

Ottimizzazione dei Sistemi Complessi

G. Liuzzi¹

Venerddì 16 Marzo 2018

¹Istituto di Analisi dei Sistemi ed Informatica IASI - CNR



Iterazione k

- Se $f_1 \leq f^r < f_n$, allora $X_{k+1} = X_k \setminus \{x_{n+1}\} \cup \{x^r\}$ **FINE**
- Se $f^r < f_1$, allora
 - Se $f^e < f^r$, allora $X_{k+1} = X_k \setminus \{x_{n+1}\} \cup \{x^e\}$ **FINE**
 - altrimenti $X_{k+1} = X_k \setminus \{x_{n+1}\} \cup \{x^r\}$ **FINE**
- Se $f_n \leq f^r < f_{n+1}$, allora
 - Se $f^{oc} \leq f^r$, allora $X_{k+1} = X_k \setminus \{x_{n+1}\} \cup \{x^{oc}\}$ **FINE**
 - altrimenti **shrink**
- Se $f_{n+1} \leq f^r$, allora
 - Se $f^{ic} < f_{n+1}$, allora $X_{k+1} = X_k \setminus \{x_{n+1}\} \cup \{x^{ic}\}$ **FINE**
 - altrimenti **shrink**
- **shrink:**
 - $X_{k+1} = \{x_1, \hat{x}_2, \dots, \hat{x}_{n+1}\}$ dove
 - $\hat{x}_i = x_1 + \gamma(x_i - x_1)$, $i = 2, \dots, n+1$, $\gamma \in (0, 1)$



Esercizio

Sia

$$X = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\}$$
$$F = \{ 3, 7, 1, \}$$

Il centroide \bar{x} di x_1 e x_2 (dopo il riordino) è

$$\bar{x} = \begin{pmatrix} 1 \\ 1/2 \end{pmatrix}$$

Il metodo considera i punti

$$x(\mu) = \bar{x} + \mu d = \bar{x} + \mu(\bar{x} - x_3)$$

per valori $\mu = 1, 2, 1/2, -1/2$



Esercizio

Sia

$$X = \left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$$
$$F = \{ 1, 3, 7, \}$$

Il centroide \bar{x} di x_1 e x_2 (dopo il riordino) è

$$\bar{x} = \begin{pmatrix} 1 \\ 1/2 \end{pmatrix}$$

Il metodo considera i punti

$$x(\mu) = \bar{x} + \mu d = \bar{x} + \mu(\bar{x} - x_3)$$

per valori $\mu = 1, 2, 1/2, -1/2$ 

Esercizio

Sia

$$X = \left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$$
$$F = \{ 1, 3, 7, \}$$

Il centroide \bar{x} di x_1 e x_2 (dopo il riordino) è

$$\bar{x} = \begin{pmatrix} 1 \\ 1/2 \end{pmatrix}$$

Il metodo considera i punti

$$x(\mu) = \bar{x} + \mu d = \bar{x} + \mu(\bar{x} - x_3)$$

per valori $\mu = 1, 2, 1/2, -1/2$



Esercizio

Sia

$$X = \left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$$
$$F = \{ 1, 3, 7, \}$$

Il centroide \bar{x} di x_1 e x_2 (dopo il riordino) è

$$\bar{x} = \begin{pmatrix} 1 \\ 1/2 \end{pmatrix}$$

Il metodo considera i punti

$$x(\mu) = \bar{x} + \mu d = \bar{x} + \mu(\bar{x} - x_3)$$

per valori $\mu = 1, 2, 1/2, -1/2$



Esercizio

- si calcola $d = \bar{x} - x_3 = (1, -1/2)^T$
- si calcola $x_r = x(1) = \bar{x} + d = (2, 0)^T$
- si calcola $x_e = x(2) = \bar{x} + 2d = (3, -1/2)^T$
- si calcola $x_{oc} = x(1/2) = \bar{x} + 1/2d = (3/2, 1/4)^T$
- si calcola $x_{ic} = x(-1/2) = \bar{x} - 1/2d = (1/2, 3/4)^T$



Esercizio

- si calcola $d = \bar{x} - x_3 = (1, -1/2)^T$
- si calcola $x_r = x(1) = \bar{x} + d = (2, 0)^T$
- si calcola $x_e = x(2) = \bar{x} + 2d = (3, -1/2)^T$
- si calcola $x_{oc} = x(1/2) = \bar{x} + 1/2d = (3/2, 1/4)^T$
- si calcola $x_{ic} = x(-1/2) = \bar{x} - 1/2d = (1/2, 3/4)^T$



Esercizio

- si calcola $d = \bar{x} - x_3 = (1, -1/2)^T$
- si calcola $x_r = x(1) = \bar{x} + d = (2, 0)^T$
- si calcola $x_e = x(2) = \bar{x} + 2d = (3, -1/2)^T$
- si calcola $x_{oc} = x(1/2) = \bar{x} + 1/2d = (3/2, 1/4)^T$
- si calcola $x_{ic} = x(-1/2) = \bar{x} - 1/2d = (1/2, 3/4)^T$



Esercizio

- si calcola $d = \bar{x} - x_3 = (1, -1/2)^T$
- si calcola $x_r = x(1) = \bar{x} + d = (2, 0)^T$
- si calcola $x_e = x(2) = \bar{x} + 2d = (3, -1/2)^T$
- si calcola $x_{oc} = x(1/2) = \bar{x} + 1/2d = (3/2, 1/4)^T$
- si calcola $x_{ic} = x(-1/2) = \bar{x} - 1/2d = (1/2, 3/4)^T$



Esercizio

- si calcola $d = \bar{x} - x_3 = (1, -1/2)^T$
- si calcola $x_r = x(1) = \bar{x} + d = (2, 0)^T$
- si calcola $x_e = x(2) = \bar{x} + 2d = (3, -1/2)^T$
- si calcola $x_{oc} = x(1/2) = \bar{x} + 1/2d = (3/2, 1/4)^T$
- si calcola $x_{ic} = x(-1/2) = \bar{x} - 1/2d = (1/2, 3/4)^T$



Nelder&Mead

In \mathbb{R}^2 siano dati i seguenti punti:

$$x_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, x_2 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}, x_3 = \begin{pmatrix} 1 \\ 3 \end{pmatrix},$$

a cui corrispondono: $f(x_1) = 1$, $f(x_2) = 3$, $f(x_3) = 5$.

Determinare i punti x_r , x_e , x_{oc} e x_{ic} utilizzati nel metodo di Nelder&Mead



Nelder&Mead

In \mathbb{R}^2 siano dati i seguenti punti:

$$x_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, x_2 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}, x_3 = \begin{pmatrix} 1 \\ 3 \end{pmatrix},$$

a cui corrispondono: $f(x_1) = 1$, $f(x_2) = 3$, $f(x_3) = 5$.

Determinare i punti x_r , x_e , x_{oc} e x_{ic} utilizzati nel metodo di Nelder&Mead



È un “buon” metodo?

La risposta è:

- generalmente **Si**
- però non ha proprietà di convergenza a punti stazionari
- esistono contro-esempi sui quali converge a punti **NON** stazionari
- p.es. la funzione di McKinnon



È un “buon” metodo?

La risposta è:

- generalmente **Si**
- però non ha proprietà di convergenza a punti stazionari
- esistono contro-esempi sui quali converge a punti **NON** stazionari
- p.es. la funzione di McKinnon



È un “buon” metodo?

La risposta è:

- generalmente **Si**
- però non ha proprietà di convergenza a punti stazionari
- esistono contro-esempi sui quali converge a punti **NON** stazionari
- p.es. la funzione di McKinnon



La funzione di McKinnon

$$f(x, y) = \begin{cases} \theta\phi|x|^\tau + y + y^2 & \text{se } x \leq 0 \\ \theta x^\tau + y + y^2 & \text{se } x > 0 \end{cases}$$

- strettamente convessa per $\tau > 1$
- continuamente differenziabile per $\tau > 1$
- due volte cont. differenziabile per $\tau > 2$
- tre volte cont. differenziabile per $\tau > 3$



La funzione di McKinnon

$$f(x, y) = \begin{cases} \theta\phi|x|^\tau + y + y^2 & \text{se } x \leq 0 \\ \theta x^\tau + y + y^2 & \text{se } x > 0 \end{cases}$$

- strettamente convessa per $\tau > 1$
- continuamente differenziabile per $\tau > 1$
- due volte cont. differenziabile per $\tau > 2$
- tre volte cont. differenziabile per $\tau > 3$



La funzione di McKinnon

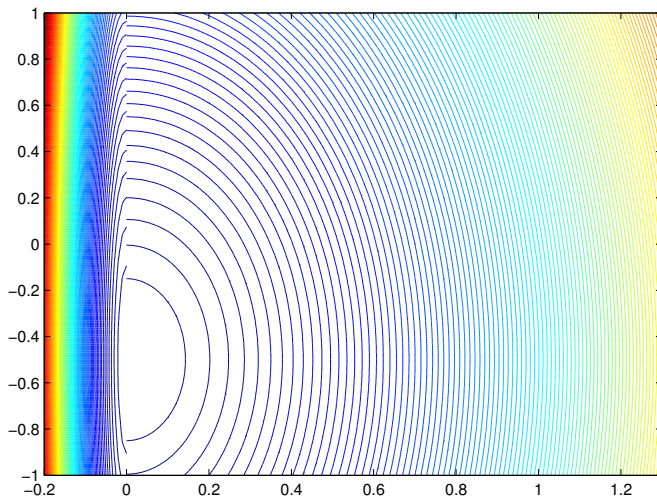
$$f(x, y) = \begin{cases} \theta\phi|x|^\tau + y + y^2 & \text{se } x \leq 0 \\ \theta x^\tau + y + y^2 & \text{se } x > 0 \end{cases}$$

- strettamente convessa per $\tau > 1$
- continuamente differenziabile per $\tau > 1$
- due volte cont. differenziabile per $\tau > 2$
- tre volte cont. differenziabile per $\tau > 3$



La funzione di McKinnon

Per $\tau = 2$, $\theta = 6$, $\phi = 60$, la funzione è



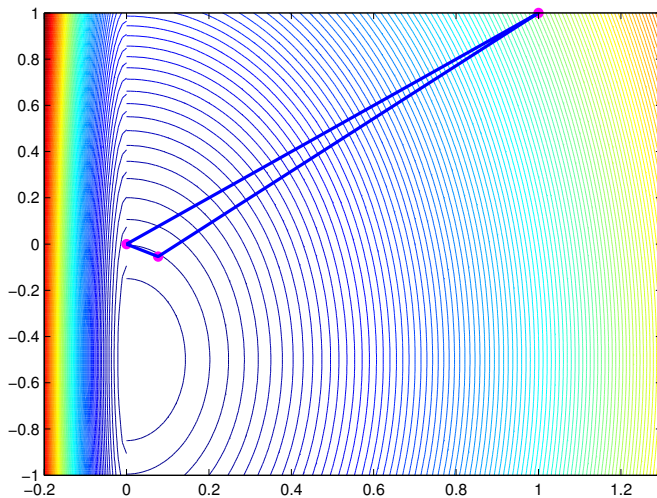
La funzione di McKinnon

Se inizializziamo il metodo di Nelder-Mead con il semplice

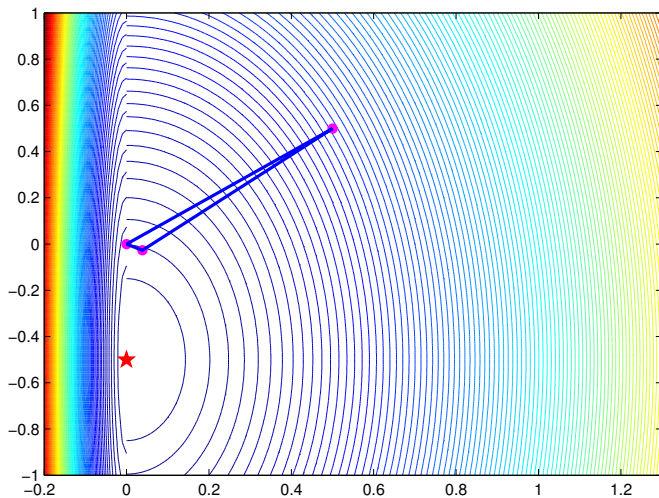
$$X = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} \frac{1+\sqrt{33}}{88} \\ \frac{1-\sqrt{33}}{88} \end{pmatrix}, \right\}$$



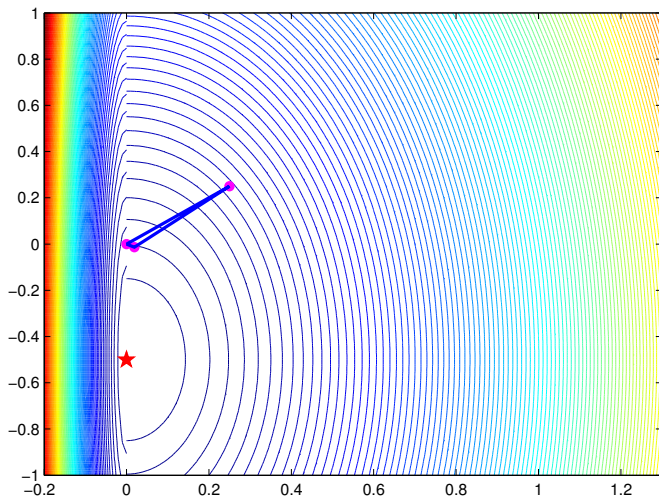
La funzione di McKinnon



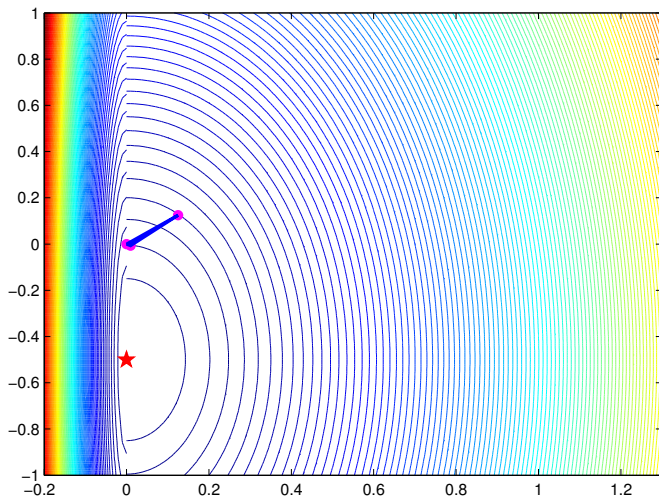
La funzione di McKinnon



La funzione di McKinnon



La funzione di McKinnon



Perchè...

- Compass Search “funziona” ?
- Hooke & Jeeves “funziona” ?
- Nelder & Mead “**non** funziona” (teoricamente) ?

Cosa “davvero” distingue i primi due dal terzo ?



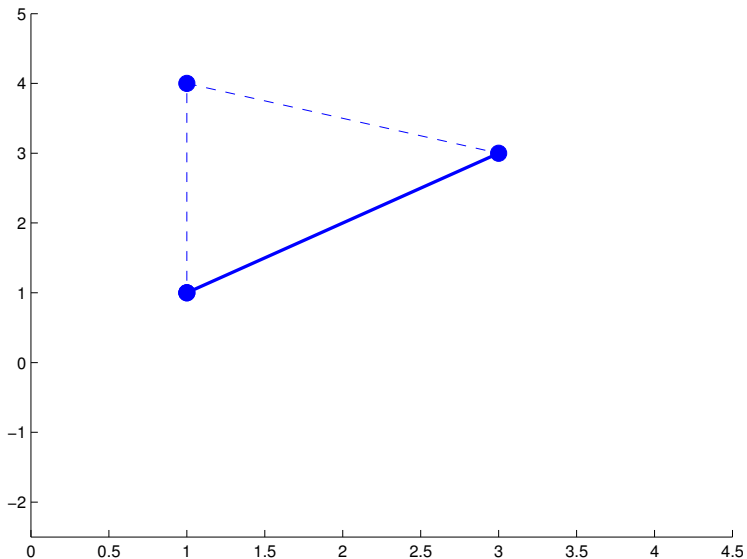
Perchè...

- Compass Search “funziona” ?
- Hooke & Jeeves “funziona” ?
- Nelder & Mead “**non** funziona” (teoricamente) ?

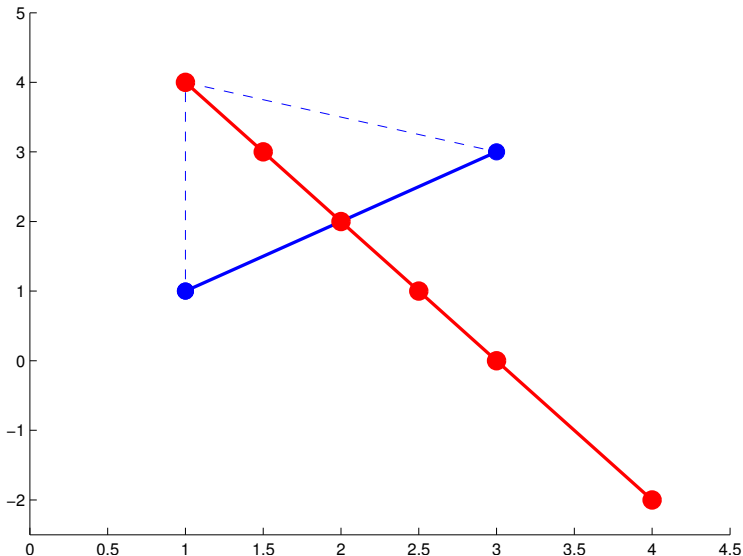
Cosa “davvero” distingue i primi due dal terzo ?



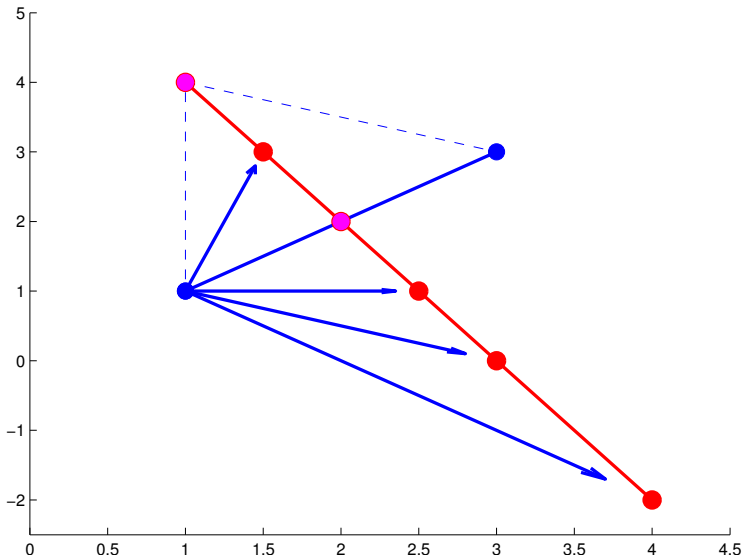
Perchè...



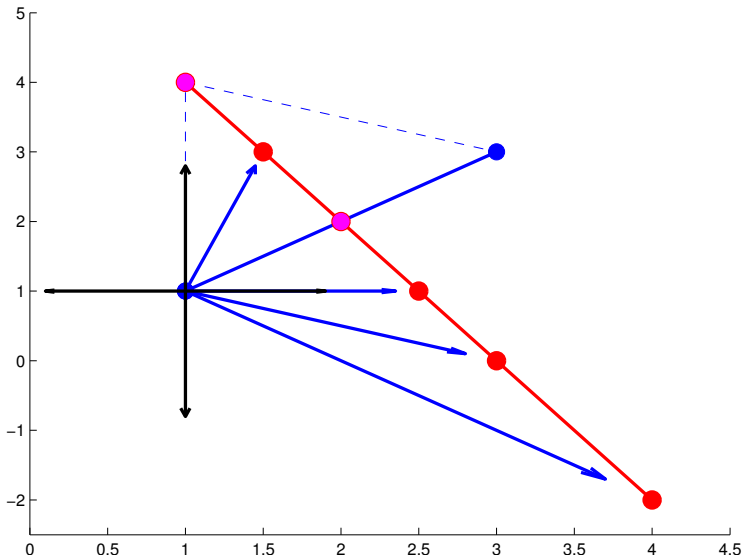
Perchè...



Perchè...



Perchè...



... e quindi ...

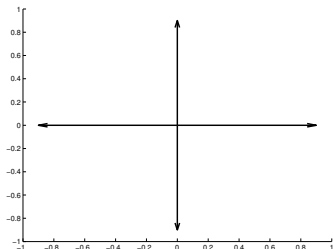
Quando $D = \{\pm e_1, \pm e_2, \dots, \pm e_n\}$

$$\kappa(D) = \frac{1}{\sqrt{n}} > 0$$

Comunque scelto un $v \in \mathbb{R}^n$, esiste $d \in D$ tale che

$$\frac{v^T d}{\|v\| \|d\|} = \cos \theta(d, v) \geq \kappa(D) > 0$$

Quando $v = -\nabla f(x) \neq 0$, esiste sempre una direzione $d \in D$ **di discesa**



... e quindi ...

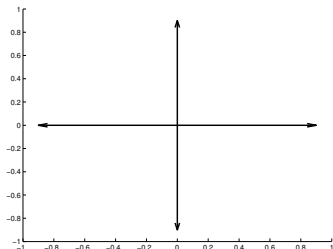
Quando $D = \{\pm e_1, \pm e_2, \dots, \pm e_n\}$

$$\kappa(D) = \frac{1}{\sqrt{n}} > 0$$

Comunque scelto un $v \in \mathbb{R}^n$, esiste $d \in D$ tale che

$$\frac{v^\top d}{\|v\| \|d\|} = \cos \theta(d, v) \geq \kappa(D) > 0$$

Quando $v = -\nabla f(x) \neq 0$, esiste sempre una direzione $d \in D$ **di discesa**



... e quindi ...

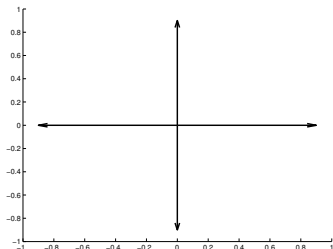
Quando $D = \{\pm e_1, \pm e_2, \dots, \pm e_n\}$

$$\kappa(D) = \frac{1}{\sqrt{n}} > 0$$

Comunque scelto un $v \in \mathbb{R}^n$, esiste $d \in D$ tale che

$$\frac{v^\top d}{\|v\| \|d\|} = \cos \theta(d, v) \geq \kappa(D) > 0$$

Quando $v = -\nabla f(x) \neq 0$, esiste sempre una direzione $d \in D$ **di discesa**



... e quindi ...

Quando $D = \{\text{vedi figura}\}$

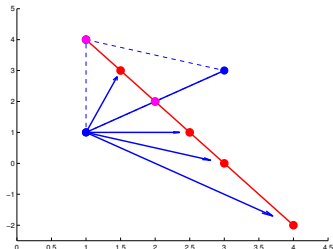
$$\kappa(D) \leq 0$$

esistono $v \in \mathbb{R}^n$ tali che, per ogni $d \in D$,
si ha

$$\frac{v^T d}{\|v\| \|d\|} = \cos \theta(d, v) \leq 0$$

Potrebbe capitare che $\nabla f(x) \neq 0$, e ogni
 $d \in D$ è tale che

$$-\nabla f(x)^T d \leq 0$$

cioè **nessuna** dir. in D è **di discesa**

... e quindi ...

Quando $D = \{\text{vedi figura}\}$

$$\kappa(D) \leq 0$$

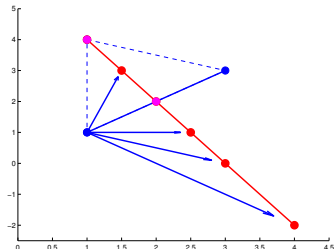
esistono $v \in \mathbb{R}^n$ tali che, per ogni $d \in D$,
si ha

$$\frac{v^\top d}{\|v\| \|d\|} = \cos \theta(d, v) \leq 0$$

Potrebbe capitare che $\nabla f(x) \neq 0$, e ogni
 $d \in D$ è tale che

$$-\nabla f(x)^\top d \leq 0$$

cioè **nessuna** dir. in D è **di discesa**



... e quindi ...

Quando $D = \{\text{vedi figura}\}$

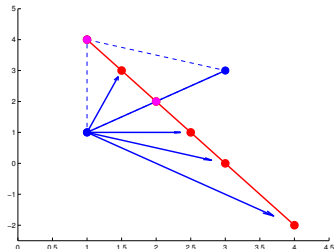
$$\kappa(D) \leq 0$$

esistono $v \in \mathbb{R}^n$ tali che, per ogni $d \in D$,
si ha

$$\frac{v^\top d}{\|v\| \|d\|} = \cos \theta(d, v) \leq 0$$

Potrebbe capitare che $\nabla f(x) \neq 0$, e ogni
 $d \in D$ è tale che

$$-\nabla f(x)^\top d \leq 0$$

cioè **nessuna** dir. in D è **di discesa**

... e quindi ...

per il metodo **compass search** che usa

- $D = \{\pm e_1, \dots, \pm e_n\}$,
- $\kappa(D) = 1/\sqrt{(n)} > 0$

quando il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- esiste sempre (almeno) una direzione $d \in D$ che è **di discesa**

$$\nabla f(x)^\top d < 0$$

- per passi Δ sufficientemente piccoli **deve accadere**

$$f(x + \Delta d) < f(x)$$

cioè il punto x corrente (se non-stazionario) prima o dopo viene aggiornato



... e quindi ...

per il metodo **compass search** che usa

- $D = \{\pm e_1, \dots, \pm e_n\}$,
- $\kappa(D) = 1/\sqrt{(n)} > 0$

quando il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- esiste sempre (almeno) una direzione $d \in D$ che è **di discesa**

$$\nabla f(x)^T d < 0$$

- per passi Δ sufficientemente piccoli **deve accadere**

$$f(x + \Delta d) < f(x)$$

cioè il punto x corrente (se non-stazionario) prima o dopo viene aggiornato



... e quindi ...

per il metodo **compass search** che usa

- $D = \{\pm e_1, \dots, \pm e_n\}$,
- $\kappa(D) = 1/\sqrt{(n)} > 0$

quando il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- esiste sempre (almeno) una direzione $d \in D$ che è **di discesa**

$$\nabla f(x)^\top d < 0$$

- per passi Δ sufficientemente piccoli **deve accadere**

$$f(x + \Delta d) < f(x)$$

cioè il punto x corrente (se non-stazionario) prima o dopo viene aggiornato



... e quindi ...

per il metodo **compass search** che usa

- $D = \{\pm e_1, \dots, \pm e_n\}$,
- $\kappa(D) = 1/\sqrt{(n)} > 0$

quando il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- esiste sempre (almeno) una direzione $d \in D$ che è **di discesa**

$$\nabla f(x)^\top d < 0$$

- per passi Δ sufficientemente piccoli **deve accadere**

$$f(x + \Delta d) < f(x)$$

cioè il punto x corrente (se non-stazionario) prima o dopo viene aggiornato



... e quindi ...

per il metodo **Nelder&Mead** che usa

- D t.c. $\kappa(D) \leq 0$

anche se il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- le direzioni $d \in D$ potrebbero tendere a direzioni \bar{d} che **non** sono di discesa

$$\nabla f(x)^\top \bar{d} \geq 0$$

- anche per spostamenti piccolissimi (i.e. punti in X vicinissimi) nessuno dei punti x_r, x_e, x_{oc}, x_{ic} migliora il valore $f(x)$

cioè il punto x corrente non-stazionario non viene **mai** aggiornato!!



... e quindi ...

per il metodo **Nelder&Mead** che usa

- D t.c. $\kappa(D) \leq 0$

anche se il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- le direzioni $d \in D$ potrebbero tendere a direzioni \bar{d} che **non** sono di discesa

$$\nabla f(x)^\top \bar{d} \geq 0$$

- anche per spostamenti piccolissimi (i.e. punti in X vicinissimi) nessuno dei punti x_r, x_e, x_{oc}, x_{ic} migliora il valore $f(x)$

cioè il punto x corrente non-stazionario non viene **mai** aggiornato!!



... e quindi ...

per il metodo **Nelder&Mead** che usa

- D t.c. $\kappa(D) \leq 0$

anche se il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- le direzioni $d \in D$ potrebbero tendere a direzioni \bar{d} che **non** sono di discesa

$$\nabla f(x)^\top \bar{d} \geq 0$$

- anche per spostamenti piccolissimi (i.e. punti in X vicinissimi) nessuno dei punti x_r, x_e, x_{oc}, x_{ic} migliora il valore $f(x)$

cioè il punto x corrente non-stazionario non viene **mai** aggiornato!!



... e quindi ...

per il metodo **Nelder&Mead** che usa

- D t.c. $\kappa(D) \leq 0$

anche se il punto corrente **non** è stazionario ($\nabla f(x) \neq 0$)

- le direzioni $d \in D$ potrebbero tendere a direzioni \bar{d} che **non** sono di discesa

$$\nabla f(x)^\top \bar{d} \geq 0$$

- anche per spostamenti piccolissimi (i.e. punti in X vicinissimi) nessuno dei punti x_r, x_e, x_{oc}, x_{ic} migliora il valore $f(x)$

cioè il punto x corrente non-stazionario non viene **mai** aggiornato!!



Pseudo-code di “compass search”

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**

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

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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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

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

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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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

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

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:  $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$ 
 $k \leftarrow 0, x \leftarrow x_0, \Delta \leftarrow \Delta_0$ 
while  $k \leq \text{maxit}$  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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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

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

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

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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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

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

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

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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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?



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

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

end while

RETURN: x (miglior punto determinato)



Il metodo di Hooke&Jeeves

INPUT: $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$

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

while $k \leq \text{maxit}$ **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: $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$

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

while $k \leq \text{maxit}$ **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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$

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

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*)

$x \leftarrow y + \gamma(y - x)$

else $\Delta \leftarrow \Delta/2$

end while

RETURN: x (miglior punto determinato)



Il metodo di Hooke&Jeeves

```
INPUT:  $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$   
 $k \leftarrow 0, x \leftarrow x_0, \Delta \leftarrow \Delta_0$   
while  $k \leq \text{maxit}$  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} , $maxit$, $D = \{\pm e_i, i = 1, \dots, n\}$
 $k \leftarrow 0$, $x \leftarrow x_0$, $\Delta \leftarrow \Delta_0$
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:  $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$   
 $k \leftarrow 0, x \leftarrow x_0, \Delta \leftarrow \Delta_0$   
while  $k \leq \text{maxit}$  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:

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



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:

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



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:

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



Un metodo generale

INPUT: $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

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

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

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

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

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

else

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

endif

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: $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

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

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

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

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

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

else

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

endif

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: $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

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

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

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

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

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

else

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

endif

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: $x_0, \Delta_0, \Delta_{min}, \text{maxit}, D = \{\pm e_i, i = 1, \dots, n\}$

$k \leftarrow 0$

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

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

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

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

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

else

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

endif

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

