Inexact Oracles in NonDifferentiable Optimization: Deflected Conditional Subgradient Methods and Generalized Bundle Methods

Antonio Frangioni

Dipartimento di Informatica, Università di Pisa

48th Workshop on Nonsmooth Analysis Optimization and Applications
Dedicated to V.F. Demyanov
Erice, May 14, 2008
Introduction, Motivation
Outline

1. Introduction, Motivation
2. Subgradient methods: introduction
Outline

1. Introduction, Motivation
2. Subgradient methods: introduction
3. Polyak-type stepsize: the abstract case
1. Introduction, Motivation

2. Subgradient methods: introduction

3. Polyak-type stepsize: the abstract case

4. Polyak-type stepsize: the implementable case
Outline

1. Introduction, Motivation
2. Subgradient methods: introduction
3. Polyak-type stepsize: the abstract case
4. Polyak-type stepsize: the implementable case
5. Deflection-restricted rules
Outline

1. Introduction, Motivation
2. Subgradient methods: introduction
3. Polyak-type stepsize: the abstract case
4. Polyak-type stepsize: the implementable case
5. Deflection-restricted rules
6. Bundle methods
1. Introduction, Motivation

2. Subgradient methods: introduction

3. Polyak-type stepsize: the abstract case

4. Polyak-type stepsize: the implementable case

5. Deflection-restricted rules

6. Bundle methods

7. Conclusions
Lagrangian Relaxation

- Difficult structured problem

\[ z(P) = \sup_u \{ c(u) : h(u) \leq 0, \; u \in U \} \tag{1} \]

with complicating constraints \( h(u) \leq 0 \) over easy set \( U \)

---

1Lemaréchal, Renaud “A geometric study of duality gaps, with applications”, Math. Prog., 2001
Lagrangian Relaxation

- **Difficult structured problem**
  \[ z(P) = \sup_u \{ c(u) : h(u) \leq 0, \ u \in U \} \]  
  with complicating constraints \( h(u) \leq 0 \) over easy set \( U \)

- **Assume Lagrangian relaxation** of complicating constraints **easy**
  \[ f(x) = \sup_u \{ c(u) + xh(u) : u \in U \} \]  

---

Lagrangian Relaxation

- Difficult **structured** problem

\[
z(P) = \sup_u \left\{ c(u) : h(u) \leq 0, \ u \in U \right\}
\]  
(1)

with **complicating constraints** \( h(u) \leq 0 \) over **easy set** \( U \)

- Assume **Lagrangian relaxation** of complicating constraints **easy**

\[
f(x) = \sup_u \left\{ c(u) + xh(u) : u \in U \right\}
\]  
(2)

- \( f \) **convex** \( \Rightarrow \) corresponding **Lagrangian dual easy**

\[
z(\Pi) = \inf_x \left\{ f(x) : x \geq 0 \right\}
\]

---

Lagrangian Relaxation

- **Difficult structured problem**

\[
z(P) = \sup_u \{ c(u) : h(u) \leq 0 , \ u \in U \}
\]

with **complicating constraints** \(h(u) \leq 0\) over **easy set** \(U\)

- **Assume Lagrangian relaxation of complicating constraints easy**

\[
f(x) = \sup_u \{ c(u) + xh(u) : u \in U \}
\]

- \(f\) convex \(\Rightarrow\) corresponding **Lagrangian dual easy**

\[
z(\Pi) = \inf_x \{ f(x) : x \geq 0 \}
\]

- **Equivalent to primal relaxation**

\[
\sup \{ \nu : (u, \nu, 0) \in U^{**} \}
\]

where \(U = \{ (u, \nu, r) : u \in U , \ \nu \leq c(u) , \ r \geq h(u) \}\)

(a more palatable object if problem “affine enough”)

---

\[^1\]Lemaréchal, Renaud “A geometric study of duality gaps, with applications”, Math. Prog., 2001
Lagrangian Relaxation (graphically)
Lagrangian Relaxation (graphically)
Lagrangian Relaxation (graphically)

- **Oracle** to (efficiently) perform the maximization (structure inside)
Oracle to (efficiently) perform the maximization (structure inside)

Solving exactly (2) provides both function value and subgradient
Primal “continuous” solutions useful to drive heuristics for (1)²

Lagrangian Relaxation: What For?

1. **Primal “continuous” solutions useful to drive heuristics for (1)**

2. **Mainly upper bounding**: $z(\Pi) \geq z(P)$, “near” if (2) “not too easy”

   $\Rightarrow$ safe (and effective) stopping criterion

---

Primal “continuous” solutions useful to drive heuristics for (1)\(^2\)

Mainly **upper bounding**: \(z(\Pi) \geq z(P)\), “near” if (2) “not too easy”
⇒ safe (and effective) stopping criterion

**Trade off**: “difficult” (2) ⇒ “good bound”\(^3\)

---

\(^2\)F., Gentile, Lacalandra “Solving Unit Commitment Problems with General Ramp Contraints”, IJEPES, 2008

1. Primal “continuous” solutions useful to drive heuristics for (1)²

2. Mainly upper bounding: \( z(\Pi) \geq z(P) \), “near” if (2) “not too easy”
   \[ \Rightarrow \] safe (and effective) stopping criterion

- **Trade off:** “difficult” (2) \( \Rightarrow \) “good bound”³

- **Enumerative approaches:** do this at each of very many nodes

---

Lagrangian Relaxation: What For?

1. Primal “continuous” solutions useful to drive heuristics for (1)²

2. Mainly upper bounding: \( z(\Pi) \geq z(P) \), “near” if (2) “not too easy”

\[ \Rightarrow \text{safe (and effective) stopping criterion} \]

- **Trade off**: “difficult” (2) \[ \Rightarrow \text{“good bound”} \]³

- **Enumerative approaches**: do this at each of very many nodes

- (\( \Pi \)) has to be (approximately) solved very efficiently =
  fast convergence + low iteration cost

---


Primal “continuous” solutions useful to drive heuristics for (1)\(^2\)

Mainly upper bounding: \(z(\Pi) \geq z(P)\), “near” if (2) “not too easy”
⇒ safe (and effective) stopping criterion

Trade off: “difficult” (2) ⇒ “good bound”\(^3\)

Enumerative approaches: do this at each of very many nodes

(\(\Pi\)) has to be (approximately) solved very efficiently =
fast convergence + low iteration cost

It thus makes sense to solve (2) approximately

\(^2\)F., Gentile, Lacalandra “Solving Unit Commitment Problems with General Ramp Contraints”, IJEPES, 2008

Lagrangian Relaxation: What For?

1. Primal “continuous” solutions useful to drive heuristics for (1)

2. Mainly upper bounding: $z(\Pi) \geq z(P)$, “near” if (2) “not too easy”
   ⇒ safe (and effective) stopping criterion

- Trade off: “difficult” (2) ⇒ “good bound”

- Enumerative approaches: do this at each of very many nodes

- $(\Pi)$ has to be (approximately) solved very efficiently =
  fast convergence + low iteration cost

- It thus makes sense to solve (2) approximately

- Which may mean different things

---


Approximate Lagrangian Relaxation I (graphically)
Approximate solution $\Rightarrow$ $\sigma$-subgradient, $\sigma \geq 0$
Approximate solution \( \Rightarrow \sigma\)-subgradient, \( \sigma \geq 0 \)

Heuristics \( \Rightarrow \) no measure of \( \sigma \) \( \Rightarrow \) useless for bounding purposes
Heuristics have no (or too weak in practice) performance guarantee

Different approach: an exact algorithm for solving (2)

---

Beltran, Tadonki, Vial “Solving the p-Median Problem with a Semi-Lagrangian Relaxation”, COAP, 2006
Heuristics have no (or too weak in practice) performance guarantee.

Different approach: an exact algorithm for solving (2)

Three main components:
- A heuristic producing $\bar{u} \in U \Rightarrow c(\bar{u}) + xh(\bar{u}) \leq f(x)$
- An upper bound $\bar{f}(x) \geq f(x)$ (further relaxation)
- Enumeration to squeeze the two together (branching)

Iterative process where $c(\bar{u}) + xh(\bar{u}) \rightarrow f(x) \leftarrow \bar{f}(x)$

---

Beltran, Tadonki, Vial “Solving the p-Median Problem with a Semi-Lagrangian Relaxation”, COAP, 2006
Approximate Lagrangian Relaxation II

- Heuristics have no (or too weak in practice) performance guarantee

- Different approach: an exact algorithm for solving (2)

- Three main components:
  - a heuristic producing $\bar{u} \in U \Rightarrow c(\bar{u}) + xh(\bar{u}) \leq f(x)$
  - an upper bound $\bar{f}(x) \geq f(x)$ (further relaxation)
  - enumeration to squeeze the two together (branching)

- Iterative process where $c(\bar{u}) + xh(\bar{u}) \rightarrow f(x) \leftarrow \bar{f}(x)$

- (2) “as difficult” as (1) in theory (but largely less so in practice$^4$)

- The gap $\sigma = \bar{f}(x) - c(\bar{u}) - xh(\bar{u}) \geq 0$ may decrease rather slowly

Heuristics have no (or too weak in practice) performance guarantee.

Different approach: an exact algorithm for solving (2)

Three main components:
- A heuristic producing $\bar{u} \in U \Rightarrow c(\bar{u}) + xh(\bar{u}) \leq f(x)$
- An upper bound $\bar{f}(x) \geq f(x)$ (further relaxation)
- Enumeration to squeeze the two together (branching)

Iterative process where $c(\bar{u}) + xh(\bar{u}) \to f(x) \leftarrow \bar{f}(x)$

(2) “as difficult” as (1) in theory (but largely less so in practice\textsuperscript{4})

The gap $\sigma = \bar{f}(x) - c(\bar{u}) - xh(\bar{u}) \geq 0$ may decrease rather slowly

For bounding purposes, $\bar{f}(x)$ “is” $f(x)$

\textsuperscript{4}Beltran, Tadonki, Vial “Solving the p-Median Problem with a Semi-Lagrangian Relaxation”, COAP, 2006
Approximate Lagrangian Relaxation II (graphically)
Approximate Lagrangian Relaxation II (graphically)
The upper bound $\bar{f}(x)$ "is" the function value
The upper bound $\bar{f}(x)$ "is" the function value

$\sigma$ decreases if either $\bar{f}(x)$ decreases or $c(\bar{u}) + xh(\bar{u})$ increases
A Somewhat Different (but related) Case

- The decomposable case:
  \[ u = (u^1, \ldots, u^k) \in U^1 \times \ldots \times U^k \]
  \[ c(u) = c^1(u^1) + \ldots + c^k(u^k) \]
  \[ h(u) = h^1(u^1) + \ldots + h^k(u^k) \]

- Computing \( f(x) \) decomposes into \( k \) independent subproblems

\[ ^5 \text{Ned}^\prime \text{c}, \text{Bertsekas “Incremental subgradient methods for nondifferentiable optimization”, SIOPT, 2001} \]
A Somewhat Different (but related) Case

- The decomposable case:
  \[ u = (u^1, \ldots, u^k) \in U^1 \times \ldots \times U^k \]
  \[ c(u) = c^1(u^1) + \ldots + c^k(u^k) \]
  \[ h(u) = h^1(u^1) + \ldots + h^k(u^k) \]

- Computing \( f(x) \) decomposes into \( k \) independent subproblems

- In some cases, the problems are “easy” but they are “many”

- Avoid computing them all for each \( x \), at least at some iterations \(^5\)

---

\(^5\) Nedíc, Bertsekas “Incremental subgradient methods for nondifferentiable optimization”, SIOPT, 2001
A Somewhat Different (but related) Case

- The decomposable case:
  \[ u = (u^1, \ldots, u^k) \in U^1 \times \ldots \times U^k \]
  \[ c(u) = c^1(u^1) + \ldots + c^k(u^k) \]
  \[ h(u) = h^1(u^1) + \ldots + h^k(u^k) \]

- Computing \( f(x) \) decomposes into \( k \) independent subproblems

- In some cases, the problems are “easy” but they are “many”

- Avoid computing them all for each \( x \), at least at some iterations \(^5\)

- Something like: lower bound always available, upper bound only available if all \( k \) problems are solved

\(^5\)Nedíč, Bertsekas “Incremental subgradient methods for nondifferentiable optimization”, SIOPT, 2001
A Somewhat Different (but related) Case

- The decomposable case:
  \[ u = (u^1, \ldots, u^k) \in U^1 \times \ldots \times U^k \]
  \[ c(u) = c^1(u^1) + \ldots + c^k(u^k) \]
  \[ h(u) = h^1(u^1) + \ldots + h^k(u^k) \]

- Computing \( f(x) \) decomposes into \( k \) independent subproblems

- In some cases, the problems are “easy” but they are “many”

- Avoid computing them all for each \( x \), at least at some iterations\(^5\)

- Something like: lower bound always available, upper bound only available if all \( k \) problems are solved

- Alternatively: \( \bar{f}(x) \) is either \(+\infty\) or \( f(x) \)

---

\(^5\)Nedíc, Bertsekas “Incremental subgradient methods for nondifferentiable optimization”, SIOPT, 2001
A Somewhat Different (but related) Case

- The decomposable case:

  \[ u = (u^1, \ldots, u^k) \in U^1 \times \ldots \times U^k \]

  \[ c(u) = c^1(u^1) + \ldots + c^k(u^k) \]

  \[ h(u) = h^1(u^1) + \ldots + h^k(u^k) \]

- Computing \( f(x) \) decomposes into \( k \) independent subproblems

- In some cases, the problems are “easy” but they are “many”

- Avoid computing them all for each \( x \), at least at some iterations \(^5\)

- Something like: lower bound always available, upper bound only available if all \( k \) problems are solved

- Alternatively: \( \bar{f}(x) \) is either \( +\infty \) or \( f(x) \)

- Then, of course, each subproblem can be solved approximately

\(^5\)Nedíc, Bertsekas “Incremental subgradient methods for nondifferentiable optimization”, SIOPT, 2001
The Issue

- Minimizing $f$ using a approximated subgradient (≡ oracle) possible \[6\]

---

7 Kiwiel “Convergence of approximate and incremental subgradient methods for convex minimization”, SIOPT, 2004
8 Kiwiel “A proximal bundle method with approximate subgradient linearizations”, SIOPT, 2006
The Issue

- Minimizing $f$ using an approximated subgradient (= oracle) possible

- Lately, the standard has been “nothing is known about $\sigma$”

---

7 Kiwiel “Convergence of approximate and incremental subgradient methods for convex minimization”, SIOPT, 2004
8 Kiwiel “A proximal bundle method with approximate subgradient linearizations”, SIOPT, 2006
Minimizing $f$ using a approximated subgradient (≈ oracle) possible \(^6\)

Lately, the standard has been “nothing is known about $\sigma$” \(^7\) \(^8\) \(^9\)

But in practice, $\sigma$ is known (if we accept that $\bar{f}(x)$ “is” $f(x)$)

---


\(^7\) Kiwiel “Convergence of approximate and incremental subgradient methods for convex minimization”, SIOPT, 2004

\(^8\) Kiwiel “A proximal bundle method with approximate subgradient linearizations”, SIOPT, 2006

The Issue

- Minimizing $f$ using a approximated subgradient (≡ oracle) possible

- Lately, the standard has been “nothing is known about $\sigma$”

- But in practice, $\sigma$ is known (if we accept that $\bar{f}(x)$ “is” $f(x)$)

- The issue:

  Does knowing $\sigma$ help in (approximately) minimizing $f$?

---

7 Kiwiel “Convergence of approximate and incremental subgradient methods for convex minimization”, SIOPT, 2004
8 Kiwiel “A proximal bundle method with approximate subgradient linearizations”, SIOPT, 2006
Minimizing $f$ using a approximated subgradient (≡ oracle) possible

Lately, the standard has been “nothing is known about $\sigma$”

But in practice, $\sigma$ is known (if we accept that $\tilde{f}(x)$ “is” $f(x)$)

The issue:

Does knowing $\sigma$ help in (approximately) minimizing $f$?

Of course, it depends on what approach is used

---


7 Kiwiel “Convergence of approximate and incremental subgradient methods for convex minimization”, SIOPT, 2004

8 Kiwiel “A proximal bundle method with approximate subgradient linearizations”, SIOPT, 2006

Subgradient Methods

(with Giacomo d’Antonio)
General problem:

$$\inf_x \{ f(x) : x \in X \}$$

$$f : \mathbb{R}^n \to \mathbb{R} \text{ convex} = \text{approximated oracle}, \ X \subseteq \mathbb{R}^n \text{ closed convex}$$
General problem:
\[
\inf_x \{ f(x) : x \in X \}
\]
\(f : \mathbb{R}^n \to \mathbb{R}\) convex = approximated oracle, \(X \subseteq \mathbb{R}^n\) closed convex

Basic approximate subgradient method:
\[g_k \in \partial_{\sigma_k} f(x_k) \quad , \quad \hat{x}_{k+1} = x_k - \nu_k g_k \quad , \quad x_{k+1} = P_X(\hat{x}_{k+1})\]
\(P_X = \) orthogonal projection on \(X\) (assumed “cheap”), \(\nu_k\) stepsize
General problem:

$$\inf_x \{ f(x) : x \in X \}$$

$$f : \mathbb{R}^n \to \mathbb{R} \text{ convex} = \text{approximated oracle, } X \subseteq \mathbb{R}^n \text{ closed convex}$$

Basic approximate subgradient method:

$$g_k \in \partial_{\sigma_k} f(x_k) , \quad \hat{x}_{k+1} = x_k - \nu_k g_k , \quad x_{k+1} = P_X(\hat{x}_{k+1})$$

$$P_X = \text{orthogonal projection on } X \text{ (assumed "cheap"), } \nu_k \text{ stepsize}$$

Very simple, almost no overhead w.r.t. $$f(x)$$ computation

Many variants (dilation methods, Bregman projections, ...)
(approximate) Subgradient Methods

- General problem:
  \[ \inf_x \{ f(x) : x \in X \} \]
  \( f : \mathbb{R}^n \to \mathbb{R} \) convex = approximated oracle, \( X \subseteq \mathbb{R}^n \) closed convex

- Basic approximate subgradient method:
  \( g_k \in \partial_{\sigma_k} f(x_k) \), \( \hat{x}_{k+1} = x_k - \nu_k g_k \), \( x_{k+1} = P_X(\hat{x}_{k+1}) \)
  \( P_X \) = orthogonal projection on \( X \) (assumed “cheap”), \( \nu_k \) stepsize

- Very simple, almost no overhead w.r.t. \( f(x) \) computation

- Many variants (dilation methods, Bregman projections, ...)

- Typically rather slow, because:
  - a \((1 - \varepsilon)\)th-order method, cannot be fast
  - zig-zagging I: in “deep and narrow valleys”, successive subgradients almost orthogonal to each other
  - zig-zagging II: at \( \partial X \), subgradients almost orthogonal to \( \partial X \)
Zig-Zagging I

Two long steps . . .

---

Two long steps ... are one short step
Two long steps . . . are one short step

Solution: use previous direction
Zig-Zagging I

- Two long steps ... are one short step
- Solution: use previous direction to deflect $g_k$ (e.g. $\rightarrow d_k d_{k-1} \geq 0$)\(^{10}\)

\(^{10}\)Camerini, Fratta, Maffioli “On Improving Relaxation Methods by Modified Gradient Techniques”, Math. Prog., 1975
Two long steps ... are one short step

Solution: use previous direction to deflect $g_k$ (e.g. $\rightarrow d_k d_{k-1} \geq 0$)

---

Projecting a long step ...
Projecting a long step . . . may result in a short step
Projecting a long step ... may result in a short step

Solution: project $g^k$ onto the tangent cone at $x^k$
Projecting a long step . . . may result in a short step

Solution: project $g^k$ onto the tangent cone at $x^k$ . . . or, equivalently, deflect using $-z^k \in \partial l_X(x^k) \rightarrow d_k \in \partial f_X(x^k)$ ($f_X = f + l_X$)
Projecting a long step ... may result in a short step

Solution: project $g^k$ onto the tangent cone at $x^k$ ... or, equivalently, deflect using $-z^k \in \partial I_X(x^k) \rightarrow d_k \in \partial f_X(x^k)$ ($f_X = f + l_X$)
Two Classes of Subgradient Methods

- **Conditional subgradient**: \( d_k = -P_{\nabla f_X(x_k)}(-g_k)^{11} \in \partial f_X(x^k) \)

---

12 Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
13 Guta “Subgradient Optimization Methods . . . with an Application to a Radiation Therapy Problem”, Ph.D., 2003
15 F., Lodi, Rinaldi “New Approaches for Optimizing over the Semimetric Polytope”, Math. Prog., 2005
Two Classes of Subgradient Methods

- **Conditional subgradient**: \( d_k = -P_{T_X(x_k)}(-g_k)^{11} \in \partial f_X(x^k) \)

- **Deflected subgradient**: \( d_k = g_k + \eta_k d_{k-1} \)

---

12 Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
13 Guta “Subgradient Optimization Methods . . . with an Application to a Radiation Therapy Problem”, Ph.D., 2003
15 F., Lodi, Rinaldi “New Approaches for Optimizing over the Semimetric Polytope”, Math. Prog., 2005
Two Classes of Subgradient Methods

- **Conditional subgradient:** \( d_k = -P_{T_x(x_k)}(-g_k) \in \partial f_X(x^k) \)

- **Deflected subgradient:** \( d_k = g_k + \eta_k d_{k-1} \) ... better, w.l.o.g.

\[
 d_k = \alpha_k g_k + (1 - \alpha_k) d_{k-1}, \quad \alpha_k \in [0, 1]
\]

(the missing scaling factor can always be attached to \( \nu_k \))

---

12 Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
13 Guta “Subgradient Optimization Methods ... with an Application to a Radiation Therapy Problem”, Ph.D., 2003
14 Crainic, F., Gendron “Bundle-based Relaxation Methods for Multicommodity ... Network Design”, DAM, 2001
15 F., Lodi, Rinaldi “New Approaches for Optimizing over the Semimetric Polytope”, Math. Prog., 2005
Two Classes of Subgradient Methods

- **Conditional subgradient**: \( d_k = -P_{T_X(x_k)}(-g_k) \in \partial f_X(x^k) \)

- **Deflected subgradient**: \( d_k = g_k + \eta_k d_{k-1} \) ... better, w.l.o.g.

\[
d_k = \alpha_k g_k + (1 - \alpha_k)d_{k-1}, \quad \alpha_k \in [0, 1]
\]

(the missing scaling factor can always be attached to \( \nu_k \))

- Funnily enough, (almost) no conditional deflected subgradient

---

12. Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
Two Classes of Subgradient Methods

- **Conditional subgradient**: \( d_k = -P_{T_X(x_k)}(-g_k) \) \( \in \partial f_X(x^k) \)

- **Deflected subgradient**: \( d_k = g_k + \eta_k d_{k-1} \) . . . better, w.l.o.g.

\[
d_k = \alpha_k g_k + (1 - \alpha_k) d_{k-1} \quad , \quad \alpha_k \in [0, 1]
\]

(the missing scaling factor can always be attached to \( \nu_k \) )

- Funnily enough, (almost) no conditional deflected subgradient

- Besides: **conditional approximate subgradient**, yes

---

12 Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
13 Guta “Subgradient Optimization Methods . . . with an Application to a Radiation Therapy Problem”, Ph.D., 2003
15 F., Lodi, Rinaldi “New Approaches for Optimizing over the Semimetric Polytope”, Math. Prog., 2005
Two Classes of Subgradient Methods

- Conditional subgradient: \( d_k = -P_{T_{X}(x_k)}(-g_k) \quad \in \partial f_{X}(x^k) \)

- Deflected subgradient: \( d_k = g_k + \eta_k d_{k-1} \) ... better, w.l.o.g.

\[
d_k = \alpha_k g_k + (1 - \alpha_k) d_{k-1} , \quad \alpha_k \in [0, 1]
\]

(the missing scaling factor can always be attached to \( \nu_k \) )

- Funnily enough, (almost) no conditional deflected subgradient

- Besides: conditional approximate subgradient, yes

  ... but deflected approximate subgradient, no.

---

12 Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
13 Guta “Subgradient Optimization Methods ... with an Application to a Radiation Therapy Problem”, Ph.D., 2003
14 Crainic, F., Gendron “Bundle-based Relaxation Methods for Multicommodity ... Network Design”, DAM, 2001
15 F., Lodi, Rinaldi “New Approaches for Optimizing over the Semimetric Polytope”, Math. Prog., 2005
Two Classes of Subgradient Methods

- **Conditional subgradient**: $d_k = -P_{T_X(x_k)}(-g_k) \in \partial f_X(x_k)$

- **Deflected subgradient**: $d_k = g_k + \eta_k d_{k-1}$ ... better, w.l.o.g.

  $d_k = \alpha_k g_k + (1 - \alpha_k) d_{k-1}$, $\alpha_k \in [0, 1]$

  (the missing scaling factor can always be attached to $\nu_k$)

- Funnily enough, (almost) no conditional deflected subgradient

- Besides: conditional approximate subgradient, yes

  ... but deflected approximate subgradient, no.

- Still there is need for good subgradient methods

---

12 Sherali, Lim “On Embedding the Volume Algorithm in a Variable Target Value Method”, ORL, 2004
13 Guta “Subgradient Optimization Methods ... with an Application to a Radiation Therapy Problem”, Ph.D., 2003
14 Crainic, F., Gendron “Bundle-based Relaxation Methods for Multicommodity ... Network Design”, DAM, 2001
15 F., Lodi, Rinaldi “New Approaches for Optimizing over the Semimetric Polytope”, Math. Prog., 2005
Why Conditional + Deflected is Not (entirely) Obvious

- Projecting ...
Projecting ... and then deflecting gives $d_{k+1} \notin T_x(x_k)$
Why Conditional + Deflected is Not (entirely) Obvious

- Projecting ... and then deflecting gives $d_{k+1} \notin T_X(x_k)$
- Solution: first deflect,
Why Conditional + Deflected is Not (entirely) Obvious

- Projecting ... and then deflecting gives $d_{k+1} \notin T_x(x_k)$
- Solution: first deflect, then project; now $d_{k+1} \in T_x(x_k)$
Why Conditional + Deflected is Not (entirely) Obvious

- Projecting . . . and then deflecting gives $d_{k+1} \notin T_{x}(x_{k})$
- Solution: first deflect, then project; now $d_{k+1} \in T_{x}(x_{k})$
Conditional Deflected (Approximate) Subgradient

\[ \hat{d}_k = \alpha_k \bar{g}_k + (1 - \alpha_k) \bar{d}_{k-1} \quad d_k = -P_{T_{x_k}}(-\hat{d}_k) \]

\[ \bar{g}_k = \text{either } g_k \text{ or } \hat{g}_k, \quad \bar{d}_k = \text{either } d_k \text{ or } \hat{d}_k \]

- Four different schemes, but unified treatment (\( \leq \text{two projections} \))
Conditional Deflected (Approximate) Subgradient

\[ \hat{d}_k = \alpha_k \bar{g}_k + (1 - \alpha_k)\bar{d}_{k-1} \quad d_k = -P_{T_X(x_k)}(-\hat{d}_k) \]

\[ \bar{g}_k = \text{either } g_k \text{ or } \hat{g}_k, \quad \bar{d}_k = \text{either } d_k \text{ or } \hat{d}_k \]

- Four different schemes, but unified treatment (\( \leq \text{two projections} \))
- Whatever the choice, \( \bar{g}_k \in \partial_{\sigma_k} f_X(x_k) \)
- Allows to unify some technical results, like

\[ \bar{d}_k (x - x_k) \leq \hat{d}_k (x - x_k) \]

(trivial if \( \bar{d}_k = \hat{d}_k \), but not otherwise), and

\[ \bar{d}_k (x_k - x_{k+1}) \leq \nu_k \|d_k\|^2 \]
\[ \hat{d}_k = \alpha_k \bar{g}_k + (1 - \alpha_k) \bar{d}_{k-1} \quad d_k = -P_{T_X(x_k)}(-\hat{d}_k) \]
\[ \bar{g}_k = \text{either } g_k \text{ or } \hat{g}_k, \quad \bar{d}_k = \text{either } d_k \text{ or } \hat{d}_k \]

- Four different schemes, but unified treatment (\( \leq \) two projections)

- Whatever the choice, \( \bar{g}_k \in \partial_{\sigma_k} f_X(x_k) \)

- Allows to unify some technical results, like
  \[ \bar{d}_k(x - x_k) \leq \hat{d}_k(x - x_k) \]
  (trivial if \( \bar{d}_k = \hat{d}_k \), but not otherwise), and
  \[ \bar{d}_k(x_k - x_{k+1}) \leq \nu_k \|d_k\|^2 \]

- Crucial result (relying on \( \alpha_k \in [0,1] \)): \( \bar{d}_k \in \partial_{\varepsilon_k} f_X(x_k) \) with
  \[ \varepsilon_k = (1 - \alpha_k) \left( f_k - f_{k-1} - \bar{d}_{k-1}(x_k - x_{k-1}) + \varepsilon_{k-1} \right) + \alpha_k \sigma_k \]  \hspace{1cm} (4)
1 Introduction, Motivation
2 Subgradient methods: introduction
3 Polyak-type stepsize: the abstract case
4 Polyak-type stepsize: the implementable case
5 Deflection-restricted rules
6 Bundle methods
7 Conclusions
(standard) Polyak Stepsize

- Standard Polyak stepsize (assuming $f^* = \inf_x f_X(x) > -\infty$)

\[
\nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2
\]
(standard) Polyak Stepsize

- Standard Polyak stepsize (assuming $f^* = \inf_x f(x) > -\infty$)

$$\nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2$$

- Abstract rule, as $f^*$ unknown in general
Standard Polyak stepsize (assuming $f^* = \inf_x f_X(x) > -\infty$)

$$\nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2$$

- Abstract rule, as $f^*$ unknown in general
- Technical (but somewhat conceptual) issue: $d_k$ can be 0
(standard) Polyak Stepsize

- Standard Polyak stepsize (assuming $f^* = \inf_x f_x(x) > -\infty$)
  \[ \nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2 \]

- Abstract rule, as $f^*$ unknown in general

- Technical (but somewhat conceptual) issue: $d_k$ can be 0

- Not an issue if $\sigma_k$ constant (e.g. $\sigma_k \equiv 0$) and no deflection
(standard) Polyak Stepsize

- Standard Polyak stepsize (assuming $f^* = \inf_x f(x) > -\infty$)
  
  $$\nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2$$

- Abstract rule, as $f^*$ unknown in general

- Technical (but somewhat conceptual) issue: $d_k$ can be 0

- Not an issue if $\sigma_k$ constant (e.g. $\sigma_k \equiv 0$) and no deflection

- “Technical” solution $\nu_k \|d_k\|^2 \leq \beta_k \lambda_k$ ($\lambda_k = f_k - f^*$), not enough
(standard) Polyak Stepsize

- Standard Polyak stepsize (assuming $f^* = \inf_x f_x(x) > -\infty$)
  \[ \nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2 \]

- Abstract rule, as $f^*$ unknown in general

- Technical (but somewhat conceptual) issue: $d_k$ can be 0

- Not an issue if $\sigma_k$ constant (e.g. $\sigma_k \equiv 0$) and no deflection

- “Technical” solution $\nu_k \|d_k\|^2 \leq \beta_k \lambda_k$ ($\lambda_k = f_k - f^*$), not enough

**Observation**

$\sigma^* = \limsup_{k \to \infty} \sigma_k < +\infty$ (asymptotic maximum error of the oracle); no subgradient method can attain error $< \sigma^*$ (if $f^* > -\infty$)
(standard) Polyak Stepsize

- Standard Polyak stepsize (assuming $f^* = \inf_x f(x) > -\infty$)

\[
\nu_k = \beta_k \frac{f_k - f^*}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq 2
\]

- Abstract rule, as $f^*$ unknown in general

- Technical (but somewhat conceptual) issue: $d_k$ can be 0

- Not an issue if $\sigma_k$ constant (e.g. $\sigma_k \equiv 0$) and no deflection

- “Technical” solution $\nu_k \|d_k\|^2 \leq \beta_k \lambda_k$ ($\lambda_k = f_k - f^*$), not enough

Observation

$\sigma^* = \limsup_{k \to \infty} \sigma_k < +\infty$ (asymptotic maximum error of the oracle);
no subgradient method can attain error $< \sigma^*$ (if $f^* > -\infty$)

Proof.

$\sigma_k \geq \sigma^*$ and $f(x_0) = f^* + \sigma^* \Rightarrow g_k$ can be 0 $\Rightarrow d_k = 0$: never moves!
Further requirement: $\beta_k \leq \alpha_k (\leq 1)$
Further requirement: $\beta_k \leq \alpha_k$ ($\leq 1$)

Main technical result (using (4)):

$$\varepsilon_k \leq (1 - \alpha_k)(f_k - f^*) + \bar{\sigma}_k$$

where

$$\bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k$$

($\alpha_1 = 1$ for “unreliability of past information”)

(5)
Further requirement: $\beta_k \leq \alpha_k (\leq 1)$

Main technical result (using (4)):

$$\varepsilon_k \leq (1 - \alpha_k)(f_k - f^*) + \bar{\sigma}_k$$

where

$$\bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k$$

($\alpha_1 = 1$ for “unreliability of past information”)

Technical corollary: for each $\bar{x} \in X$

$$d_k(\bar{x} - x_k) \leq \alpha_k (f^* - f_k) + [f(\bar{x}) - f^* + \bar{\sigma}_k]$$
Polyak Stepsize (cont. d)

- Further requirement: $\beta_k \leq \alpha_k (\leq 1)$

- Main technical result (using (4)):

$$
\varepsilon_k \leq (1 - \alpha_k)(f_k - f^*) + \bar{\sigma}_k \\
\bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k
$$

(5) (6)

($\alpha_1 = 1$ for “unreliability of past information”)

- Technical corollary: for each $\bar{x} \in X$

$$
d_k(\bar{x} - x_k) \leq \alpha_k (f^* - f_k) + [f(\bar{x}) - f^* + \bar{\sigma}_k]
$$

(7)

- “Exact” convergence result at hand\(^7\): $\sigma_k \equiv 0 \Rightarrow$

$$
\exists \xi \in [0, 1) \quad \varepsilon_k \leq \xi(2 - \beta_k)(f_k - f^*)/2
$$

$\Rightarrow \liminf_{k\to\infty} f_k = f^\infty \leq f^*$
What about the approximate case?
What about the approximate case?

“Asymptotic error”: \[ \limsup_{k \to \infty} \bar{\sigma}_k = \bar{\sigma}^* \leq \sigma^* \]
Polyak Stepsize: the Approximate Case

- What about the approximate case?
- "Asymptotic error": \( \limsup_{k \to \infty} \bar{\sigma}_k = \bar{\sigma}^* \leq \sigma^* \)
- For "Asymptotically non-deflected" method (\( \lim_{k \to \infty} \alpha_k = 1 \))
  \[
  f^\infty \leq f^* + \frac{2\sigma^*}{(2 - \sup_k \beta_k)}
  \]
- Error twice as large than "optimal", basically no deflection
Polyak Stepsize: the Approximate Case

- What about the approximate case?
- “Asymptotic error”: \( \limsup_{k \to \infty} \bar{\sigma}_k = \bar{\sigma}^* \leq \sigma^* \)
- For “Asymptotically non-deflected” method (\( \lim_{k \to \infty} \alpha_k = 1 \))
  \[
  f^\infty \leq f^* + \frac{2\sigma^*}{(2 - \sup \beta_k)}
  \]
- Error twice as large than “optimal”, basically no deflection

**Theorem**

*Without any assumption on deflection*

\[
 f^\infty \leq f^* + \frac{2\sigma^*}{\Gamma} \quad \text{where} \quad \Gamma = \inf_k 2\alpha_k - \beta_k \geq \beta^*
\]

- Deflecting is possible, but does not look a good idea
Polyak Stepsize: the Approximate Case

- What about the approximate case?

- “Asymptotic error”: \( \limsup_{k \to \infty} \bar{\sigma}_k = \bar{\sigma}^* \leq \sigma^* \)

- For “Asymptotically non-deflected” method (\( \lim_{k \to \infty} \alpha_k = 1 \))

  \[ f^\infty \leq f^* + \frac{2\sigma^*}{2 - \sup_k \beta_k} \]

- Error twice as large than “optimal”, basically no deflection

**Theorem**

*Without any assumption on deflection*

\[ f^\infty \leq f^* + \frac{2\sigma^*}{\Gamma} \]

where \( \Gamma = \inf_k 2\alpha_k - \beta_k \geq \beta^* \)

- Deflecting is possible, but does not look a good idea

- However, knowing \( \sigma_k \) we can do better than that
**Corrected Polyak Stepsize**

- **Corrected** Polyak stepsize: $\lambda_k = f_k - f^* - \sigma_k$

$$0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \quad (8)$$
Corrected Polyak Stepsize

- **Corrected** Polyak stepsize: \( \lambda_k = f_k - f^* - \sigma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2} , \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1
\]

- **Issue:** \( \sigma_k > f_k - f^* \Rightarrow \lambda_k < 0 \). **Solution:**

\[
0 \leq \nu_k \|d_k\|^2 \leq \beta_k \lambda_k , \quad 0 \leq \beta_k \leq \alpha_k \leq 1
\]

which implies \( \lambda_k < 0 \Rightarrow \beta_k = 0 \Rightarrow \nu_k = 0 \) *(loops!)*
Corrected Polyak Stepsize

- **Corrected** Polyak stepsize: \( \lambda_k = f_k - f^* - \sigma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1
\]  

(8)

- **Issue:** \( \sigma_k > f_k - f^* \Rightarrow \lambda_k < 0 \). **Solution:**

\[
0 \leq \nu_k \|d_k\|^2 \leq \beta_k \lambda_k, \quad 0 \leq \beta_k \leq \alpha_k \leq 1
\]

which implies \( \lambda_k < 0 \Rightarrow \beta_k = 0 \Rightarrow \nu_k = 0 \) (*loops!*)

- In plain words: if the error is too large, stop until it decreases enough
Corrected Polyak Stepsize

- **Corrected** Polyak stepsize: \( \lambda_k = f_k - f^* - \sigma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1
\]  

(8)

- **Issue:** \( \sigma_k > f_k - f^* \Rightarrow \lambda_k < 0 \). **Solution:**

\[
0 \leq \nu_k \|d_k\|^2 \leq \beta_k \lambda_k, \quad 0 \leq \beta_k \leq \alpha_k \leq 1
\]

which implies \( \lambda_k < 0 \Rightarrow \beta_k = 0 \Rightarrow \nu_k = 0 \) (**loops!**)

- **In plain words:** if the error is too large, stop until it decreases enough

- **Actually,** a slightly stronger form is required:

\[
\lambda_k \geq 0 \Rightarrow (\alpha_k \geq) \beta_k \geq \beta^* > 0, \quad \lambda_k < 0 \Rightarrow \alpha_k = 0 (\Rightarrow \beta_k = 0)
\]
Corrected Polyak Stepsize

- **Corrected** Polyak stepsize: \( \lambda_k = f_k - f^* - \sigma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \tag{8}
\]

- **Issue**: \( \sigma_k > f_k - f^* \Rightarrow \lambda_k < 0 \). **Solution**:\[
0 \leq \nu_k \|d_k\|^2 \leq \beta_k \lambda_k, \quad 0 \leq \beta_k \leq \alpha_k \leq 1
\]

which implies \( \lambda_k < 0 \Rightarrow \beta_k = 0 \Rightarrow \nu_k = 0 \) (loops!)

- In plain words: if the error is too large, stop until it decreases enough

- Actually, a slightly stronger form is required:

\[
\lambda_k \geq 0 \Rightarrow (\alpha_k \geq \beta_k \geq \beta^* > 0), \\
\lambda_k < 0 \Rightarrow \alpha_k = 0 (\Rightarrow \beta_k = 0)
\]

(8) \( \Rightarrow \) (5) + (7) with \( \bar{\sigma}_k = \alpha_k \sigma_k \);  

good deflecting “shaves away” a part of the error
Corrected Polyak Stepsize

- Without any assumption on deflection: $(8) \Rightarrow$
  - $f^\infty \leq f^* + \sigma^*$
  - $X^* \neq \emptyset \Rightarrow \exists$ subsequence $\{x_{k_i}\} \rightarrow x^\infty \in X$ s.t. $f(x^\infty) = f^\infty$
  - $X^* \neq \emptyset$ & $\sigma^* = 0 \Rightarrow$ the whole $\{x_k\} \rightarrow x^* \in X^*$
Without any assumption on deflection: (8) ⇒
- \( f^\infty \leq f^* + \sigma^* \)
- \( X^* \neq \emptyset \) ⇒ ∃ subsequence \( \{x_k\} \rightarrow x^\infty \in X \) s.t. \( f(x^\infty) = f^\infty \)
- \( X^* \neq \emptyset \) & \( \sigma^* = 0 \) ⇒ the whole \( \{x_k\} \rightarrow x^* \in X^* \)

Better result than the available ones:\(^7\):
- **Optimal** error attained even in inexact case
- Convergence of the iterates (in the exact case)
- Deflection does not worsen results
Corrected Polyak Stepsize

- Without any assumption on deflection: (8) \( \Rightarrow \)
  - \( f^\infty \leq f^* + \sigma^* \)
  - \( X^* \neq \emptyset \Rightarrow \exists \text{ subsequence} \{ x_{k_i} \} \to x^\infty \in X \text{ s.t.} \ f(x^\infty) = f^\infty \)
  - \( X^* \neq \emptyset \& \sigma^* = 0 \Rightarrow \text{the whole} \{ x_k \} \to x^* \in X^* \)

- Better result than the available ones\(^7\):
  - **Optimal** error attained even in inexact case
  - Convergence of the iterates (in the exact case)
  - Deflection does not worsen results

- Interesting detail of the proof:
  - some things only hold if \( \lambda_k \geq 0 \) for *infinitely many* \( k \),
Without any assumption on deflection: (8) \( \Rightarrow \)

- \( f^\infty \leq f^* + \sigma^* \)
- \( X^* \neq \emptyset \Rightarrow \exists \) subsequence \( \{x_{k_i}\} \rightarrow x^\infty \in X \text{ s.t. } f(x^\infty) = f^\infty \)
- \( X^* \neq \emptyset \) & \( \sigma^* = 0 \Rightarrow \) the whole \( \{x_k\} \rightarrow x^* \in X^* \)

Better result than the available ones:\(^7\):

- **Optimal** error attained even in inexact case
- Convergence of the iterates (in the exact case)
- Deflection does not worsen results

Interesting detail of the proof:

some things only hold if \( \lambda_k \geq 0 \) for *infinitely many* \( k \), which does not necessarily happen
Corrected Polyak Stepsize

Without any assumption on deflection: (8) ⇒
- \( f^\infty \leq f^* + \sigma^* \)
- \( X^* \neq \emptyset \Rightarrow \exists \) subsequence \( \{x_{k_i}\} \to x^\infty \in X \) s.t. \( f(x^\infty) = f^\infty \)
- \( X^* \neq \emptyset \) & \( \sigma^* = 0 \Rightarrow \) the whole \( \{x_k\} \to x^* \in X^* \)

Better result than the available ones\(^7\):
- **Optimal** error attained even in inexact case
- Convergence of the iterates (in the exact case)
- Deflection does not worsen results

Interesting detail of the proof:
some things only hold if \( \lambda_k \geq 0 \) for *infinitely many* \( k \),
which does not necessarily happen
but if not, a \( \sigma^* \)-optimal solution is finitely attained
Corrected Polyak Stepsize

- Without any assumption on deflection: (8) \( \Rightarrow \)
  - \( f^\infty \leq f^* + \sigma^* \)
  - \( X^* \neq \emptyset \Rightarrow \exists \) subsequence \( \{x_{k_i}\} \rightarrow x^\infty \in X \) s.t. \( f(x^\infty) = f^\infty \)
  - \( X^* \neq \emptyset \) \& \( \sigma^* = 0 \Rightarrow \) the whole \( \{x_k\} \rightarrow x^* \in X^* \)

- Better result than the available ones\(^7\):
  - **Optimal** error attained even in inexact case
  - Convergence of the iterates (in the exact case)
  - Deflection does not worsen results

- Interesting detail of the proof:
  - some things only hold if \( \lambda_k \geq 0 \) for *infinitely many* \( k \), which does not necessarily happen
  - but if not, a \( \sigma^* \)-optimal solution is finitely attained

- Potential issue: **exact** knowledge of \( \sigma_k \) required
The general form: $\lambda_k = f_k - f^* - \gamma_k$

$$0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1$$ (9)
The general form: \( \lambda_k = f_k - f^* - \gamma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \tag{9}
\]

(9) \(\Rightarrow\) (5) + (7) with \( \bar{\sigma}_k = (1 - \alpha_k)(\bar{\sigma}_{k-1} - \alpha_{k-1}\gamma_{k-1}) + \alpha_k \sigma_k \)
Generalized Corrected Polyak Stepsize

- The general form: $\lambda_k = f_k - f^* - \gamma_k$

  $$0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \quad (9)$$

- $(9) \Rightarrow (5) + (7)$ with $\bar{\sigma}_k = (1 - \alpha_k)(\bar{\sigma}_{k-1} - \alpha_{k-1}\gamma_{k-1}) + \alpha_k\sigma_k$

- General convergence:

  $$f^\infty \leq f^* + 2\Delta / \Gamma$$

  $$\Delta = \sigma^* + \tilde{\gamma}(\frac{1 - \beta^*}{\beta^*} + \sup_k \alpha_k / 2)$$

  $$\tilde{\gamma} = - \min \left\{ \gamma^* = \lim \inf_{k \to \infty} \gamma_k, \ 0 \right\}$$
Generalized Corrected Polyak Stepsize

- The general form: \( \lambda_k = f_k - f^* - \gamma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1
\]  

(9)

- (9) \( \Rightarrow \) (5) + (7) with \( \bar{\sigma}_k = (1 - \alpha_k)(\bar{\sigma}_{k-1} - \alpha_{k-1}\gamma_{k-1}) + \alpha_k\sigma_k \)

- General convergence:

\[
f^\infty \leq f^* + 2\Delta / \Gamma
\]

\[
\Delta = \sigma^* + \tilde{\gamma}( (1 - \beta^*)/\beta^* + \sup_k \alpha_k/2 )
\]

\[
\tilde{\gamma} = - \min \{ \gamma^* = \lim \inf_{k \to \infty} \gamma_k, 0 \}
\]

- “aiming higher than \( f^* \)” (\( \gamma_k > 0 \)) good,
  “aiming lower than \( f^* \)” (\( \gamma_k < 0 \)) bad
Generalized Corrected Polyak Stepsize

- The general form: \( \lambda_k = f_k - f^* - \gamma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \tag{9}
\]

- \( (9) \Rightarrow (5) + (7) \) with \( \bar{\sigma}_k = (1 - \alpha_k)(\bar{\sigma}_{k-1} - \alpha_{k-1} \gamma_{k-1}) + \alpha_k \sigma_k \)

- General convergence:

\[
f^\infty \leq f^* + 2\Delta / \Gamma
\]

\[
\Delta = \sigma^* + \tilde{\gamma} \left( \frac{1 - \beta^*}{\beta^*} + \sup_k \alpha_k / 2 \right)
\]

\[
\tilde{\gamma} = - \min \left\{ \gamma^* = \lim \inf_{k \to \infty} \gamma_k, 0 \right\}
\]

- “aiming higher than \( f^* \)” \( (\gamma_k > 0) \) good,
  “aiming lower than \( f^* \)” \( (\gamma_k < 0) \) bad

- On the other hand: aiming too high \( \Rightarrow \lambda_k < 0 \Rightarrow \) loop
Generalized Corrected Polyak Stepsize

- The general form: \( \lambda_k = f_k - f^* - \gamma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \quad (9)
\]

- (9) \( \Rightarrow \) (5) + (7) with \( \bar{\sigma}_k = (1 - \alpha_k)(\bar{\sigma}_{k-1} - \alpha_{k-1}\gamma_{k-1}) + \alpha_k\sigma_k \)

- General convergence:

\[
f^\infty \leq f^* + 2\Delta/\Gamma
\]

\[
\Delta = \sigma^* + \bar{\gamma}(1 - \beta^*)/\beta^* + \sup_k \alpha_k/2
\]

\[
\bar{\gamma} = - \min \{ \gamma^* = \liminf_{k \to \infty} \gamma_k , 0 \}
\]

- “aiming higher than \( f^* \)” (\( \gamma_k > 0 \)) good,
- “aiming lower than \( f^* \)” (\( \gamma_k < 0 \)) bad

- On the other hand: aiming too high \( \Rightarrow \lambda_k < 0 \Rightarrow \) loop

- The highest safe value: \( \sigma_k \) (surprised?)
Generalized Corrected Polyak Stepsize

- The general form: \( \lambda_k = f_k - f^* - \gamma_k \)

\[
0 \leq \nu_k = \beta_k \frac{\lambda_k}{\|d_k\|^2} \quad , \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1 \tag{9}
\]

- (9) \( \Rightarrow \) (5) + (7) with \( \bar{\sigma}_k = (1 - \alpha_k)(\bar{\sigma}_{k-1} - \alpha_{k-1}\gamma_{k-1}) + \alpha_k\sigma_k \)

- General convergence:

\[
f^\infty \leq f^* + 2\Delta/\Gamma
\]

\[
\Delta = \sigma^* + \bar{\gamma}( (1 - \beta^*)/\beta^* + \sup_k \alpha_k/2 )
\]

\[
\bar{\gamma} = - \min \{ \gamma^* = \lim inf_{k \to \infty} \gamma_k , 0 \}
\]

- “aiming higher than \( f^* \)” (\( \gamma_k > 0 \)) good,
- “aiming lower than \( f^* \)” (\( \gamma_k < 0 \)) bad

- On the other hand: aiming too high \( \Rightarrow \lambda_k < 0 \Rightarrow \) loop

- The highest safe value: \( \sigma_k \) (surprised?)

- What if I do not know \( \sigma_k \) exactly?
Reminder: $\gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k\sigma_k$

$\gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k\sigma_k$
Reminder: \( \gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k\sigma_k \)
\( \gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k\sigma_k \)

What if \( \gamma_k > 0 \) and “not too far” from \( \sigma_k \)?
Reminder: $\gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k$
$\gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k \sigma_k$

What if $\gamma_k > 0$ and “not too far” from $\sigma_k$?

Abstract condition ($\Rightarrow \tilde{\gamma} = 0$):

$$\liminf_{k \to \infty} \gamma_k = \gamma^* \geq \xi \sigma^* \quad \xi \in [0, 1] \quad (10)$$
Generalized (approximately) Corrected Polyak Stepsize

- Reminder: \( \gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k \)
  
  \( \gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k \sigma_k \)

- What if \( \gamma_k > 0 \) and “not too far” from \( \sigma_k \)?

- Abstract condition \( \Rightarrow \bar{\gamma} = 0 \):
  \[
  \liminf_{k \to \infty} \gamma_k = \gamma^* \geq \xi \sigma^* \quad \xi \in [0, 1] \quad (10)
  \]

- \( (10) \Rightarrow \bar{\sigma}_k \approx \sigma_k (1 - (1 - \alpha_k)\xi) \) (technical form really ugly)
Reminder: \( \gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k \)
\( \gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k \sigma_k \)

What if \( \gamma_k > 0 \) and “not too far” from \( \sigma_k \)?

Abstract condition (\( \Rightarrow \bar{\gamma} = 0 \)):
\[
\lim_{k \to \infty} \inf \gamma_k = \gamma^* \geq \xi \sigma^* \quad \xi \in [0, 1] \quad (10)
\]

\((10) \Rightarrow \bar{\sigma}_k \approx \sigma_k \left( 1 - (1 - \alpha_k)\xi \right) \quad \text{(technical form really ugly)}\)

Convergence: \((10) \Rightarrow f^\infty \leq f^* + \sigma^* \left( \xi + 2(1 - \xi)/\Gamma \right)\)
Generalized (approximately) Corrected Polyak Stepsize

- Reminder: $\gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k$
  $\gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k \sigma_k$

- What if $\gamma_k > 0$ and “not too far” from $\sigma_k$?

- Abstract condition ($\Rightarrow \bar{\gamma} = 0$):
  \[ \liminf_{k \to \infty} \gamma_k = \gamma^* \geq \xi \sigma^* \quad \xi \in [0, 1] \quad (10) \]

- $\Rightarrow \bar{\sigma}_k \approx \sigma_k \left( 1 - (1 - \alpha_k)\xi \right)$ (technical form really ugly)

- Convergence: $\Rightarrow f^\infty \leq f^* + \sigma^* \left( \xi + 2(1 - \xi)/\Gamma \right)$

- $\xi = 1 \Rightarrow$ “optimal” error
Reminder: \( \gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k) \bar{\sigma}_{k-1} + \alpha_k \sigma_k \)
\( \gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k \sigma_k \)

What if \( \gamma_k > 0 \) and “not too far” from \( \sigma_k \)?

Abstract condition (\( \Rightarrow \bar{\gamma} = 0 \)):

\[
\liminf_{k \to \infty} \gamma_k = \gamma^* \geq \xi \sigma^* \quad \xi \in [0, 1] \tag{10}
\]

\( (10) \Rightarrow \bar{\sigma}_k \approx \sigma_k (1 - (1 - \alpha_k) \xi) \) (technical form really ugly)

Convergence: \( (10) \Rightarrow f^\infty \leq f^* + \sigma^* (\xi + 2(1 - \xi)/\Gamma) \)

\( \xi = 1 \Rightarrow “optimal” \ \text{error} \)

Again, asymptotic results require \( \lambda_k \geq 0 \) for infinitely many \( k \), if not a solution with prescribed accuracy finitely attained.
Reminder: \( \gamma_k = 0 \Rightarrow \bar{\sigma}_k = (1 - \alpha_k)\bar{\sigma}_{k-1} + \alpha_k \sigma_k \)
\( \gamma_k = \sigma_k \Rightarrow \bar{\sigma}_k = \alpha_k \sigma_k \)

What if \( \gamma_k > 0 \) and “not too far” from \( \sigma_k \)?

Abstract condition \( \Rightarrow \bar{\gamma} = 0 \):

\[
\liminf_{k \to \infty} \gamma_k = \gamma^* \geq \xi \sigma^* \quad \xi \in [0, 1] \tag{10}
\]

(10) \( \Rightarrow \bar{\sigma}_k \approx \sigma_k (1 - (1 - \alpha_k)\xi) \) (technical form really ugly)

Convergence: (10) \( \Rightarrow f^\infty \leq f^* + \sigma^* (\xi + 2(1 - \xi)/\Gamma) \)

\( \xi = 1 \Rightarrow \) “optimal” error

Again, asymptotic results require \( \lambda_k \geq 0 \) for infinitely many \( k \), if not a solution with prescribed accuracy finitely attained

Is (10) reasonable?
1 Introduction, Motivation
2 Subgradient methods: introduction
3 Polyak-type stepsize: the abstract case
4 Polyak-type stepsize: the implementable case
5 Deflection-restricted rules
6 Bundle methods
7 Conclusions
Target-level Approaches

- In general, $f^*$ unknown (and it may be $-\infty$)
Target-level Approaches

- In general, $f^*$ unknown (and it may be $-\infty$)
- Solution: replace it with a target $f^k_{lev}$, revise it appropriately

$$0 \leq \nu_k = \beta_k \frac{f_k - f^k_{lev}}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1$$
Target-level Approaches

- In general, $f^*$ unknown (and it may be $-\infty$)

- Solution: replace it with a target $f^k_{lev}$, revise it appropriately

$$0 \leq \nu_k = \beta_k \frac{f_k - f^k_{lev}}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1$$

- Usually, $f^k_{lev} = f^k_{ref}$ (reference) $-\delta_k$ (threshold)

- Typical choice: $f^k_{ref} = f^k_{rec} = \min_{h\leq k} f(x_h)$ (record value)
Target-level Approaches

- In general, \( f^* \) unknown (and it may be \(-\infty\))

- Solution: replace it with a target \( f_{lev}^k \), revise it appropriately

\[
0 \leq \nu_k = \beta_k \frac{f_k - f_{lev}^k}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1
\]

- Usually, \( f_{lev}^k = f_{ref}^k \) (reference) \(-\delta_k \) (threshold)

- Typical choice: \( f_{ref}^k = f_{rec}^k = \min_{h \leq k} f(x_h) \) (record value)

- Looks uncorrected but it is not necessarily so:

\[
\lambda_k = f_k - f_{lev}^k = f_k - f^* - (f_{ref}^k - f^* - \delta_k)
\]

\[
\gamma_k = f_{ref}^k - f^* - \delta_k \text{ unknown}
\]
Target-level Approaches

- In general, \( f^* \) unknown (and it may be \(-\infty\))
- Solution: replace it with a target \( f^k_{\text{lev}} \), revise it appropriately

\[
0 \leq \nu_k = \beta_k \frac{f_k - f^k_{\text{lev}}}{\|d_k\|^2}, \quad 0 < \beta^* \leq \beta_k \leq \alpha_k \leq 1
\]

- Usually, \( f^k_{\text{lev}} = f^k_{\text{ref}} \) (reference) \(-\delta_k \) (threshold)
- Typical choice: \( f^k_{\text{ref}} = f^k_{\text{rec}} = \min_{h \leq k} f(x_h) \) (record value)
- Looks uncorrected but it is not necessarily so:

\[
\lambda_k = f_k - f^k_{\text{lev}} = f_k - f^* - (f^k_{\text{ref}} - f^* - \delta_k)
\]

\[
\gamma_k = f^k_{\text{ref}} - f^* - \delta_k \text{ unknown}
\]

- Small technical hurdle: all previous proofs require \( f^* > -\infty \)
- Solution: \( f^\infty_{\text{rec}} = -\infty \Rightarrow f^* = -\infty \), otherwise
  feasible target \( \bar{f} > -\infty, \bar{f} \geq f^*, \bar{f} \leq f^\infty_{\text{rec}} \) (\( \Rightarrow f_k - \bar{f} \geq 0 \)
Abstract property:

either \( f_{\text{ref}}^{\infty} = -\infty \), or \( \liminf_{k \to \infty} \delta_k = \delta^* > 0 \)
Non-vanishing Threshold

- Abstract property:

  either \( f^\infty_{\text{ref}} = -\infty \), or \( \liminf_{k \to \infty} \delta_k = \delta^* > 0 \)

- Implementation: \( \mu \in [0,1) \)

\[
\delta_{k+1} \in \begin{cases} 
  [ \delta^*, \infty ) & \text{if } f_{k+1} \leq f^k_{\text{lev}} \\
  [ \delta^*, \max\{ \delta^*, \mu \delta_k \} ] & \text{if } f_{k+1} > f^k_{\text{lev}}
\end{cases}
\]
Non-vanishing Threshold

bullet Abstract property:

either \( f_{\text{ref}}^\infty = -\infty \), or \( \lim \inf_{k \to \infty} \delta_k = \delta^* > 0 \)

bullet Implementation: \( \mu \in [0, 1) \)

\[
\delta_{k+1} \in \begin{cases} 
[ \delta^*, \infty ) & \text{if } f_{k+1} \leq f_{\text{lev}}^k \\
[ \delta^*, \max\{ \delta^*, \mu \delta_k \} ) & \text{if } f_{k+1} > f_{\text{lev}}^k 
\end{cases}
\]

bullet Convergence: either \( f_{\text{ref}}^\infty = -\infty = f^* \), or \( f_{\text{ref}}^\infty \leq f^* + \xi \sigma^* + \delta^* \) where \( 0 \leq \xi = \max \left\{ 1 - \delta^* \Gamma / 2 \sigma^* , 0 \right\} < 1 \)
Non-vanishing Threshold

- **Abstract property:**
  
  either \( f_{\text{ref}}^\infty = -\infty \),  
  or  
  \( \liminf_{k \to \infty} \delta_k = \delta^* > 0 \)

- **Implementation:**  \( \mu \in [0, 1) \)
  
  \[
  \delta_{k+1} \in \begin{cases} 
  \left[ \delta^* , \infty \right) & \text{if } f_{k+1} \leq f_{\text{lev}}^k \\
  \left[ \delta^* , \max\{ \delta^* , \mu \delta_k \} \right] & \text{if } f_{k+1} > f_{\text{lev}}^k 
  \end{cases}
  \]

- **Convergence:**  either \( f_{\text{ref}}^\infty = -\infty = f^* \),  
  or  
  \( f_{\text{ref}}^\infty \leq f^* + \xi \sigma^* + \delta^* \)  
  where  
  \( 0 \leq \xi = \max \{ 1 - \delta^* \Gamma / 2 \sigma^* , 0 \} < 1 \)

- **Proof:** (almost) straightforward, \( \gamma^* \geq \xi \sigma^* \)
Non-vanishing Threshold

- Abstract property:
  
  \[ f_{\text{ref}}^{\infty} = -\infty, \quad \text{or} \quad \liminf_{k \to \infty} \delta_k = \delta^* > 0 \]

- Implementation: \( \mu \in [0, 1) \)

  \[ \delta_{k+1} \in \begin{cases} [\delta^*, \infty) & \text{if } f_{k+1} \leq f_{\text{lev}}^k \\ [\delta^*, \max\{\delta^*, \mu\delta_k\}] & \text{if } f_{k+1} > f_{\text{lev}}^k \end{cases} \]

- Convergence: either \( f_{\text{ref}}^{\infty} = -\infty = f^* \), or \( f_{\text{ref}}^{\infty} \leq f^* + \xi\sigma^* + \delta^* \) where

  \[ 0 \leq \xi = \max \{ 1 - \delta^*\Gamma/2\sigma^*, 0 \} < 1 \]

- Proof: (almost) straightforward, \( \gamma^* \geq \xi\sigma^* \)

- Compares favorably with \( f_{\text{ref}}^{\infty} \leq f^* + \sigma^* + \delta^* \) (without deflection)
Non-vanishing Threshold

- Abstract property:
  
  either \( f^\infty_{\text{ref}} = -\infty \), or \( \lim \inf_{k \to \infty} \delta_k = \delta^* > 0 \)

- Implementation: \( \mu \in [0, 1) \)

  \[
  \delta_{k+1} \in \begin{cases} 
  [\delta^*, \infty) & \text{if } f_{k+1} \leq f^k_{\text{lev}} \\
  [\delta^*, \max\{\delta^*, \mu \delta_k\}] & \text{if } f_{k+1} > f^k_{\text{lev}} 
  \end{cases}
  \]

- Convergence: either \( f^\infty_{\text{ref}} = -\infty = f^* \), or \( f^\infty_{\text{ref}} \leq f^* + \xi \sigma^* + \delta^* \) where \( 0 \leq \xi = \max\{1 - \delta^* \Gamma / 2 \sigma^*, 0\} < 1 \)

- Proof: (almost) straightforward, \( \gamma^* \geq \xi \sigma^* \)

- Compares favorably with \( f^\infty_{\text{ref}} \leq f^* + \sigma^* + \delta^* \) (without deflection)

- Note: it may seem that “small \( \xi \) is good”, but \( \xi \sigma^* + \delta^* \geq \sigma^* \)
Abstract property:

\[ \text{either } f_{\text{ref}}^\infty = f^* = -\infty , \text{ or } \liminf_{k \to \infty} \delta_k = 0 \text{ and } \sum_{k=1}^{\infty} \frac{\lambda_k}{\|d_k\|^2} = \infty \]
Vanishing Threshold

- Abstract property:
  
  \[ f_{\text{ref}}^\infty = f^* = -\infty, \quad \text{or} \quad \liminf_{k \to \infty} \delta_k = 0 \quad \text{and} \quad \sum_{k=1}^{\infty} \lambda_k / \|d_k\|^2 = \infty \]

- Implementation: \( R > 0 \) and \( \mu \in [0, 1) \)
  
  - \( f_{\text{ref}}^1 = f(x_1), \quad \delta_1 \in (0, \infty), \quad r_1 = 0; \)
  
  - if \( f_k \leq f_{\text{ref}}^k - \delta_k / 2 \) (sufficient descent condition) then \( f_{\text{ref}}^k = f_{\text{rec}}^k, \quad r_k = 0; \)
  
  - else, if \( r_k > R \) (target infeasibility condition) then \( \delta_k = \mu \delta_{k-1}, \quad r_k = 0; \)
  
  - otherwise, \( f_{\text{ref}}^k = f_{\text{ref}}^{k-1}, \quad \delta_k = \delta_{k-1}, \quad r_k = r_{k-1} + \|\hat{x}_{k+1} - x_k\| \)

---

16 Lim, Sherali “Convergence . . . for Some Variable Target Value and Subgradient Deflection Methods”, COAP, 2006
Vanishing Threshold

Abstract property:

either \( f_{\text{ref}}^\infty = f^* = -\infty \), or \( \lim \inf_{k \to \infty} \delta_k = 0 \) and \( \sum_{k=1}^{\infty} \lambda_k / \|d_k\|^2 = \infty \)

Implementation: \( R > 0 \) and \( \mu \in [0, 1) \)

- \( f_{\text{ref}}^1 = f(x_1), \) \( \delta_1 \in (0, \infty), \) \( r_1 = 0; \)
- if \( f_k \leq f_{\text{ref}}^k - \delta_k / 2 \) (sufficient descent condition) then \( f_{\text{ref}}^k = f_{\text{rec}}^k, \) \( r_k = 0; \)
- else, if \( r_k > R \) (target infeasibility condition) then \( \delta_k = \mu \delta_{k-1}, \) \( r_k = 0; \)
- otherwise, \( f_{\text{ref}}^k = f_{\text{ref}}^{k-1}, \) \( \delta_k = \delta_{k-1}, \) \( r_k = r_{k-1} + \|\hat{x}_{k+1} - x_k\| \)

Convergence: either \( f_{\text{ref}}^\infty = -\infty = f^*, \) or \( f_{\text{ref}}^\infty \leq f^* + \sigma^* \)

Proof: again (almost) straightforward, \( \gamma^* \geq \sigma^* (\xi = 1), \) minor quirks

---

Lim, Sherali “Convergence . . . for Some Variable Target Value and Subgradient Deflection Methods”, COAP, 2006
Vanishing Threshold

- Abstract property:
  
  either \( f_{\infty}^{\text{ref}} = f^* = -\infty \), or \( \liminf_{k \to \infty} \delta_k = 0 \) and \( \sum_{k=1}^{\infty} \lambda_k / \|d_k\|^2 = \infty \).

- Implementation: \( R > 0 \) and \( \mu \in [0, 1) \)
  
  - \( f_{\text{ref}}^1 = f(x_1), \delta_1 \in (0, \infty), r_1 = 0; \)
  
  - if \( f_k \leq f_{\text{ref}}^k - \delta_k/2 \) (sufficient descent condition) then \( f_{\text{ref}}^k = f_{\text{rec}}^k, r_k = 0; \)
  
  - else, if \( r_k > R \) (target infeasibility condition) then \( \delta_k = \mu \delta_{k-1}, r_k = 0; \)
  
  - otherwise, \( f_{\text{ref}}^k = f_{\text{ref}}^{k-1}, \delta_k = \delta_{k-1}, r_k = r_{k-1} + \|\hat{x}_{k+1} - x_k\| \)

- Convergence: either \( f_{\infty}^{\text{ref}} = -\infty = f^* \), or \( f_{\infty}^{\text{ref}} \leq f^* + \sigma^* \)

- Proof: again (almost) straightforward, \( \gamma^* \geq \sigma^* (\xi = 1) \), minor quirks

Optimal error, extends known results\(^\text{16}\) to projection and errors

\(^\text{16}\) Lim, Sherali “Convergence . . . for Some Variable Target Value and Subgradient Deflection Methods”, COAP, 2006
Abstract property:

either \( f^\infty_{\text{ref}} = f^* = -\infty \), or \( \liminf_{k \to \infty} \delta_k = 0 \) and \( \sum_{k=1}^{\infty} \lambda_k / \|d_k\|^2 = \infty \)

Implementation: \( R > 0 \) and \( \mu \in [0, 1) \)

- \( f^1_{\text{ref}} = f(x_1), \delta_1 \in (0, \infty), r_1 = 0; \)
- if \( f_k \leq f^k_{\text{ref}} - \delta_k / 2 \) (sufficient descent condition) then \( f^k_{\text{ref}} = f^k_{\text{rec}}, r_k = 0; \)
- else, if \( r_k > R \) (target infeasibility condition) then \( \delta_k = \mu \delta_{k-1}, r_k = 0; \)
- otherwise, \( f^k_{\text{ref}} = f^{k-1}_{\text{ref}}, \delta_k = \delta_{k-1}, r_k = r_{k-1} + \|\hat{x}_{k+1} - x_k\| \)

Convergence: either \( f^\infty_{\text{ref}} = -\infty = f^* \), or \( f^\infty_{\text{ref}} \leq f^* + \sigma^* \)

Proof: again (almost) straightforward, \( \gamma^* \geq \sigma^* \) (\( \xi = 1 \)), minor quirks

Optimal error, extends known results\(^\text{16}\) to projection and errors

Weaker results than (8) \( (f^\infty \to f^\infty_{\text{ref}}, \) no convergence of \( \{x_k\}) \)

\(^\text{16}\) Lim, Sherali “Convergence . . . for Some Variable Target Value and Subgradient Deflection Methods”, COAP, 2006
1 Introduction, Motivation

2 Subgradient methods: introduction

3 Polyak-type stepsize: the abstract case

4 Polyak-type stepsize: the implementable case

5 Deflection-restricted rules

6 Bundle methods

7 Conclusions
Other main class of stepsize rules: diminishing/square summable

\[ \sum_{k=1}^{\infty} \nu_k = \infty \quad , \quad \sum_{k=1}^{\infty} \nu_k^2 < \infty \]  

(11)
Other main class of stepsize rules: **diminishing/square summable**

\[
\sum_{k=1}^{\infty} \nu_k = \infty, \quad \sum_{k=1}^{\infty} \nu_k^2 < \infty
\]  

- **Pros:** do not need \( f^* \), not even any estimate
Other main class of stepsize rules: diminishing/square summable

\[ \sum_{k=1}^{\infty} \nu_k = \infty \quad , \quad \sum_{k=1}^{\infty} \nu_k^2 < \infty \quad (11) \]

- Pros: do not need \( f^* \), not even any estimate
- Cons: no control over \( \varepsilon_k \) (cf. (5), (6))

All our results hinge over these estimates
Diminishing/Square Summable Stepsize

- Other main class of stepsize rules: diminishing/square summable

\[ \sum_{k=1}^{\infty} \nu_k = \infty \quad , \quad \sum_{k=1}^{\infty} \nu_k^2 < \infty \]  \hspace{1cm} (11)

- Pros: do not need \( f^* \), not even any estimate
- Cons: no control over \( \varepsilon_k \) (cf. (5), (6))

- All our results hinge over these estimates

- Solution: restrict the deflection instead of the stepsize

\[ 0 \leq \zeta_k = \frac{\nu_{k-1} \|d_{k-1}\|^2}{(f_k - f^*) + \nu_{k-1} \|d_{k-1}\|^2} \leq \alpha_k \leq 1 \]
Other main class of stepsize rules: diminishing/square summable

\[
\sum_{k=1}^{\infty} \nu_k = \infty, \quad \sum_{k=1}^{\infty} \nu_k^2 < \infty
\]  

Pros: do not need \( f^* \), not even any estimate
Cons: no control over \( \varepsilon_k \) (cf. (5), (6))

All our results hinge over these estimates

Solution: restrict the deflection instead of the stepsize

\[
0 \leq \zeta_k = \frac{\nu_{k-1}\|d_{k-1}\|^2}{(f_k - f^*) + \nu_{k-1}\|d_{k-1}\|^2} \leq \alpha_k \leq 1
\]

Gives analogous to (5), (6)

\[
\varepsilon_k \leq f_k - f^* + \bar{\sigma}_k
\]

where \( \bar{\sigma}_k = \alpha_k \sigma_k + (1 - \alpha_k) \bar{\sigma}_{k-1} \)
Moving “towards $x^*$” is a short enough step
Deflection Rule (geometrically)

- Moving “towards $x^*$” is a short enough step and then any deflection
Moving “towards $x^*$” is a **short enough step** and then **any deflection**

... or **any step**
Moving “towards \( x^* \)” is a **short enough step** and then any deflection

... or any **step** and a **proper deflection**
We learnt our lesson: corrected deflection rule

\[ 0 \leq \zeta_k = \frac{\nu_{k-1}\|d_{k-1}\|^2}{(f_k - f^* - \gamma_k) + \nu_{k-1}\|d_{k-1}\|^2} \leq \alpha_k \leq 1 \]
Corrected Deflection Rule

- We learnt our lesson: corrected deflection rule

\[
0 \leq \zeta_k = \frac{\nu_{k-1} \|d_{k-1}\|^2}{(f_k - f^* - \gamma_k) + \nu_{k-1} \|d_{k-1}\|^2} \leq \alpha_k \leq 1
\]

- Avoid \( \zeta_k \) is undefined (\( \lambda_k = f_k - f^* - \gamma_k \)):

\[
\nu_{k-1} \|d_{k-1}\|^2 \leq \alpha_k (\lambda_k + \nu_{k-1} \|d_{k-1}\|^2) \quad (13)
\]

- Avoid negative \( \lambda_k \): makes (13) impossible

\[
\lambda_k \geq 0 \implies \alpha_k \geq \alpha^* > 0
\]

\[
\lambda_k < 0 \implies \alpha_k = 0 \implies \nu_k = 0 \quad (14)
\]
We learnt our lesson: corrected deflection rule

\[ 0 \leq \zeta_k = \frac{\nu_{k-1}\|d_{k-1}\|^2}{(f_k - f^* - \gamma_k) + \nu_{k-1}\|d_{k-1}\|^2} \leq \alpha_k \leq 1 \]

Avoid \( \zeta_k \) is undefined (\( \lambda_k = f_k - f^* - \gamma_k \)):

\[ \nu_{k-1}\|d_{k-1}\|^2 \leq \alpha_k(\lambda_k + \nu_{k-1}\|d_{k-1}\|^2) \quad (13) \]

Avoid negative \( \lambda_k \): makes (13) impossible

\[ \lambda_k \geq 0 \implies \alpha_k \geq \alpha^* > 0 \]
\[ \lambda_k < 0 \implies \alpha_k = 0 \implies \nu_k = 0 \quad (14) \]

Now \( \varepsilon_k \) is controlled: (12) holds with

\[ \bar{\sigma}_k = \alpha_k(\sigma_k - \gamma_k) + (1 - \alpha_k)\bar{\sigma}_{k-1} \]

Yields the crucial technical relationship, similar to (7)

\[ \bar{d}_k(\bar{x} - x_k) \leq f(\bar{x}) - f^* + \bar{\sigma}_k \]
Convergence Results

- Relationships between $\sigma^*$ and $\bar{\sigma}^*$:
  - in general, $\bar{\sigma}^* \leq \sigma^* + \bar{\gamma}$
  - $\gamma_k \geq \xi \sigma_k \ \forall k \text{ large enough} \Rightarrow \bar{\sigma}^* \leq (1 - \xi)\sigma^*$
Convergence Results

- Relationships between $\sigma^*$ and $\bar{\sigma}^*$:
  - in general, $\bar{\sigma}^* \leq \sigma^* + \bar{\gamma}$
  - $\gamma_k \geq \xi \sigma_k \ \forall k \text{ large enough} \Rightarrow \bar{\sigma}^* \leq (1 - \xi)\sigma^*$

- Convergence: under $\sup_k \|d_k\| < \infty$
  1) in general, $f^\infty \leq f^* + \gamma^{\sup} + (\sigma^* + \bar{\gamma})/\alpha^*$
  2) $\gamma_k \geq \xi \sigma_k \Rightarrow f^\infty \leq f^* + \sigma^*(1 + (1 - \xi)(1 - \alpha^*)/\alpha^*)$
  3) $\gamma_k = \sigma_k \Rightarrow f^\infty \leq f^* + \sigma^*$
     furthermore, $X^* \neq \emptyset \Rightarrow \{x_k\} \to x^\infty \in X \text{ s.t. } f(x^\infty) = f^\infty$
Convergence Results

- Relationships between $\sigma^*$ and $\bar{\sigma}^*$:
  - in general, $\bar{\sigma}^* \leq \sigma^* + \bar{\gamma}$
  - $\gamma_k \geq \xi \sigma_k \ \forall k$ large enough $\Rightarrow \bar{\sigma}^* \leq (1 - \xi)\sigma^*$

- Convergence: under $\sup_k \|d_k\| < \infty$
  - i) in general, $f^\infty \leq f^* + \gamma^{\sup} + (\sigma^* + \bar{\gamma}) / \alpha^*$
  - ii) $\gamma_k \geq \xi \sigma_k \Rightarrow f^\infty \leq f^* + \sigma^*(1 + (1 - \xi)(1 - \alpha^*) / \alpha^*)$
  - iii) $\gamma_k = \sigma_k \Rightarrow f^\infty \leq f^* + \sigma^*$
    furthermore, $X^* \neq \emptyset \Rightarrow \{x_k\} \rightarrow x^\infty \in X \text{ s.t. } f(x^\infty) = f^\infty$

- Analogous to previous results, **optimal** error

- Boundedness assumption easily attained (bounding strategies\(^7\))
Convergence Results

- Relationships between $\sigma^*$ and $\bar{\sigma}^*$:
  - in general, $\bar{\sigma}^* \leq \sigma^* + \bar{\gamma}$
  - $\gamma_k \geq \xi \sigma_k \ \forall k$ large enough $\Rightarrow \bar{\sigma}^* \leq (1 - \xi)\sigma^*$

- Convergence: under $\sup_k \|d_k\| < \infty$
  1) in general, $f^\infty \leq f^* + \gamma^{\sup} + (\sigma^* + \bar{\gamma})/\alpha^*$
  2) $\gamma_k \geq \xi \sigma_k \Rightarrow f^\infty \leq f^* + \sigma^*(1 + (1 - \xi)(1 - \alpha^*)/\alpha^*)$
  3) $\gamma_k = \sigma_k \Rightarrow f^\infty \leq f^* + \sigma^*$
    furthermore, $X^* \neq \emptyset \Rightarrow \{x_k\} \to x^\infty \in X \ s.t. \ f(x^\infty) = f^\infty$

- Analogous to previous results, optimal error

- Boundedness assumption easily attained (bounding strategies$^7$)

- Technical notes: $\nu_k = 0$ from (14) at odds with the very (11)
  $\Rightarrow$ finite case to be considered carefully
Relationships between $\sigma^*$ and $\bar{\sigma}^*$:

- in general, $\bar{\sigma}^* \leq \sigma^* + \bar{\gamma}$
- $\gamma_k \geq \xi \sigma_k \ \forall k$ large enough $\Rightarrow \bar{\sigma}^* \leq (1 - \xi)\sigma^*$

Convergence: under $\sup_k \|d_k\| < \infty$

i) in general, $f^* \leq f^* + \gamma^{\sup} + (\sigma^* + \bar{\gamma})/\alpha^*$

ii) $\gamma_k \geq \xi \sigma_k \Rightarrow f^* \leq f^* + \sigma^* \left(1 + (1 - \xi)(1 - \alpha^*)/\alpha^* \right)$

iii) $\gamma_k = \sigma_k \Rightarrow f^* \leq f^* + \sigma^*$
   furthermore, $X^* \neq \emptyset \Rightarrow \{x_k\} \rightarrow x^\infty \in X \text{ s.t. } f(x^\infty) = f^*$

- Analogous to previous results, optimal error

- Boundedness assumption easily attained (bounding strategies\(^7\))

- Technical notes: $\nu_k = 0$ from (14) at odds with the very (11) $\Rightarrow$ finite case to be considered carefully

- As usual, $f^*$ not available (and may be $-\infty$) $\Rightarrow$ same trick
Target Value Deflection

- Target value deflection rule

\[
0 \leq \zeta_k = \frac{\nu_{k-1}\|d_{k-1}\|^2}{(f_k - f^k_{lev}) + \nu_{k-1}\|d_{k-1}\|^2} \leq \alpha_k \leq 1
\]

(as before, looks uncorrected but it is not: \(\gamma_k\) unknown)
Target Value Deflection

- Target value deflection rule

\[ 0 \leq \zeta_k = \frac{\nu_{k-1}\|d_{k-1}\|^2}{(f_k - f_{\text{lev}}^k) + \nu_{k-1}\|d_{k-1}\|^2} \leq \alpha_k \leq 1 \]

(as before, looks uncorrected but it is not: \(\gamma_k\) unknown)

- Abstract property:

  either \(f_{\text{ref}}^\infty = f^* = -\infty\), or \(\liminf_{k \to \infty} \delta_k = 0\).
Target Value Deflection

- **Target value deflection rule**

\[
0 \leq \zeta_k = \frac{\nu_{k-1} \|d_{k-1}\|^2}{(f_k - f_{lev}^k) + \nu_{k-1} \|d_{k-1}\|^2} \leq \alpha_k \leq 1
\]

(as before, looks uncorrected but it is not: \(\gamma_k\) unknown)

- **Abstract property:**

  either \(f^\infty_{\text{ref}} = f^* = -\infty\), or \(\lim_{k \to \infty} \delta_k = 0\).

- **Implementation:**

\[
\delta_{k+1} \in \begin{cases} 
\Delta_{r(k)+1}, \infty \quad & \text{if } f(x_{k+1}) \leq f_{lev}^k \\
\Delta_{k+1} \quad & \text{if } f(x_{k+1}) > f_{lev}^k
\end{cases}
\]

where \(r(k) = \#h \leq k\) s.t. \(f_{h+1} \leq f_{lev}^h\) and

\[
\Delta_k > 0 \ , \quad \lim_{k \to \infty} \inf \Delta_k = 0 \ , \quad \sum_{k=1}^{\infty} \Delta_k = \infty
\]
Similar technical hurdles (reference value, \ldots)

Convergence: either $f_{\text{ref}}^\infty = -\infty = f^*$, or $f_{\text{ref}}^\infty \leq f^* + \sigma^*$

Easy proof (all the dirty work done already)
Similar technical hurdles (reference value, ...)

Convergence: either \( f_{ref}^{\infty} = -\infty = f^* \), or \( f_{ref}^{\infty} \leq f^* + \sigma^* \)

Easy proof (all the dirty work done already)

Same as stepsize-restricted (but it was not obvious beforehand)
Similar technical hurdles (reference value, . . . )

Convergence: either $f_{ref}^\infty = -\infty = f^*$, or $f_{ref}^\infty \leq f^* + \sigma^*$

Easy proof (all the dirty work done already)

Same as stepsize-restricted (but it was not obvious beforehand)

Conclusions (for now)
Similar technical hurdles (reference value, ...)

Convergence: either \( f_{\text{ref}}^\infty = -\infty = f^* \), or \( f_{\text{ref}}^\infty \leq f^* + \sigma^* \)

Easy proof (all the dirty work done already)

Same as steps-size-restricted (but it was not obvious beforehand)

Conclusions (for now)

1. If \( \sigma^* \) is your error, then \( f^* + \sigma^* \) is your target
Similar technical hurdles (reference value, . . . )

Convergence: either $f_{\text{ref}}^\infty = -\infty = f^*$, or $f_{\text{ref}}^\infty \leq f^* + \sigma^*$

Easy proof (all the dirty work done already)

Same as stepsize-restricted (but it was not obvious beforehand)

Conclusions (for now)

1. If $\sigma^*$ is your error, then $f^* + \sigma^*$ is your target

2. Knowing $\sigma_k$, even approximately, is useful
Bundle Methods

(with Giovanni Giallombardo)
Any iterative algorithm produces a sequence $\{x_k\}$ of tentative points $\Rightarrow$ the $f$-values sequence $\{f_k\}$ and the bundle $\mathcal{B} = \{z_k \in \partial f(x_k)\}$
Any iterative algorithm produces a sequence \( \{x_k\} \) of tentative points \( \Rightarrow \) the \( f \)-values sequence \( \{f_k\} \) and the bundle \( \mathcal{B} = \{z_k \in \partial f(x_k)\} \)

Idea: use \( \mathcal{B} \) to construct a model \( f^k_\mathcal{B} \) of \( f \), e.g.

\[
\hat{f}^k_\mathcal{B}(x) = \sup_{\bar{z}} \left\{ \bar{z} x - f^*(\bar{z}) : \bar{z} \in \mathcal{B} \right\}
\]

(cutting plane model)

---

(exact) Bundle Methods: the Basic Ideas

- Any iterative algorithm produces a sequence $\{x_k\}$ of tentative points ⇒ the $f$-values sequence $\{f_k\}$ and the bundle $\mathcal{B} = \{z_k \in \partial f(x_k)\}$

- Idea: use $\mathcal{B}$ to construct a model $f^k_{\mathcal{B}}$ of $f$, e.g.

$$\hat{f}^k_{\mathcal{B}}(x) = \sup_{\bar{z}} \{ \bar{z}x - f^*(\bar{z}) : \bar{z} \in \mathcal{B} \}$$

(cutting plane model)

- Immediate consequence: cutting plane algorithm

$$x_{k+1} = \text{argmin} \{ \hat{f}^k_{\mathcal{B}}(x) : x \in X \}$$

---

Any iterative algorithm produces a sequence \( \{x_k\} \) of tentative points \( \Rightarrow \) the \( f \)-values sequence \( \{f_k\} \) and the bundle \( B = \{z_k \in \partial f(x_k)\} \)

Idea: use \( B \) to construct a model \( f^k_B \) of \( f \), e.g.

\[
\hat{f}^k_B(x) = \sup_{\bar{z}} \{ \bar{z}x - f^*(\bar{z}) : \bar{z} \in B \}
\]

(cutting plane model)

Immediate consequence: cutting plane algorithm

\[
x_{k+1} = \operatorname{argmin} \{ \hat{f}^k_B(x) : x \in X \}
\]

Simple to implement, one linear program at each iteration

---

Any iterative algorithm produces a sequence \(\{x_k\}\) of tentative points ⇒ the \(f\)-values sequence \(\{f_k\}\) and the bundle \(\mathcal{B} = \{z_k \in \partial f(x_k)\}\)

**Idea:** use \(\mathcal{B}\) to construct a model \(f^k_\mathcal{B}\) of \(f\), e.g.

\[
\hat{f}^k_\mathcal{B}(x) = \sup_{\bar{z}} \left\{ \bar{z}x - f^*(\bar{z}) : \bar{z} \in \mathcal{B} \right\}
\]

(cutting plane model)

**Immediate consequence:** cutting plane algorithm

\[
x_{k+1} = \arg\min \left\{ \hat{f}^k_\mathcal{B}(x) : x \in \mathcal{X} \right\}
\]

Simple to implement, one linear program at each iteration

Unfortunately, often rather slow in practice (with exceptions)\(^{17}\)

\(^{17}\) Jones, Lustig, Farwolden, Powell “Multicommodity Network Flows: The Impact of Formulation on Decomposition” Math. Prog., 1993
Any iterative algorithm produces a sequence \( \{x_k\} \) of tentative points ⇒ the \( f \)-values sequence \( \{f_k\} \) and the bundle \( B = \{z_k \in \partial f(x_k)\} \)

Idea: use \( B \) to construct a model \( f^k_B \) of \( f \), e.g.

\[
\hat{f}^k_B(x) = \sup_{\bar{z}} \left\{ \bar{z}x - f^*(\bar{z}) : \bar{z} \in B \right\}
\]

(cutting plane model)

Immediate consequence: cutting plane algorithm

\[
x_{k+1} = \arg\min \{ \hat{f}^k_B(x) : x \in X \}
\]

Simple to implement, one linear program at each iteration

Unfortunately, often rather slow in practice (with exceptions)\(^{17}\)

Problem: instability

---

\(^{17}\) Jones, Lustig, Farwolden, Powell “Multicommodity Network Flows: The Impact of Formulation on Decomposition” Math. Prog., 1993
Issue: $x_{k+1}$ can be far from $x_k$
Issue: $x_{k+1}$ can be far from $x_k$ ... even infinitely far
Issue: $x_{k+1}$ can be far from $x_k$ ... even infinitely far

Solution: stabilize the model
Issue: $x_{k+1}$ can be far from $x_k$ ... even infinitely far

Solution: stabilize the model ... with the right weight
Primal View of (Generalized) Bundle Methods

- **Stabilization**: Stabilized primal problem \( (\mathcal{X} = \mathbb{R}^n \text{ for simplicity}) \)

\[
(\Pi_{\bar{x}, t}) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\}
\]  

(15)
Primal View of (Generalized) Bundle Methods

- **Stabilization**: stabilized primal problem \( (\mathcal{X} = \mathbb{R}^n \text{ for simplicity}) \)

\[
(\Pi_{\bar{x}, t}) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\}
\]  

- **current point** \( \bar{x} \)
**Stabilization**: stabilized primal problem ($X = \mathbb{R}^n$ for simplicity)

$$(\Pi_{\bar{x},t}) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\} \quad (15)$$

- current point $\bar{x}$
- $\phi_t = (generalized) \text{ Moreau–Yosida regularization of } f$
**Stabilization**: stabilized primal problem \((\mathcal{X} = \mathbb{R}^n \text{ for simplicity})\)

\[
(\Pi_{\bar{x}, t}) \quad \phi_t(\bar{x}) = \inf_d \{ f(\bar{x} + d) + D_t(d) \} \quad (15)
\]

- current point \(\bar{x}\)
- \(\phi_t = (\text{generalized}) \text{ Moreau–Yosida regularization of } f\)
- \(D_t = \text{stabilizing term (≈ norm), } t = \text{proximity weight}\)
Primal View of (Generalized) Bundle Methods

- **Stabilization**: stabilized primal problem ($X = \mathbb{R}^n$ for simplicity)

  \[
  (\Pi_{\bar{x},t}) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\} \quad (15)
  \]

  - current point $\bar{x}$
  - $\phi_t$ = (generalized) Moreau–Yosida regularization of $f$
  - $D_t$ = stabilizing term ($\approx$ norm), $t =$ proximity weight

- With proper $D_t$, good properties (e.g. smooth)
Stabilization: stabilized primal problem ($X = \mathbb{R}^n$ for simplicity)

\[
(\Pi_{\bar{x}}, t) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\}
\]

- current point $\bar{x}$
- $\phi_t$ = (generalized) Moreau–Yosida regularization of $f$
- $D_t$ = stabilizing term ($\approx$ norm), $t$ = proximity weight

With proper $D_t$, good properties (e.g. smooth)

But computing $\phi_t$ with an oracle for $f$ is difficult $\Rightarrow$ approximation
**Stabilization**: stabilized primal problem ($\mathcal{X} = \mathbb{R}^n$ for simplicity)

$$
(\Pi_{\bar{x},t}) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\} \quad (15)
$$

- current point $\bar{x}$
- $\phi_t$ = (generalized) Moreau–Yosida regularization of $f$
- $D_t$ = stabilizing term ($\approx$ norm), $t$ = proximity weight

- With proper $D_t$, good properties (e.g. smooth)

- But computing $\phi_t$ with an oracle for $f$ is difficult $\Rightarrow$ approximation

**Stabilized primal master problem**

$$
(\Pi_{\mathcal{B},\bar{x},t}) \quad \phi_{\mathcal{B},t}(\bar{x}) = \inf_d \left\{ f_{\mathcal{B}}(\bar{x} + d) + D_t(d) \right\} \quad (16)
$$
Primal View of (Generalized) Bundle Methods

- **Stabilization**: stabilized primal problem \((\mathcal{X} = \mathbb{R}^n \text{ for simplicity})\)

\[(\Pi_{\bar{x}, t}) \quad \phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\} \]  \hspace{1cm} (15)

  - current point \(\bar{x}\)
  - \(\phi_t = \text{(generalized) Moreau–Yosida regularization of } f\)
  - \(D_t = \text{stabilizing term (≈ norm), } t = \text{proximity weight}\)

- With proper \(D_t\), good properties (e.g. smooth)

- But computing \(\phi_t\) with an oracle for \(f\) is difficult \(\Rightarrow\) approximation

- Stabilized primal master problem

\[(\Pi_{\mathcal{B}, \bar{x}, t}) \quad \phi_{\mathcal{B}, t}(\bar{x}) = \inf_d \left\{ f_{\mathcal{B}}(\bar{x} + d) + D_t(d) \right\} \]  \hspace{1cm} (16)

  - \(x_{k+1} = \bar{x} + d^*, \text{ compute } f_{k+1}, \mathcal{B} = \mathcal{B} \cup \{z_{k+1}\}\)
Stabilization: stabilized primal problem ($X = \mathbb{R}^n$ for simplicity)

$$\phi_t(\bar{x}) = \inf_d \left\{ f(\bar{x} + d) + D_t(d) \right\}$$  \hspace{1cm} (15)

- current point $\bar{x}$
- $\phi_t = (\text{generalized})$ Moreau–Yosida regularization of $f$
- $D_t = \text{stabilizing term} \ (\approx \text{norm}), \ t = \text{proximity weight}$

With proper $D_t$, good properties (e.g. smooth)

But computing $\phi_t$ with an oracle for $f$ is difficult $\Rightarrow$ approximation

Stabilized primal master problem

$$\phi_{B,t}(\bar{x}) = \inf_d \left\{ f_B(\bar{x} + d) + D_t(d) \right\}$$ \hspace{1cm} (16)

- $x_{k+1} = \bar{x} + d^*$, compute $f_{k+1}, B = B \cup \{z_{k+1}\}$
- if $f_{k+1} \ll f(\bar{x})$, then $\bar{x} = x_{k+1}$
Dual View of (Generalized) Bundle Methods

- **Dual of (Π)\(^{18}\):** \(\Delta\) \(f^*(0) = \inf_z \{ f^*(z) : z = 0 \} \)

\(^{18}\) F. “Generalized Bundle Methods”, SIOPT, 2002
Dual View of (Generalized) Bundle Methods

- **Dual** of $(\Pi)^{18}$: 
  \[
  (\Delta) \quad f^*(0) = \inf_z \{ f^*(z) : z = 0 \}
  \]

- May look funny, but then every $f$ is a Lagrangian function:
  \[
  (\Delta_{\bar{x}}) \quad f(\bar{x}) = -\inf_z \{ f^*(z) - z\bar{x} \}
  \]

---

18 F. “Generalized Bundle Methods”, SIOPT, 2002
Dual View of (Generalized) Bundle Methods

- **Dual of \((\Pi)^{18}\)**: \((\Delta)\)
  \[
  f^*(0) = \inf_z \{ f^*(z) : z = 0 \}
  \]

- May look funny, but then **every \(f\) is a Lagrangian function**:
  \[
  \left(\Delta_x\right) f(\bar{x}) = -\inf_z \{ f^*(z) - z\bar{x} \}
  \]

- **Further, (15)** has a non-weird (Fenchel’s) dual
  \[
  \left(\Delta_{\bar{x}, t}\right) \inf_z \{ f^*(z) - z\bar{x} + D^*_t(-z) \}
  \]
  \[
  = \text{(generalized) Augmented Lagrangian of } (\Delta) \Rightarrow \text{so has } (16)
  \]
  \[
  \left(\Delta_{\mathcal{B}, \bar{x}, t}\right) \inf_z \{ f_{\mathcal{B}}^*(z) - z\bar{x} + D_t^*(-z) \}
  \]

---

\(^{18}\) F. “Generalized Bundle Methods”, SIOPT, 2002
Dual View of (Generalized) Bundle Methods

- **Dual of** $(\Pi)^{18}$:  
  \[ (\Delta) \quad f^*(0) = \inf_z \{ f^*(z) : z = 0 \} \]

- May look funny, but then every $f$ is a Lagrangian function:
  \[ (\Delta \bar{x}) \quad f(\bar{x}) = -\inf_z \{ f^*(z) - z\bar{x} \} \]

- Further, (15) has a non-weird (Fenchel’s) dual
  \[ (\Delta \bar{x}, t) \quad \inf_z \{ f^*(z) - z\bar{x} + D^*_t(-z) \} \]
  \[ = \text{(generalized) Augmented Lagrangian of } (\Delta) \Rightarrow \text{so has (16)} \]
  \[ (\Delta B, \bar{x}, t) \quad \inf_z \{ f^*_B(z) - z\bar{x} + D^*_t(-z) \} \]

- Illustration: $f_B = \hat{f}_B$, $g(u) = Au - b$, $x \geq 0$
  \[ (\Delta B, \bar{x}, t) \equiv \sup_u \left\{ \begin{array}{l} c(u) + \bar{x}z - D^*_t(-z) \\ z = b + \omega - Au, \quad \omega \geq 0, \quad u \in \text{co } B \subseteq U \end{array} \right. \]
  \[ \Rightarrow \text{actually solving the weird convexification (3)} \]

---

18 F. “Generalized Bundle Methods”, SIOPT, 2002
The Decomposable Case

- \( f(x) = \sum_{h \in \mathcal{K}} f^h(x) \), computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)

---

The Decomposable Case

- \( f(x) = \sum_{h \in \mathcal{K}} f^h(x) \), computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)

- Can aggregate: \( \sum_{h \in \mathcal{K}} z^h = z \in \partial f(x) \)

---

The Decomposable Case

- \( f(x) = \sum_{h \in \mathcal{K}} f^h(x) \), computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)
- Can aggregate: \( \sum_{h \in \mathcal{K}} z^h = z \in \partial f(x) \)
- Better yet: use separate models \( f^h_{\mathcal{B}} \) for each component

---

The Decomposable Case

- \( f(x) = \sum_{h \in \mathcal{K}} f^h(x) \), computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)

- Can aggregate: \( \sum_{h \in \mathcal{K}} z^h = z \in \partial f(x) \)

- Better yet: use separate models \( f^h_B \) for each component

- **Disaggregated** master problems (\( \mathcal{X} = \mathbb{R}^n \) for simplicity)

\[
\begin{align*}
(\Pi_B, \bar{x}, t) & \quad \inf_d \left\{ \sum_{h \in \mathcal{K}} f^h_B(\bar{x} + d) + D_t(d) \right\} \\
(\Delta_B, \bar{x}, t) & \quad \inf_z \left\{ \sum_{h \in \mathcal{K}} (f^h_B)^*(z^h) - \left( \sum_{h \in \mathcal{K}} z^h \right)^\ast \bar{x} + D_t^* \left( - \sum_{h \in \mathcal{K}} z^h \right) \right\}
\end{align*}
\]

---

The Decomposable Case

- \( f(x) = \sum_{h \in K} f^h(x) \), computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)

- Can aggregate: \( \sum_{h \in K} z^h = z \in \partial f(x) \)

- Better yet: use separate models \( f^h_B \) for each component

- **Disaggregated** master problems (\( X = \mathbb{R}^n \) for simplicity)

\[
(\Pi_{B, \bar{x}, t}) \quad \inf_d \left\{ \sum_{h \in K} f^h_B(\bar{x} + d) + D_t(d) \right\}
\]

\[
(\Delta_{B, \bar{x}, t}) \quad \inf_z \left\{ \sum_{h \in K}(f^h_B)^*(z^h) - \left( \sum_{h \in K} z^h \right)\bar{x} + D_t^* \left( - \sum_{h \in K} z^h \right) \right\}
\]

- Often more efficient in practice\(^{17,19}\), for good reasons

\(^{19}\) Bacaud, Lemaréchal, Renaud, Sagastizábal “Bundle methods in stochastic optimal power management: a disaggregated approach using preconditioners” COAP, 2001
The Decomposable Case

\[ f(x) = \sum_{h \in \mathcal{K}} f^h(x), \] computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)

Can aggregate: \( \sum_{h \in \mathcal{K}} z^h = z \in \partial f(x) \)

Better yet: use separate models \( f^h_B \) for each component

Disaggregated master problems (\( X = \mathbb{R}^n \) for simplicity)

\[
\begin{align*}
(\Pi_{B,\bar{x},t}) & \quad \inf_d \left\{ \sum_{h \in \mathcal{K}} f^h_B(\bar{x} + d) + D_t(d) \right\} \\
(\Delta_{B,\bar{x},t}) & \quad \inf_z \left\{ \sum_{h \in \mathcal{K}} (f^h_B)^*(z^h) - \left( \sum_{h \in \mathcal{K}} z^h \right) \bar{x} + D^*_t \left( - \sum_{h \in \mathcal{K}} z^h \right) \right\}
\end{align*}
\]

Often more efficient in practice\(^{17,19}\), for good reasons

Master problem more costly to solve, but faster convergence

\(^{19}\) Bacaud, Lemaréchal, Renaud, Sagastizábal “Bundle methods in stochastic optimal power management: a disaggregated approach using preconditioners” COAP, 2001
The Decomposable Case

- \( f(x) = \sum_{h \in \mathcal{K}} f^h(x) \), computing each \( f^h \) produces \( z^h \in \partial f^h(x) \)

- Can aggregate: \( \sum_{h \in \mathcal{K}} z^h = z \in \partial f(x) \)

- Better yet: use separate models \( f^h_B \) for each component

- **Disaggregated** master problems (\( X = \mathbb{R}^n \) for simplicity)

\[
(\Pi_{\mathcal{B},\bar{x},t}) \quad \inf_d \left\{ \sum_{h \in \mathcal{K}} f^h_B(\bar{x} + d) + D_t(d) \right\}
\]

\[
(\Delta_{\mathcal{B},\bar{x},t}) \quad \inf_z \left\{ \sum_{h \in \mathcal{K}} (f^h_B)^*(z^h) - \left( \sum_{h \in \mathcal{K}} z^h \right) \bar{x} + D_t^*(\bar{\sum}_{h \in \mathcal{K}} z^h) \right\}
\]

- Often more efficient in practice\(^{17} 19\), for good reasons

- Master problem more costly to solve, but faster convergence

- **No incremental version** as yet

\(^{19}\) Bacaud, Lemaréchal, Renaud, Sagastizábal “Bundle methods in stochastic optimal power management: a disaggregated approach using preconditioners” COAP, 2001
Approximate Bundle Methods

- Proposal exist only using lower bound\textsuperscript{8, 9} or for finite min-max\textsuperscript{20}
- Unify and extend these.

Approximate Bundle Methods

- Proposal exist only using lower bound \(^8^9\) or for finite min-max\(^20\)
- Unify and extend these.

### Definition

**Incremental inexact oracle** for \(f\): inputs \(\bar{x} \in \mathbb{R}^n\), outputs:

- \(\underline{f} \leq f(\bar{x})\), \(z \in \mathbb{R}^n\) s.t. \(\underline{f} + z(x - \bar{x}) \leq f(x) \quad \forall x\) (lower linearization)
- \(\bar{f} \geq f(\bar{x})\) (upper bound, may be \(+\infty\))

Can be called **repeatedly** on the same \(\bar{x}\).

- Different rules governing the produced sequences \(\{f_j\}, \{\bar{f}_j\}\)

---

Approximate Bundle Methods

- Proposal exist only using lower bound \(^8^9\) or for finite min-max\(^{20}\)
- Unify and extend these.

**Definition**

**Incremental inexact oracle** for \(f\): inputs \(\bar{x} \in \mathbb{R}^n\), outputs:

- \(\bar{f} \leq f(\bar{x})\), \(z \in \mathbb{R}^n\) s.t. \(\bar{f} + z(x - \bar{x}) \leq f(x) \forall x\) (lower linearization)
- \(\bar{f} \geq f(\bar{x})\) (upper bound, may be \(+\infty\))

Can be called **repeatedly** on the same \(\bar{x}\).

- Different rules governing the produced sequences \(\{f_j\}, \{\bar{f}_j\}\)
- Bundle algorithm works in different “modes” (LB/UB following)

---

Approximate Bundle Methods

- Proposal exist only using lower bound or for finite min-max.
- Unify and extend these.

**Definition**

**Incremental inexact oracle** for $f$: inputs $\bar{x} \in \mathbb{R}^n$, outputs:

- $f \leq f(\bar{x})$, $z \in \mathbb{R}^n$ s.t. $f + z(x - \bar{x}) \leq f(x) \ \forall x$ (lower linearization)
- $\bar{f} \geq f(\bar{x})$ (upper bound, may be $+\infty$)

Can be called repeatedly on the same $\bar{x}$.

- Different rules governing the produced sequences $\{f_j\}$, $\{\bar{f}_j\}$
- Bundle algorithm works in different “modes” (LB/UB following)

Results still preliminary, but knowing the gap helps

---

Conclusions

Conclusions

- Errors are a fact of life

---

Errors are a fact of life

You can pretend they don’t exist, but you’re better off not to
Conclusions

- Errors are a fact of life
- You can pretend they don’t exist, but you’re better off not to
- Knowing something about them helps

---

Conclusions

- Errors are a fact of life
- You can pretend they don’t exist, but you’re better off not to
- Knowing something about them helps
- Errors may even be a good thing

---

Conclusions

- Errors are a fact of life

- You can pretend they don’t exist, but you’re better off not to

- Knowing something about them helps

- Errors may even be a good thing

- Lots of work still to be done
  - incremental subgradient
  - “dual” subgradient convergence
  - incremental bundle
  - software development/refinement, numerical testing

---