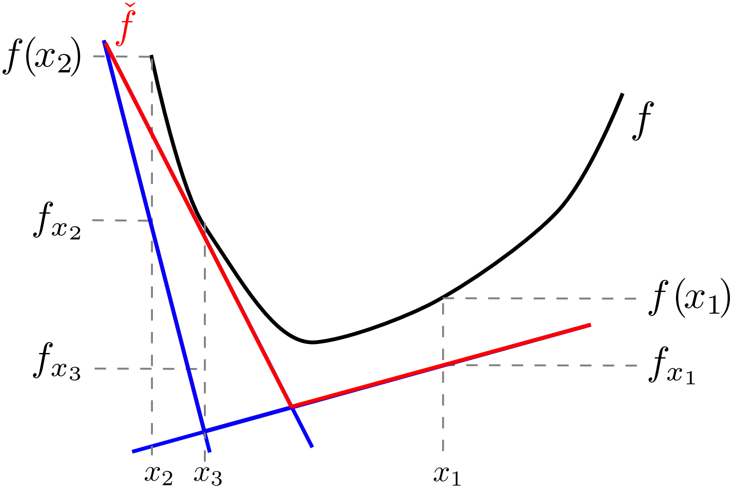
For example: application to Kelley method (illustration)



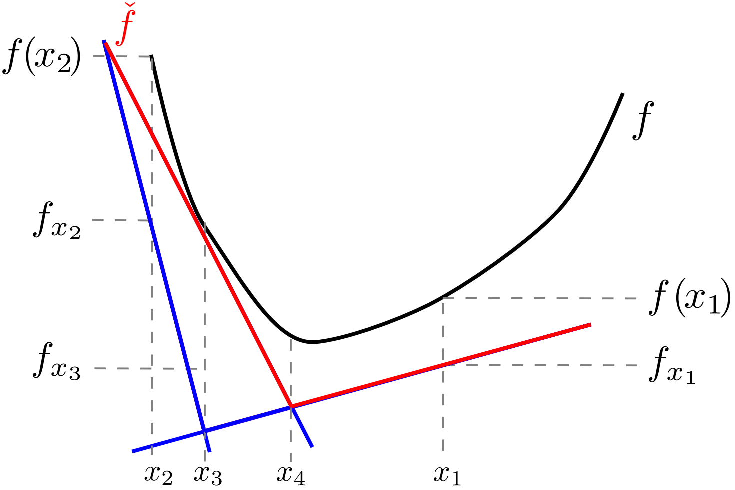
For example: application to Kelley method (illustration)



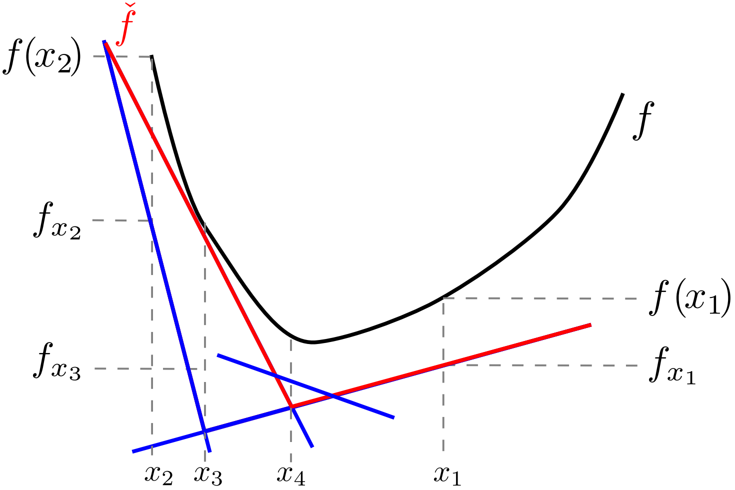
For example: application to Kelley method (illustration)



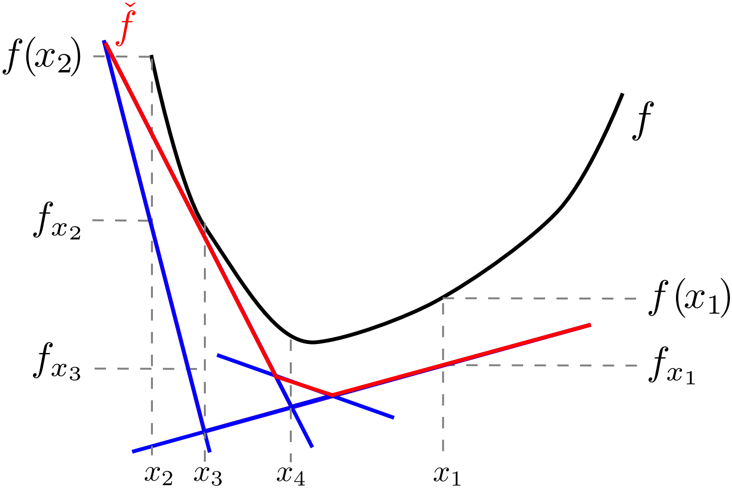
For example: application to Kelley method (illustration)



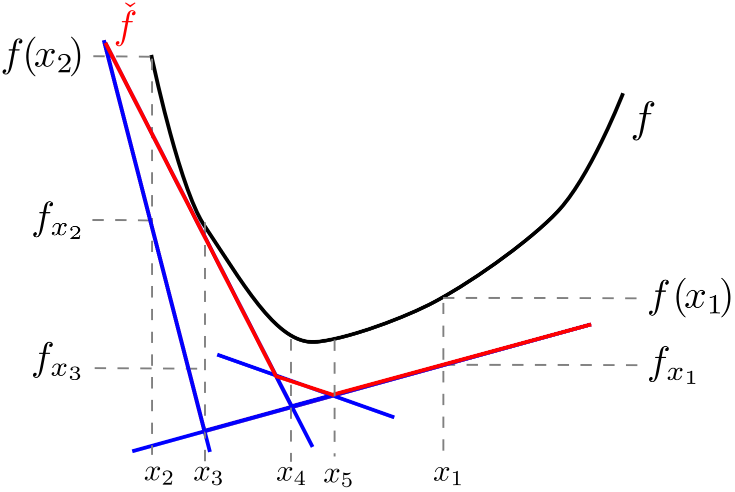
For example: application to Kelley method (illustration)



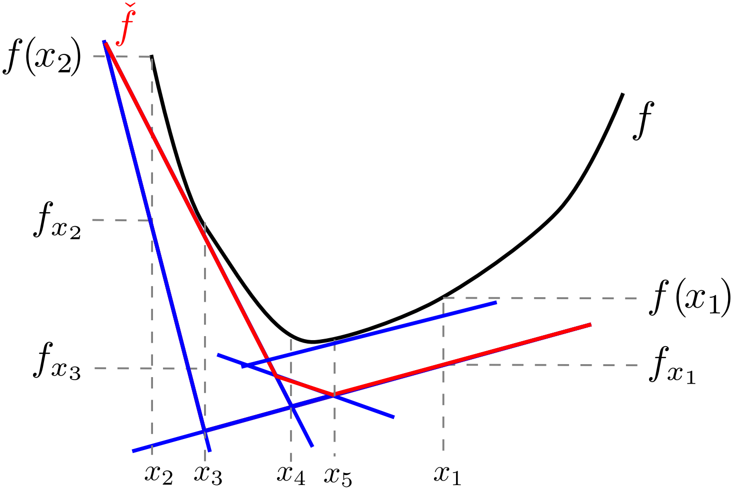
For example: application to Kelley method (illustration)



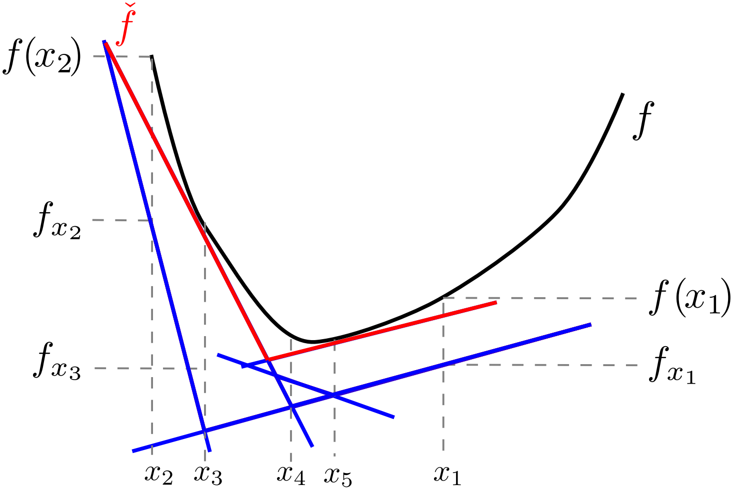
For example: application to Kelley method (illustration)



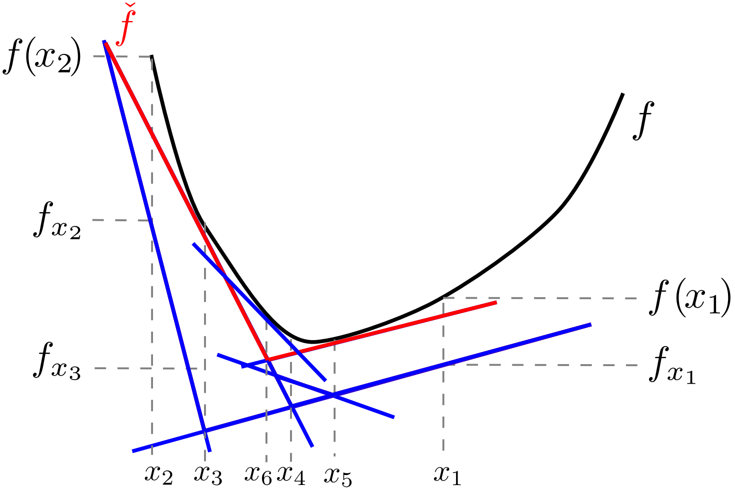
For example: application to Kelley method (illustration)



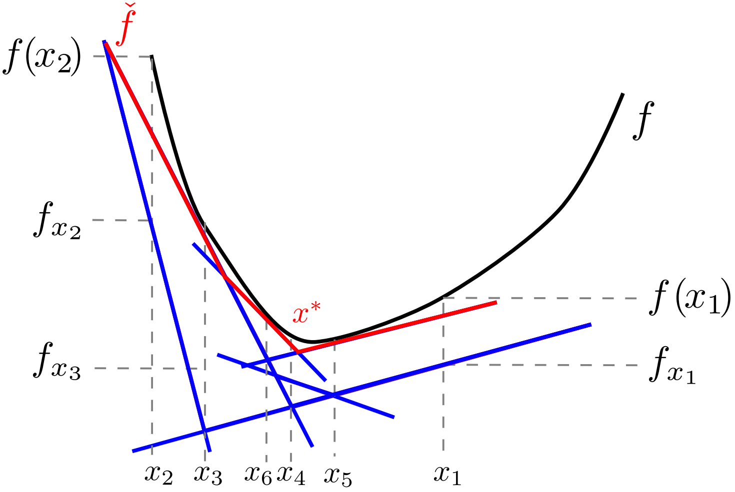
For example: application to Kelley method (illustration)



For example: application to Kelley method (illustration)



For example: application to Kelley method (illustration)



For example: application to Kelley method (convergence)

Lemma (Nonpositivity of the gap and convergence)

Suppose:

- f_k^{low} a nondecreasing sequence
- $\hat{x}_k$ such that $f_{\hat{x}_k}$ is a nonincreasing sequence
- for all k, $f_k^{\text{low}} \leq f_* \leq f_{\hat{x}_k} + \eta$

If $\lim_k f_{\hat{x}_k} - f_k^{\text{low}} \leq 0$, then

$$f_* - \eta \leq \lim_k f_{\hat{x}_k} \leq f_*$$

Theorem (convergence of inexact Kelley method)

If X is compact and the tolerance is set to 0, then the iterates $\hat{x}_k$ generate an η-minimizing sequence:

$$f_* - \eta \leq \lim_k f_{\hat{x}_k} \leq f_* \leq \liminf f(\hat{x}_k) \leq f_* + \eta$$

A more elaborated application: level bundle method

Main ingredients

- Abstract external module using the coarse oracle
- More sophisticated stopping test and level management

Nice features

- No assumption on the set X
- Bundle compression possible
  - Coarse linearizations can be numerous (with little information)
  - In theory: only two planes are necessary to ensure convergence

Asymptotic convergence to an η-solution

$$f_* \leq \liminf f(\hat{x}_k) \leq f_* + \eta$$

Outline

1. Nonsmooth optimization and fine oracles
2. Additional information: uncontrolled coarse oracles
3. A way to exploit uncontrolled information
4. Numerical illustration

Illustration on unit-commitment problems

Test problems

- Unit commitment problems with classical constraints:
  - Min/max generation
  - Min/max ramps
  - Minimum uptime and downtime
  - Load balance
- 10 randomly generated, realistic instances

Our settings

- **2 oracles**
  - Fine oracle: Gurobi MILP solver
  - Coarse oracle: all the feasible solutions visited by the fine oracle
- **2 algorithms**
  - Standard Kelley method using fine oracle
  - Enhanced Kelley method using both fine and coarse oracles

Illustration on unit-commitment problems

Results

Instance #periods, #units	Time reduction	Oracle calls reduction	Avg nb of added planes
24, 8	19 %	19 %	3
24, 16	28 %	29 %	7
24, 32	25 %	17 %	15
24, 48	17 %	15 %	23
24, 81	42 %	43 %	23
32, 16	21 %	20 %	10
48, 6	14 %	25 %	2
48, 12	26 %	27 %	8
48, 16	21 %	22 %	10
Average	24 %	24 %	11

Illustration on unit-commitment problems

Results

Instance #periods, #units	Time reduction	Oracle calls reduction	Avg nb of added planes
24, 8	19 %	19 %	3
24, 16	28 %	29 %	7
24, 32	25 %	17 %	15
24, 48	17 %	15 %	23
24, 81	42 %	43 %	23
32, 16	21 %	20 %	10
48, 6	14 %	25 %	2
48, 12	26 %	27 %	8
48, 16	21 %	22 %	10
Average	24 %	24 %	11

Illustration on two-stage stochastic problems

Test problems

- 9 families of public instances [I. Deàk]
- For each family, we considered a number of scenarios

$$N \in \{100, 200, 500, 800, 1000, 1200, 1500\}$$

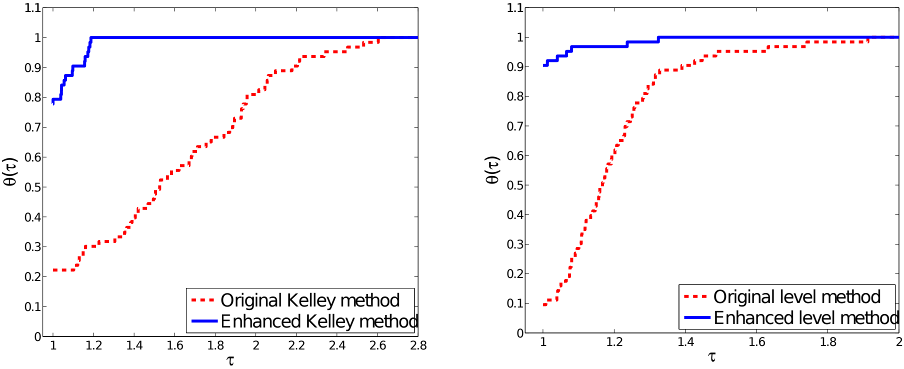
$\rightsquigarrow$ 63 problems

Our settings

- 2 oracles
  - **Fine oracle:** solving exactly the N scenarios subproblems
  - **Coarse oracle:** solving $\frac{N}{5}$ scenarios ($\rightsquigarrow$ about 5 times faster)
- 4 algorithms
  - **Kelley method**: standard and enhanced versions
  - **Level bundle method**: standard and enhanced versions

Synthetic results on two stage stochastic problems

Performance profiles for Kelley and level bundle variants



$$\left(\theta_A(\tau) := \frac{\text{nb of pbs solved by algo. } A \text{ within } \tau \times \text{ the best time}}{\text{total nb of pbs}}\right)$$

Conclusions

Summary

- The idea: **exploiting cheap or free uncontrolled information**
  - Situation 1: approximation given by some (good) heuristic
  - Situation 2: free approximates computed during exact evaluation
- **General scheme** to make use of coarse oracles in bundle-like methods
- **Two successful instances**: Kelley method and level bundle method
  - proof of convergence
  - acceleration of the algorithms

Ongoing work

- Level bundle variant to unit-commitment problem

Conclusions

Summary

- The idea: **exploiting cheap or free uncontrolled information**
  - Situation 1: approximation given by some (good) heuristic
  - Situation 2: free approximates computed during exact evaluation
- **General scheme** to make use of coarse oracles in bundle-like methods
- **Two successful instances**: Kelley method and level bundle method
  - proof of convergence
  - acceleration of the algorithms

Ongoing work

- Level bundle variant to unit-commitment problem

Thank you!

Improvements for Kelley method

	CPU time reduction (%)			Exact calls reduction (%)		
Pb family	min	avg	max	min	avg	max
F1	31	44	54	56	79	87
F2	12	35	48	62	73	75
F3	11	33	54	57	75	90
F4	26	47	60	52	81	90
F5	40	50	61	71	83	87
F6	9	32	47	51	77	87
F7	-18	12	48	47	71	96
F8	-18	-4	36	38	63	94
F9	-17	1	38	38	60	91
Total	8	28	50	52	74	88

Table : Our Kelley method using the two oracles vs.usual Kelley method

Improvements for level bundle method

	CPU time reduction (%)			Exact calls reduction (%)		
Pb family	min	avg	max	min	avg	max
F1	23	31	51	42	53	72
F2	-5	15	38	16	42	66
F3	5	20	30	17	33	41
F4	10	27	45	33	42	50
F5	19	29	39	40	49	57
F6	8	15	28	20	25	35
F7	11	19	31	15	30	42
F8	0	14	33	8	24	44
F9	-6	5	17	10	18	24
Total	7	19	35	22	35	48

Table : Our level method using two oracles vs usual level bundle method