

---

# Metodi Numerici per l'Ingegneria LS

## a.a. 2008-2009

### ● Sistemi Lineari: metodi Iterativi

Fabiana Zama

<http://www.dm.unibo.it/~zama>

[zama@dm.unibo.it](mailto:zama@dm.unibo.it)

# Metodi Iterativi

---

- Metodi utilizzati per risolvere sistemi lineari quando la matrice è **sparsa** e di **grandi dimensioni**.
- La soluzione è ottenuta come limite di una successione convergente: la efficienza è valutata in base alla velocità con cui si riduce l'errore (**velocità di convergenza**).
- Le operazioni che coinvolgono la matrice del problema sono del tipo  $Av$  oppure  $A^T w$  (moltiplicazione fra matrice e vettore). Quindi si tratta di operazioni semplici che preservano la struttura della matrice.
- Per migliorare l'efficienza si introducono tecniche di **accelerazione**.

# Generalità

---

Dato il sistema lineare  $\mathbf{Ax} = \mathbf{b}$  i metodi iterativi ricercano la soluzione mediante una opportuna successione  $\mathbf{x}_{k+1}$  che può avere una delle seguenti forme:

$$\mathbf{x}_{k+1} = \mathbf{H}\mathbf{x}_k + \mathbf{d} \quad \left\{ \begin{array}{l} \text{Metodo di Jacobi} \\ \text{Metodo di Gauss Seidel} \\ \text{Metodi di Rilassamento (SOR, SSOR)} \end{array} \right.$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \quad \left\{ \begin{array}{l} \text{Metodo Gradienti Coniugati} \\ \text{Metodo GMRES} \\ \text{Metodi di Krylov} \end{array} \right.$$

dove  $\mathbf{H} \in \mathbb{C}^{n \times n}$ ,  $\mathbf{d}, \mathbf{p}_k \in \mathbb{C}^n$ ,  $\alpha \in \mathbb{R}$ .

# Convergenza delle successioni

---

Consideriamo il metodo iterativo:

$$\mathbf{x}^{(k+1)} = \mathbf{H}\mathbf{x}^{(k)} + \mathbf{d} \quad (1)$$

per calcolare la soluzione del sistema lineare:

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

con  $\mathbf{A} \in \mathbb{C}^{n \times n}$  non singolare e  $\mathbf{x}^* = \mathbf{A}^{-1}\mathbf{b}$ .

● Il metodo iterativo (??) è detto consistente se

$$\mathbf{x}^* = \mathbf{H}\mathbf{x}^* + \mathbf{d} \quad (2)$$

● Si calcola una successione  $\mathbf{x}_k$  tale che

$$\lim_{k \rightarrow \infty} \mathbf{x}_k = \mathbf{x}^*, \quad \mathbf{x}^* = \mathbf{A}^{-1}\mathbf{b}$$



$$\lim_{k \rightarrow \infty} \|\mathbf{x}_k - \mathbf{x}^*\| = 0$$

# Criteri di Arresto

---

- Errore assoluto:

$$\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\| \leq \tau_a$$

- Errore Relativo:

$$\frac{\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|}{\|\mathbf{x}^{(k)}\|} \leq \tau_r$$

- Residuo:

$$\|\mathbf{b} - \mathbf{Ax}_k\| \leq \tau_b \|\mathbf{b}\|$$

La scelta è legata al metodo e all'efficienza di calcolo.

# Errore Assoluto

---

Si prova la seguente relazione:

$$\|\mathbf{x}^* - \mathbf{x}_k\| \leq \frac{1}{1 - \|\mathbf{H}\|} \|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|$$

Se  $\|\mathbf{H}\| \simeq 1$  l'errore assoluto può essere grande anche se  $\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|$  è  
piccolo.

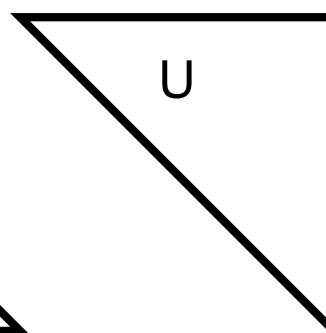
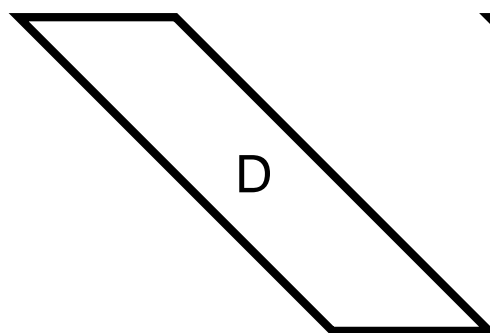
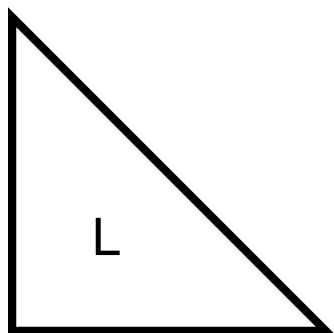
# Metodi Iterativi

Sia  $a_{i,i} \neq 0, i = 1, \dots, n$ . Si decompone la matrice  $\mathbf{A}$  come segue:

$$\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$$

con  $\mathbf{D}$  matrice diagonale  $d_{i,i} = a_{i,i}, i = 1, \dots, n$ ,  $\mathbf{L}, \mathbf{U} \in \mathbb{R}^{n \times n}$  strettamente triangolari cosí definite:

$$\mathbf{L}_{i,j} = \begin{cases} 0 & \text{se } i \leq j \\ -a_{i,j} & \text{se } i > j \end{cases}, \quad \mathbf{U}_{i,j} = \begin{cases} 0 & \text{se } i \geq j \\ -a_{i,j} & \text{se } i < j \end{cases},$$



# Metodo di Jacobi

---

Si definisce  $\mathbf{J} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$  Il metodo iterativo è :

$$\mathbf{x}^{(k+1)} = \mathbf{J}\mathbf{x}^{(k)} + \mathbf{D}^{-1}\mathbf{b}$$

L'aggiornamento delle singole componenti è calcolato come segue:

$$\mathbf{x}_i^{(k+1)} = \frac{1}{a_{i,i}} \left\{ \mathbf{b}_i - \sum_{j=1}^{i-1} a_{i,j} \mathbf{x}_j^{(k)} - \sum_{j=i+1}^n a_{i,j} \mathbf{x}_j^{(k)} \right\}$$

$$i = 1, \dots, n$$



# Algoritmo

---

```
function[x, iter] = jacobi(a, b)
x = zeros(size(b)); n = length(b);
iter = 0; maxiter = n4; x1 = x;
condizione = iter < maxiter;
while condizione
    iter = iter + 1;
    for i = 1 : n
        sl = 0; su = 0;
        if i > 1, sl = a(i, 1 : i - 1) * x(1 : i - 1); end
        if i < n, su = a(i, i + 1 : n) * x(i + 1 : n); end
        x1(i) = (b(i) - sl - su)/a(i, i);
    end
    err = norm(x1 - x); x = x1;
    condizione = err > eps & iter < maxiter;
end
```

# Metodo di Gauss Seidel

---

$\mathbf{G} = (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U}$  Il metodo iterativo è :

$$\mathbf{x}^{(k+1)} = \mathbf{G}\mathbf{x}^{(k)} + (\mathbf{D} - \mathbf{L})^{-1}\mathbf{b}$$

L'aggiornamento delle singole componenti è calcolato come segue:

$$\mathbf{x}_i^{(k+1)} = \frac{1}{a_{i,i}} \left\{ \mathbf{b}_i - \sum_{j=1}^{i-1} a_{i,j} \mathbf{x}_j^{(k+1)} - \sum_{j=i+1}^n a_{i,j} \mathbf{x}_j^{(k)} \right\}$$

$$i = 1, \dots, n$$

# Algoritmo

---

```
function[x, iter] = gseidel(a, b)
x = zeros(size(b)); n = length(b);
iter = 0; maxiter = n4;
condizione = iter < maxiter;
while condizione
    iter = iter + 1;
    for i = 1 : n
        sl = 0; su = 0;
        if i > 1, sl = a(i, 1 : i - 1) * x(1 : i - 1); end
        if i < n, su = a(i, i + 1 : n) * x(i + 1 : n); end
        temp = (b(i) - sl - su)/a(i, i);
        err = err + (temp - x(i))2; x(i) = temp;
    end
    err = sqrt(err);
    condizione = err > tol & iter < maxiter;
end
```

---

# Osservazioni

---

- La complessità computazionale per iterazione è  $\mathcal{O}(n^2)$  se  $\mathbf{A}$  è piena, altrimenti è  $\mathcal{O}(n \times n_r)$  dove  $n_r$  massimo numero di elementi non nulli su ciascuna riga di  $\mathbf{A}$ .
- L'implementazione del metodo di jacobi richiede la memorizzazione di entrambi i vettori  $\mathbf{x}^{(k+1)}$  e  $\mathbf{x}_i^{(k)}$  mentre il metodo di Gauss–Seidel consente di sovrascrivere  $\mathbf{x}^{(k+1)}$  a  $\mathbf{x}^{(k)}$ .

# Convergenza

---

- Condizione necessaria e sufficiente :

Metodo di Jacobi:  $\rho(\mathbf{J}) < 1$

Metodo di Gauss Seidel:  $\rho(\mathbf{G}) < 1$

- Condizioni necessarie:

Metodo di Jacobi:  $\det(\mathbf{J}) < 1, \quad \text{tr}(\mathbf{J}) < n$

Metodo di Gauss Seidel:  $\det(\mathbf{G}) < 1, \quad \text{tr}(\mathbf{G}) < n$

## Esempio 1

$$A = \begin{pmatrix} 3 & 0 & 4 \\ 7 & 4 & 2 \\ -1 & -1 & -2 \end{pmatrix} \quad b = \begin{pmatrix} 7 \\ 13 \\ -4 \end{pmatrix} \quad \mathbf{Ax} = \mathbf{b}$$

Questo sistema lineare ammette soluzione (1, 1, 1, ).

$$J = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 0 & 0 & -4 \\ -7 & 0 & -2 \\ 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & -4/3 \\ -7/4 & 0 & -1/2 \\ -1/2 & -1/2 & 0 \end{pmatrix}$$

$$G = \begin{pmatrix} 3 & & & \\ 7 & 4 & & \\ -1 & -1 & -2 & \end{pmatrix}^{-1} \begin{pmatrix} 0 & 0 & -4 \\ 0 & 0 & -2 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & -4/3 \\ 0 & 0 & 11/6 \\ 0 & 0 & 1/4 \end{pmatrix}$$

$$\rho(J) = 1.34 \quad \rho(G) = 0.25$$

Il metodo di Gauss Seidel converge mentre quello di Jacobi no.

## Esempio 2

$$A = \begin{pmatrix} -3 & 3 & -6 \\ -4 & 7 & -8 \\ 5 & 7 & -9 \end{pmatrix} \quad b = \begin{pmatrix} -6 \\ -5 \\ 3 \end{pmatrix} \quad \mathbf{Ax} = \mathbf{b}$$

Questo sistema lineare ammette soluzione  $(1, 1, 1)$ .

$$J = \begin{pmatrix} -3 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & -9 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 0 & -3 & 6 \\ 4 & 0 & 8 \\ -5 & -7 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & -2 \\ 4/7 & 0 & 8/7 \\ 5/9 & 7/9 & 0 \end{pmatrix}$$

$$G = \begin{pmatrix} -3 & 0 & 0 \\ -4 & 7 & 0 \\ 5 & 7 & -9 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 0 & -3 & 6 \\ 0 & 0 & 8 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & -2 \\ 0.7/4 & 0 & \\ 0 & 1 & -10/9 \end{pmatrix}$$

In questo caso si ha  $\rho(J) = 0.81$  e  $\rho(G) = 1.1$ , quindi il metodo di Jacobi è convergente mentre il metodo di Gauss Seidel no.

## Esempio 3

---

$$A = \begin{pmatrix} 4 & 1 & 1 \\ 2 & -9 & 0 \\ 0 & -8 & -6 \end{pmatrix} \quad b = \begin{pmatrix} 6 \\ -7 \\ -14 \end{pmatrix}$$

- Si ha  $\rho(J) = 0.4$  e  $\rho(G) = 0.019$ . I metodi sono convergenti.
- Se si risolve il sistema lineare  $Ax = b$  partendo dall'iterato iniziale  $x_0 = 0$  usando una tolleranza pari a  $10^{-8}$  si hanno
  - 25 iterazioni con Jacobi,
  - 7 iterazioni con il metodo di Gauss Seidel.
- Dunque Gauss Seidel risulta piú veloce.

Si osserva che  $\rho(G) < \rho(J)$ .



## Esempio 4

---

$$A = \begin{pmatrix} 7 & 6 & 9 \\ 4 & 5 & -4 \\ -7 & -3 & 8 \end{pmatrix} \quad b = \begin{pmatrix} 22 \\ 5 \\ -2 \end{pmatrix}$$

- Si ha  $\rho(J) = 6.41e - 1$  e  $\rho(G) = 7.75e - 1$ .
- Se si risolve il sistema lineare  $Ax = b$  partendo dall'iterato iniziale  $x_0 = 0$  usando una tolleranza pari a  $10^{-8}$  si hanno:
  - 46 iterazioni con Jacobi,
  - 77 iterazioni con il metodo di Gauss Seidel.
- Dunque Jacobi è piú veloce.

Si osserva che  $\rho(J) < \rho(G)$ .

# Velocità di Convergenza (I)

---

La velocità di convergenza è legata al raggio spettrale della matrice di iterazione

$$\left. \begin{array}{l} \mathbf{x}_{k+1} = \mathbf{J}\mathbf{x}_k + \mathbf{d}_j \\ \mathbf{x}_{k+1} = \mathbf{G}\mathbf{x}_k + \mathbf{d}_G \end{array} \right\} \mathbf{x}_{k+1} = \mathbf{H}\mathbf{x}_k + \mathbf{d}$$

$$\lim_{k \rightarrow \infty} \mathbf{x}_k = \mathbf{x}^* \Rightarrow \mathbf{x}^* = \mathbf{H}\mathbf{x}^* + \mathbf{d}$$

$$\mathbf{e}^{(k+1)} = \mathbf{x}^* - \mathbf{x}_{k+1} \Rightarrow \|\mathbf{e}^{(k+1)}\| \leq \|\mathbf{H}^{k+1}\| \|\mathbf{e}^{(0)}\|$$

quindi  $\|\mathbf{H}^{k+1}\|$  esprime la riduzione dell'errore al passo  $k + 1$ -esimo.

## Velocità di Convergenza (II)

---

Se si considera la media geometrica delle riduzioni dell'errore sui primi  $k + 1$  passi si ha:

$$\sigma_{k+1} = \left( \frac{\|e^{(1)}\|}{\|e^{(0)}\|} \frac{\|e^{(2)}\|}{\|e^{(1)}\|} \cdots \frac{\|e^{(k+1)}\|}{\|e^{(k)}\|} \right)^{1/(k+1)} = \left( \frac{\|e^{(k+1)}\|}{\|e^{(0)}\|} \right)^{1/(k+1)}$$

Il valore  $\sigma_{k+1}$  esprime la riduzione media per passo dell'errore iniziale nei primi  $k + 1$  passi.

$$\sigma_{k+1} \leq \left( \|\mathbf{H}^{k+1}\| \right)^{1/(k+1)}$$

Tale misura è inadatta a valutare la velocità di convergenza di un metodo iterativo in quanto dipende dall'indice di iterazione  $k$ .

# Velocità di Convergenza

---

Si prova che:

$$\lim_{k \rightarrow \infty} \sigma_k = \rho(\mathbf{H})$$

Il raggio spettrale non dipende dalle norme utilizzate e dall'indice di iterazione e viene usato come misura della velocità di convergenza del metodo iterativo.

Si definisce **tasso asintotico di convergenza**:

$$R_{\infty} = -\log_{10} \rho(\mathbf{H})$$

## Osservazione

---

Se si vuole stimare il numero di iterazioni richieste affinché  $\|e^{(k)}\| \leq \|e^{(0)}\|/10$  si ha:

$$\left( \frac{\|e^{(k)}\|}{\|e^{(0)}\|} \right)^{1/k} \leq \rho(\mathbf{H})$$

quindi:

$$\rho(\mathbf{H})^k \leq \frac{1}{10} \rightarrow k \simeq -\frac{1}{\log_{10}(\rho(\mathbf{H}))} = \frac{1}{R_\infty}$$

## Esempio

---

Secondo i diversi valori di  $\rho(\mathbf{H})$ , il numero di iterazioni  $\tilde{k}$  richieste per ridurre l'errore iniziale di un fattore  $10^{-1}$  è dato da:

$$\rho(\mathbf{H}) = 0.1 \quad R_\infty = 1 \quad \tilde{k} \geq 1$$

$$\rho(\mathbf{H}) = 0.5 \quad R_\infty = 0.3 \quad \tilde{k} \geq 3.3$$

$$\rho(\mathbf{H}) = 0.9 \quad R_\infty = 0.046 \quad \tilde{k} \geq 21.7$$

# Convergenza

---

- A diagonale dominante:

$$|a_{i,i}| > \sum_{j=1, j \neq i}^n |a_{i,j}|$$

allora i metodi di Jacobi e Gauss Seidel sono convergenti.

- A matrice simmetrica non singolare con elementi diagonali reali e positivi. Il metodo di Gauss Seidel è convergente se e solo se A è definita positiva.

# Confronto metodi Jacobi e Gauss Seidel

---

Se  $\mathbf{A} \in \mathbb{R}^{n \times n}$  è diagonale dominante in senso stretto allora:

$$\|\mathbf{G}\|_{\infty} \leq \|\mathbf{J}\|_{\infty} < 1$$

Questa relazione non significa che Gauss Seidel sia più veloce di Jacobi. Per avere un confronto sulla velocità di convergenza dei due metodi si deve aggiungere un'ulteriore ipotesi:

● gli elementi di  $\mathbf{J}$  sono positivi:  $\mathbf{J}_{i,j} \geq 0, \forall (i, j)$

allora vale una sola delle seguenti relazioni:

(a)  $\rho(\mathbf{G}) = \rho(\mathbf{J}) = 0$     (c)  $\rho(\mathbf{G}) > \rho(\mathbf{J}) > 1$

(b)  $\rho(\mathbf{G}) < \rho(\mathbf{J}) < 1$     (d)  $\rho(\mathbf{G}) = \rho(\mathbf{J}) = 1$

Molte matrici che si ottengono dalla discretizzazione di operatori differenziali soddisfano tali condizioni. Quindi in questo caso il metodo più conveniente è quello di Gauss–Seidel.



# Matrici tridiagonali

---

Se  $A$  è tridiagonale si riesce a stabilire esattamente quanto Gauss–Seidel sia più veloce di Jacobi. Infatti si ha:

(a)  $\mu$  autovalore di  $J \rightarrow \mu^2$  autovalore di  $G$ .

(b)  $\lambda \neq 0$  autovalore di  $G \rightarrow \sqrt{\lambda}$  autovalore di  $J$ .

Per le matrici tridiagonali il metodo di Gauss Seidel converge se e solo se converge il metodo di Jacobi e vale:

$$\rho(G) = \rho(J)^2$$

Perciò Gauss–Seidel richiede circa la metà delle iterazioni di Jacobi. Infatti confrontando il tasso asintotico di convergenza si ha:

$$R_{\infty}(G) = -\log_{10}(\rho(G)) = -\log_{10}(\rho(J)^2) = 2R_{\infty}(J)$$

# Metodi di Rilassamento

---

Si introducono metodi dipendenti da un parametro per accelerare la convergenza.

Sia  $\mathbf{x}^{(k)}$  il  $k$ -esimo iterato del metodo di Gauss Seidel:

$$\mathbf{x}^{(k)} = \mathbf{D}^{-1}\mathbf{L}\mathbf{x}^{(k)} + \mathbf{D}^{-1}\mathbf{U}\mathbf{x}^{(k-1)} + \mathbf{D}^{-1}\mathbf{b}$$

Si può scrivere:

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + \mathbf{r}^{(k)}$$

dove

$$\mathbf{r}^{(k)} = \mathbf{D}^{-1} \left[ \mathbf{L}\mathbf{x}^{(k)} + \mathbf{U}\mathbf{x}^{(k-1)} + \mathbf{b} \right] - \mathbf{x}^{(k-1)} \quad (3)$$

# Metodo di Rilassamento

---

Si introduce un parametro di rilassamento  $\omega$  al fine di rendere più veloce la convergenza.

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + \omega \mathbf{r}^{(k)}$$

con  $\mathbf{r}^{(k)}$  definito in (??) cioè :

$$\begin{aligned}\mathbf{x}^{(k)} &= \mathbf{x}^{(k-1)} + \omega \left\{ \mathbf{D}^{-1} \mathbf{L} \mathbf{x}^{(k)} + \mathbf{D}^{-1} \mathbf{U} \mathbf{x}^{(k-1)} + \mathbf{D}^{-1} \mathbf{b} - \mathbf{x}^{(k-1)} \right\} = \\ &= (1 - \omega) \mathbf{x}^{(k-1)} + \omega \mathbf{D}^{-1} \mathbf{b} + \omega \mathbf{D}^{-1} \mathbf{L} \mathbf{x}^{(k)} + \omega \mathbf{D}^{-1} \mathbf{U} \mathbf{x}^{(k-1)}\end{aligned}$$

# Metodo SOR

---

Si ha il metodo di rilassamento Successive OverRelaxation method (SOR):

$$\mathbf{x}^{(k)} = \mathbf{H}_\omega \mathbf{x}^{(k-1)} + \omega (\mathbf{D} - \omega \mathbf{L})^{-1} \mathbf{b}$$

con  $\mathbf{H}_\omega = (\mathbf{D} - \omega \mathbf{L})^{-1} ((1 - \omega)\mathbf{D} + \omega \mathbf{U})$ , le singole componenti vengono aggiornate come segue:

$$\mathbf{x}_i^{(k)} = (1 - \omega)\mathbf{x}_i^{(k-1)} + \frac{\omega}{a_{i,i}} \left\{ \mathbf{b}_i - \sum_{j=1}^{i-1} a_{i,j} \mathbf{x}_j^{(k)} - \sum_{j=i+1}^n a_{i,j} \mathbf{x}_j^{(k-1)} \right\}, \quad i = 1, \dots, n \quad (4)$$

Si osserva che per  $\omega = 1$  si ha il metodo di Gauss–Seidel.

# Algoritmo SOR

---

```
function [x,iter,err]=sor(b,a,omega);  
[n,m]=size(a);  
maxit=n^2;tol=eps;  
x=zeros(size(b));z=x;y=x;  
err=norm(b);iter=0;  
continua=(err > tol) & (iter < maxit);  
while continua  
    iter=iter+1;  
    err=0;  
    for i=1:n,  
        sl=0;su=0;  
        if i > 1 sl=a(i,1:i-1)*x(1:i-1); end  
        if i < n su=a(i,i+1:n)*x(i+1:n); end  
        e1=(1-omega)*x(i)+omega*((b(i)-sl-su)/a(i,i));  
        err=err+abs(e1-x(i));  
        x(i)=e1;  
    end;  
    err=sqrt(err);  
    continua=(err > tol) & (iter < maxit);  
end
```

---

## Scelta del parametro di rilassamento (I)

---

Il raggio spettrale della matrice  $\mathbf{H}_\omega$  ha la proprietà :

$$\rho(\mathbf{H}_\omega) \geq |1 - \omega|$$

La matrice  $(\mathbf{D} - \omega\mathbf{L})$  è triangolare inferiore con elementi diagonali dati da  $\mathbf{D}$ , quindi:

$$\det(\mathbf{D} - \omega\mathbf{L}) = \det(\mathbf{D}) \neq 0$$

Analogamente

$$\det((1 - \omega)\mathbf{D} + \omega\mathbf{U}) = (1 - \omega)^n \det(\mathbf{D})$$

quindi  $\det(\mathbf{H}_\omega) = (1 - \omega)^n$ .

## Scelta del parametro di rilassamento (II)

---

$$\det(\mathbf{H}_\omega) = \prod_{i=1}^n \lambda_i \leq \rho(\mathbf{H}_\omega)^n$$

ne segue che:

$$\rho(\mathbf{H}_\omega) \geq |(1 - \omega)|$$

- La relazione fornisce una **condizione necessaria** ma non sufficiente per la convergenza del metodo di rilassamento, cioè :

$$\rho(\mathbf{H}_\omega) < 1 \Rightarrow |(1 - \omega)| < 1 \Rightarrow 0 < \omega < 2$$

- La **condizione** diventa anche **sufficiente** se  $\mathbf{A}$  è definita positiva.
- Se  $\mathbf{A}$  è hermitiana e definita positiva e  $\omega \in (0, 2)$ , allora il metodo SOR è convergente.

## Scelta del parametro di rilassamento (III)

Nel seguente caso particolare è possibile stabilire una relazione fra gli autovalori di  $\mathbf{H}_\omega$  e quelli del metodo di jacobi ed indicare il valore di  $\omega$  che rende massima la velocità di convergenza.

Sia  $\mathbf{A}$  tridiagoale:

$$\mathbf{A} = \begin{pmatrix} a_1 & c_1 & & & \\ b_1 & a_2 & c_2 & & \\ 0 & \ddots & \ddots & \ddots & \\ & & & b_{n-2} & a_{n-1} & c_{n-1} \\ & & & & b_{n-1} & a_n \end{pmatrix} \quad (5)$$

e  $0 < \omega < 2$  allora valgono le seguenti relazioni:



## Scelta del parametro di rilassamento (IV)

---

(a) se  $\mu$  è autovalore di  $\mathbf{J}$ , ogni  $\lambda$  tale che

$$(\lambda + \omega - 1)^2 = \lambda \omega^2 \mu^2 \quad (6)$$

è autovalore di  $\mathbf{H}_\omega$ ;

- (b) se  $\lambda$  è autovalore di  $\mathbf{H}_\omega$  e  $\lambda \neq 0$  allora  $\mu$  che soddisfa (??) è autovalore di  $\mathbf{J}$ ;
- (c) se gli autovalori della matrice di iterazione  $\mathbf{J}$  sono reali e tali che  $\rho(\mathbf{J}) < 1$ , esiste un solo valore  $\omega_0$  per cui vale:

$$\rho(\mathbf{H}_{\omega_0}) = \min_{0 < \omega < 2} \rho(\mathbf{H}_\omega)$$

ed è

$$\omega_0 = \frac{2}{1 + \sqrt{1 - \rho^2(\mathbf{J})}}$$

## Esempio

---

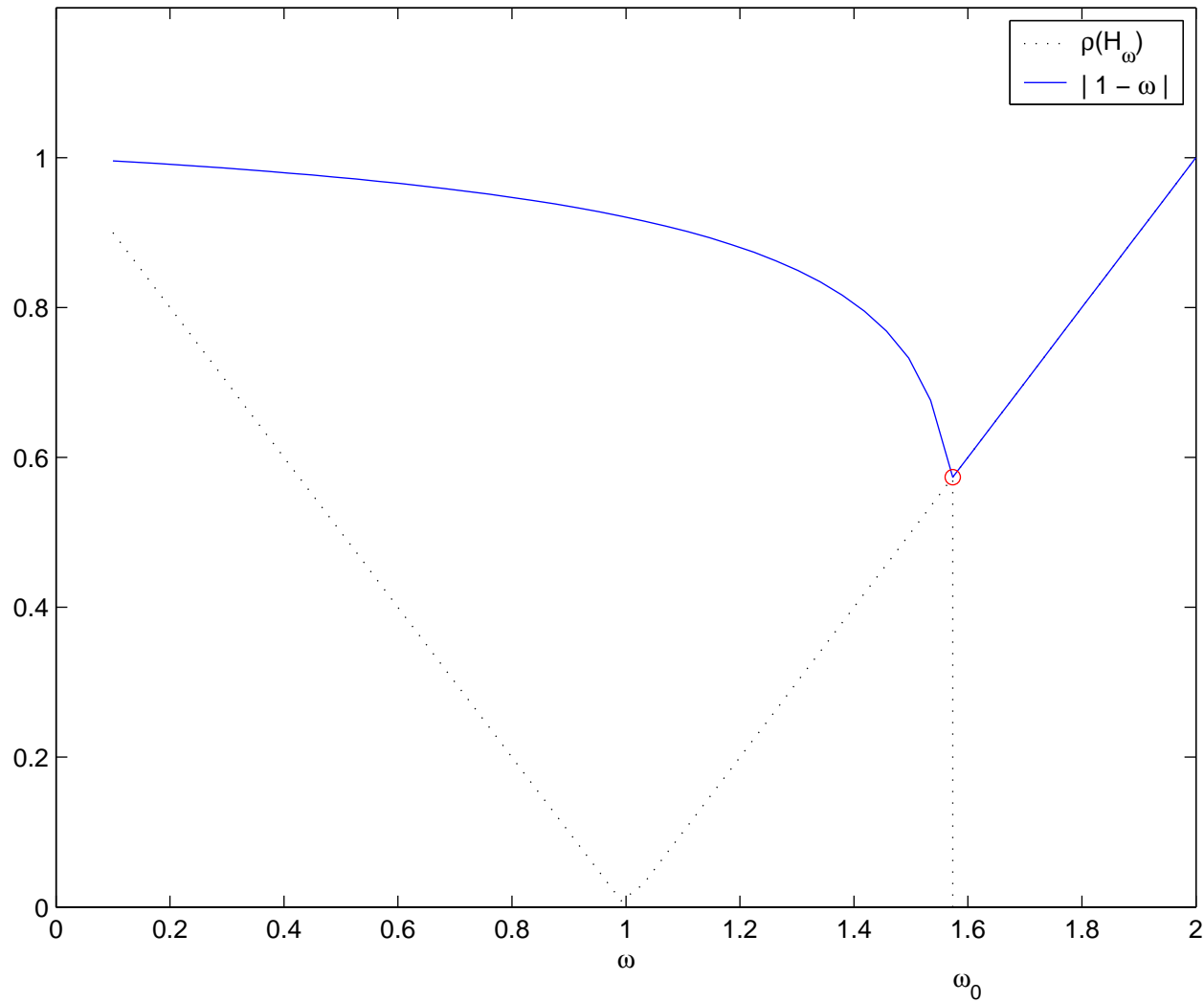
Nel caso della matrice tridiagonale :

$$A = \begin{cases} 2 & i = j \\ -1 & |i - j| = 1 \\ 0 & \text{altrimenti} \end{cases}$$

il comportamento di  $\rho(\mathbf{H}_\omega)$  al variare di  $\omega$  è riportato nel grafico.

- Si osserva che  $1 < \omega_0 < 2$ .
- Una stima per eccesso del parametro ottimale è da preferirsi in quanto la crescita di  $\rho(\mathbf{H}_\omega)$  è lineare.

# Raggio Spettrale della matrice di iterazione



# Metodi per blocchi

La matrice  $\mathbf{A}$  viene suddivisa in blocchi  $A_{i,j} \in \mathbb{R}^{m \times m}$  come segue:

$$\mathbf{A} = \begin{pmatrix} A_{1,1} & A_{1,2} & A_{1,3} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & A_{2,3} & \dots & A_{2,n} \\ A_{3,1} & A_{3,2} & A_{3,3} & \dots & A_{1,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n,1} & A_{n,2} & A_{n,3} & \dots & A_{n,n} \end{pmatrix}$$

Siano:

$$\mathbf{b} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \\ \vdots \\ \mathbf{b}_n \end{pmatrix} \quad \mathbf{x}_B^{(k)} = \begin{pmatrix} \mathbf{x}_1^{(k)} \\ \mathbf{x}_1^{(k)} \\ \mathbf{x}_1^{(k)} \\ \vdots \\ \mathbf{x}_1^{(k)} \end{pmatrix}$$

# Metodi per blocchi

Si definisce la partizione:  $\mathbf{A} = \mathbf{D}_B - \mathbf{L}_B - \mathbf{U}_B$  dove  
 $D_B = \text{diag}(A_{i,i}), \quad i = 1, \dots, n$

$$\mathbf{L}_B = \begin{pmatrix} \mathbf{O} & \mathbf{O} & \mathbf{O} & \dots & \mathbf{O} \\ -A_{2,1} & \mathbf{O} & \mathbf{O} & \dots & \mathbf{O} \\ -A_{3,1} & -A_{3,2} & \mathbf{O} & \dots & \mathbf{O} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -A_{n,1} & -A_{n,2} & -A_{n,3} & \dots & \mathbf{O} \end{pmatrix}$$

$$\mathbf{U}_B = \begin{pmatrix} \mathbf{O} & -A_{1,2} & -A_{1,3} & \dots & -A_{1,n} \\ \mathbf{O} & \mathbf{O} & -A_{2,3} & \dots & -A_{2,n} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \dots & -A_{1,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \dots & \mathbf{O} \end{pmatrix}$$

# Jacobi e Gauss Seidel per Blocchi

---

Si definiscono le matrici di iterazione del metodo di Jacobi e Gauss Seidel a Blocchi:

$$\mathbf{J}_B = \mathbf{D}_B^{-1}(\mathbf{L}_B + \mathbf{U}_B) \quad \mathbf{G}_B = (\mathbf{D}_B - \mathbf{L}_B)\mathbf{U}_B$$

Il metodo di Jacobi a blocchi è dato da:

$$\mathbf{A}_{i,i}\mathbf{x}_i^{(k)} = \mathbf{b}_i - \sum_{j=1}^{i-1} \mathbf{A}_{i,j}\mathbf{x}_i^{(k-1)} - \sum_{j=i+1}^n \mathbf{A}_{i,j}\mathbf{x}_i^{(k-1)}$$

Il metodo di Gauss Seidel a blocchi è dato da:

$$\mathbf{A}_{i,i}\mathbf{x}_i^{(k)} = \mathbf{b}_i - \sum_{j=1}^{i-1} \mathbf{A}_{i,j}\mathbf{x}_i^{(k)} - \sum_{j=i+1}^n \mathbf{A}_{i,j}\mathbf{x}_i^{(k-1)}$$

# Convergenza

---

- L'estensione dei criteri di convergenza non è immediata: non è detto, in generale, che se il metodo scalare converge allora converge quello a blocchi o viceversa.
- Sia  $\mathbf{A} \in \mathbb{R}^{n \times n}$  una matrice a blocchi  $A_{i,j} \in \mathbb{R}^{m \times m}$ , tale che i blocchi diagonali sono non singolari. Se  $a_{i,j} \leq 0, \forall i \neq j$ , e gli elementi della matrice  $\mathbf{A}^{-1}$  sono tutti positivi, allora i metodi di Jacobi e Gauss Seidel applicati scalarmente e a blocchi sono convergenti e vale:

$$\begin{aligned}\rho(\mathbf{G}_B) &< \rho(\mathbf{J}_B) < 1 \\ \rho(\mathbf{G}_B) &\leq \rho(\mathbf{G}) < 1 \\ \rho(\mathbf{J}_B) &\leq \rho(\mathbf{J}) < 1\end{aligned}\tag{7}$$

il segno di uguaglianza vale solo se  $\mathbf{G}_B = \mathbf{G}$  o  $\mathbf{J}_B = \mathbf{J}$  rispettivamente.

# Convergenza

---

- Fra le matrici che verificano le condizioni (??) ci sono alcune fra quelle che si ottengono dalla discretizzazione di problemi differenziali di tipo ellittici.
- Per tali matrici risulta quindi più conveniente l'uso del metodo di Gauss Seidel a blocchi.
- Per le matrici tridiagonali a blocchi, vale inoltre un risultato analogo al teorema ??, e quindi in tal caso il metodo di Gauss Seidel a blocchi è convergente se e solo se lo è Jacobi a blocchi e:

$$\rho(\mathbf{G}_B) = \rho^2(\mathbf{J}_B)$$