Ottimizzazione dei Sistemi Complessi

G. Liuzzi[1]

Giovedì 28 Febbraio 2019

[1] Istituto di Analisi dei Sistemi ed Informatica IASI - CNR

Pseudo-code di "compass search"

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$

  Let $\bar{d} \in D$ be s.t. $f(x + \Delta \bar{d}) = \min_{d_i \in D} f(x + \Delta d_i)$

  if $f(x + \Delta \bar{d}) < f(x)$ **then**

    $x \leftarrow x + \Delta \bar{d}$

    endif

  end for

  if $f(y) < f(x)$ **then**

    $x \leftarrow y$

  else

    $\Delta \leftarrow \Delta/2$

  endif

end while

RETURN: x (miglior punto determinato)

Pseudo-code di "compass search" rivisto o debole

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \ldots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$

  if $\exists\ \bar{d} \in D$ s.t. $f(x + \Delta\bar{d}) < f(x)$ **then**

    $x \leftarrow x + \Delta\bar{d}$

    endif

  end for

  if $f(y) < f(x)$ **then**

    $x \leftarrow y$

  else

    $\Delta \leftarrow \Delta/2$

  endif

end while

RETURN: x (miglior punto determinato)

Pseudo-code di "compass search" rivisto o debole

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \ldots, n\}$
$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$
while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**
  $k \leftarrow k + 1$, $y \leftarrow x$

  if $\exists\ \bar{d} \in D$ s.t. $f(y + \Delta\bar{d}) < f(y)$ **then**
    $y \leftarrow y + \Delta\bar{d}$
    endif
  end for
  if $f(y) < f(x)$ **then**
    $x \leftarrow y$
  else
    $\Delta \leftarrow \Delta/2$
  endif
end while
RETURN: x (miglior punto determinato)

Pseudo-code di "compass search" rivisto o debole

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$

  for each $\bar{d} \in D$

    if $f(y + \Delta\bar{d}) < f(y)$ **then**

      $y \leftarrow y + \Delta\bar{d}$, **break**

    endif

  end for

  if $f(y) < f(x)$ **then**

    $x \leftarrow y$

  else

    $\Delta \leftarrow \Delta/2$

  endif

end while

RETURN: x (miglior punto determinato)

Un nuovo metodo

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$

  for each $\bar{d} \in D$

    if $f(y + \Delta\bar{d}) < f(y)$ **then**

      $y \leftarrow y + \Delta\bar{d}$

    endif

  end for

  if $f(y) < f(x)$ **then**

    $x \leftarrow y$

  else

    $\Delta \leftarrow \Delta/2$

  endif

end while

RETURN: x (miglior punto determinato)

Un "nuovo" metodo

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$\quad k \leftarrow k + 1$, $y \leftarrow x$

end while

RETURN: x (miglior punto determinato)

Domande?

Un "nuovo" metodo

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \ldots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$k \leftarrow k + 1$, $y \leftarrow x$

for each $d \in D$

if $f(y + \Delta d) < f(y)$ then $y \leftarrow y + \Delta d$

end for

end while

RETURN: x (miglior punto determinato)

Domande?

Un "nuovo" metodo

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \ldots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$

  for each $d \in D$

    if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$

  end for

  if $f(y) < f(x)$ **then**

    $x \leftarrow y$

  else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Domande?

Un "nuovo" metodo

INPUT: x_0, Δ_0, Δ_{min}, maxit, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq$maxit **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$
  for each $d \in D$
    if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
  end for
  if $f(y) < f(x)$ **then**
    $x \leftarrow y$
  else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Domande?

Un "nuovo" metodo

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$
  for each $d \in D$
    if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
  end for
  if $f(y) < f(x)$ **then**
    $x \leftarrow y$
  else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Domande?

Un "nuovo" metodo

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \ldots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$k \leftarrow k + 1$, $y \leftarrow x$
for each $d \in D$
if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
end for
if $f(y) < f(x)$ **then**
$x \leftarrow y$
else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Domande?

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$k \leftarrow k + 1$, $y \leftarrow x$

end while

RETURN: x (miglior punto determinato)

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$k \leftarrow k + 1$, $y \leftarrow x$

for each $d \in D$ (*exploratory moves from* x)

if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$

end for

end while

RETURN: x (miglior punto determinato)

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$k \leftarrow k + 1$, $y \leftarrow x$

for each $d \in D$ (*exploratory moves from x*)

if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$

end for

if $f(y) < f(x)$ **then**

else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

> $k \leftarrow k + 1$, $y \leftarrow x$
>
> **for each** $d \in D$ (*exploratory moves from x*)
>
> > **if** $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
>
> **end for**
>
> **if** $f(y) < f(x)$ **then** (*pattern move along $y - x$*)
>
> > $z \leftarrow y + (y - x)$
>
> **else** $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, maxit, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq$maxit **and** $\Delta \geq \Delta_{min}$ **do**

  $k \leftarrow k + 1$, $y \leftarrow x$
  for each $d \in D$ (*exploratory moves from x*)
    if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
  end for
  if $f(y) < f(x)$ **then** (*pattern move along $y - x$*)
    $z \leftarrow y + (y - x)$
    for each $d \in D$ (*exploratory moves from z*)
      if $f(z + \Delta d) < f(z)$ **then** $z \leftarrow z + \Delta d$
    end for
    if $f(z) < f(y)$ **then** $x \leftarrow z$ **else** $x \leftarrow y$
  else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq \mathtt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

> $k \leftarrow k + 1$, $y \leftarrow x$
> **for each** $d \in D$ (*exploratory moves from x*)
>   **if** $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
> **end for**
> **if** $f(y) < f(x)$ **then** (*pattern move along $y - x$*)
> $z \leftarrow y + (y - x)$
>   **for each** $d \in D$ (*exploratory moves from z*)
>   **if** $f(z + \Delta d) < f(z)$ **then** $z \leftarrow z + \Delta d$
>   **end for**
>   **if** $f(z) < f(y)$ **then** $x \leftarrow z$ **else** $x \leftarrow y$
> **else** $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Il metodo di Hooke&Jeeves

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$

while $k \leq$ $\texttt{maxit}$ **and** $\Delta \geq \Delta_{min}$ **do**

$k \leftarrow k + 1$, $y \leftarrow x$
for each $d \in D$ (*exploratory moves from x*)
if $f(y + \Delta d) < f(y)$ **then** $y \leftarrow y + \Delta d$
end for
if $f(y) < f(x)$ **then** (*pattern move along y − x*)
$z \leftarrow y + (y - x)$
for each $d \in D$ (*exploratory moves from z*)
if $f(z + \Delta d) < f(z)$ **then** $z \leftarrow z + \Delta d$
end for
if $f(z) < f(y)$ **then** $x \leftarrow z$ **else** $x \leftarrow y$
else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)

Convergenza di Hooke&Jeeves

Nota bene: Come per il "compass search"

- Se il passo non viene più aggiornato, e.g. $\Delta_{k+1} = \Delta_k$, tutti i punti x_k, da un k in poi, appartengono ad una "griglia";
- Se il punto non cambia, e.g. $x_{k+1} = x_k$,

$$f(x_k + \Delta_k d) \geq f(x_k) \text{ per ogni } d \in D.$$

Quindi, se $L(x_0)$ è compatto e f è cont. differenziabile:

- $\lim_{k \to \infty} \Delta_k = 0$
- almeno un punto limite è stazionario
  - $\nabla f(\bar{x}) = 0$
  - $\liminf_{k \to \infty} \|\nabla f(x_k)\| = 0$

Convergenza di Hooke&Jeeves

Nota bene: Come per il "compass search"

- Se il passo non viene più aggiornato, e.g. $\Delta_{k+1} = \Delta_k$, tutti i punti x_k, da un k in poi, appartengono ad una "griglia";
- Se il punto non cambia, e.g. $x_{k+1} = x_k$,

$$f(x_k + \Delta_k d) \geq f(x_k) \text{ per ogni } d \in D.$$

Quindi, se $L(x_0)$ è compatto e f è cont. differenziabile:

- $\lim_{k\to\infty} \Delta_k = 0$
- almeno un punto limite è stazionario
  - $\nabla f(\bar{x}) = 0$
  - $\liminf_{k\to\infty} \|\nabla f(x_k)\| = 0$

Convergenza di Hooke&Jeeves

Nota bene: Come per il "compass search"

- Se il passo non viene più aggiornato, e.g. $\Delta_{k+1} = \Delta_k$, tutti i punti x_k, da un k in poi, appartengono ad una "griglia";
- Se il punto non cambia, e.g. $x_{k+1} = x_k$,

$$f(x_k + \Delta_k d) \geq f(x_k) \text{ per ogni } d \in D.$$

Quindi, se $L(x_0)$ è compatto e f è cont. differenziabile:

- $\lim_{k\to\infty} \Delta_k = 0$
- almeno un punto limite è stazionario
  - $\nabla f(\bar{x}) = 0$
  - $\liminf_{k\to\infty} \|\nabla f(x_k)\| = 0$

Un metodo generale

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

while $k \leq \texttt{maxit}$ **and** $\Delta_k \geq \Delta_{min}$ **do**

if $\boxed{\exists\ y_k \text{ s.t. } f(y_k) \leq f(x_k) - \gamma\Delta_k}$

$\Delta_{k+1} \leftarrow \Delta_k$

elseif $\boxed{\exists\ \bar{d} \in D \text{ s.t. } f(x_k + \Delta_k \bar{d}) \leq f(x_k) - \gamma\Delta_k^2}$

$y_k \leftarrow x_k + \Delta_k \bar{d}$, $\Delta_{k+1} \leftarrow \Delta_k$

else

$y_k \leftarrow x_k$, $\Delta_{k+1} \leftarrow \Delta_k/2$

endif

$\boxed{\text{Find } x_{k+1} \text{ s.t. } f(x_{k+1}) \leq f(y_k)}$

$k \leftarrow k + 1$

end while

RETURN: $\{x_k\}, \{\Delta_k\}$ successioni di punti e passi

Un metodo generale

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

while $k \leq \texttt{maxit}$ **and** $\Delta_k \geq \Delta_{min}$ **do**

if $\boxed{\exists\ y_k \text{ s.t. } f(y_k) \leq f(x_k) - \gamma\Delta_k}$

$\Delta_{k+1} \leftarrow \Delta_k$

elseif $\boxed{\exists\ \bar{d} \in D \text{ s.t. } f(x_k + \Delta_k \bar{d}) \leq f(x_k) - \gamma\Delta_k^2}$

$y_k \leftarrow x_k + \Delta_k \bar{d}$, $\Delta_{k+1} \leftarrow \Delta_k$

else

$y_k \leftarrow x_k$, $\Delta_{k+1} \leftarrow \Delta_k/2$

endif

$\boxed{\text{Find } x_{k+1} \text{ s.t. } f(x_{k+1}) \leq f(y_k)}$

$k \leftarrow k + 1$

end while

RETURN: $\{x_k\}, \{\Delta_k\}$ successioni di punti e passi

Un metodo generale

INPUT: x_0, Δ_0, Δ_{min}, $\texttt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

while $k \leq \texttt{maxit}$ **and** $\Delta_k \geq \Delta_{min}$ **do**

if $\boxed{\exists\ y_k \text{ s.t. } f(y_k) \leq f(x_k) - \gamma\Delta_k}$

$\Delta_{k+1} \leftarrow \Delta_k$

elseif $\boxed{\exists\ \bar{d} \in D \text{ s.t. } f(x_k + \Delta_k \bar{d}) \leq f(x_k) - \gamma\Delta_k^2}$

$y_k \leftarrow x_k + \Delta_k \bar{d}$, $\Delta_{k+1} \leftarrow \Delta_k$

else

$y_k \leftarrow x_k$, $\Delta_{k+1} \leftarrow \Delta_k/2$

endif

$\boxed{\text{Find } x_{k+1} \text{ s.t. } f(x_{k+1}) \leq f(y_k)}$

$k \leftarrow k + 1$

end while

RETURN: $\{x_k\}, \{\Delta_k\}$ successioni di punti e passi

Un metodo generale

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

while $k \leq$maxit **and** $\Delta_k \geq \Delta_{min}$ **do**

if $\boxed{\exists\ y_k \text{ s.t. } f(y_k) \leq f(x_k) - \gamma\Delta_k}$

$\Delta_{k+1} \leftarrow \Delta_k$

elseif $\boxed{\exists\ \bar{d} \in D \text{ s.t. } f(x_k + \Delta_k \bar{d}) \leq f(x_k) - \gamma\Delta_k^2}$

$y_k \leftarrow x_k + \Delta_k \bar{d}$, $\Delta_{k+1} \leftarrow \Delta_k$

else

$y_k \leftarrow x_k$, $\Delta_{k+1} \leftarrow \Delta_k/2$

endif

$\boxed{\text{Find } x_{k+1} \text{ s.t. } f(x_{k+1}) \leq f(y_k)}$

$k \leftarrow k + 1$

end while

RETURN: $\{x_k\}, \{\Delta_k\}$ successioni di punti e passi

Un metodo generale

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

while $k \leq \mathtt{maxit}$ **and** $\Delta_k \geq \Delta_{min}$ **do**

if $\boxed{\exists\ y_k \text{ s.t. } f(y_k) \leq f(x_k) - \gamma\Delta_k}$

$\Delta_{k+1} \leftarrow \Delta_k$

elseif $\boxed{\exists\ \bar{d} \in D\ \alpha \geq \Delta_k \text{ s.t. } f(x_k + \Delta_k \bar{d}) \leq f(x_k)\ - \gamma\Delta_k^2}$

$y_k \leftarrow x_k + \Delta_k \bar{d}$, $\Delta_{k+1} \leftarrow \Delta_k$

else

$y_k \leftarrow x_k$, $\Delta_{k+1} \leftarrow \Delta_k / 2$

endif

$\boxed{\text{Find } x_{k+1} \text{ s.t. } f(x_{k+1}) \leq f(y_k)}$

$k \leftarrow k + 1$

end while

RETURN: $\{x_k\}, \{\Delta_k\}$ successioni di punti e passi

Un metodo generale

INPUT: x_0, Δ_0, Δ_{min}, $\mathtt{maxit}$, $D = \{\pm e_i, i = 1, \ldots, n\}$

$k \leftarrow 0$

while $k \leq \mathtt{maxit}$ **and** $\Delta_k \geq \Delta_{min}$ **do**

if $\boxed{\exists\ y_k \text{ s.t. } f(y_k) \leq f(x_k) - \gamma\Delta_k}$

$\Delta_{k+1} \leftarrow \Delta_k$

elseif $\boxed{\exists\ \bar{d} \in D\ \alpha \geq \Delta_k \text{ s.t. } f(x_k + \Delta_k \bar{d}) \leq f(x_k)\ - \gamma\alpha^2}$

$y_k \leftarrow x_k + \alpha\bar{d}$, $\Delta_{k+1} \leftarrow \alpha$

else

$y_k \leftarrow x_k$, $\Delta_{k+1} \leftarrow \Delta_k/2$

endif

$\boxed{\text{Find } x_{k+1} \text{ s.t. } f(x_{k+1}) \leq f(y_k)}$

$k \leftarrow k + 1$

end while

RETURN: $\{x_k\}, \{\Delta_k\}$ successioni di punti e passi