

Metodo Iterativo per la Risoluzione di un Sistema Lineare : Jacobi

Risoluzione di un Sistema di Equazioni Lineare $Ax = b$ con A sparsa.

Sintassi

- $x = \text{Jacobi}(A,b)$
- $x = \text{Jacobi}(A,b,TOL)$
- $x = \text{Jacobi}(A,b,TOL,MAXITER)$
- $[x,niter] = \text{Jacobi}(A,b)$
- $[x,niter] = \text{Jacobi}(A,b,TOL)$
- $[x,niter] = \text{Jacobi}(A,b,TOL,MAXITER)$
- $[x,niter,resrel] = \text{Jacobi}(A,b)$
- $[x,niter,resrel] = \text{Jacobi}(A,b,TOL)$
- $[x,niter,resrel] = \text{Jacobi}(A,b,TOL,MAXITER)$

Descrizione

1. $x = \text{Jacobi}(A,b)$ risolve il sistema di equazioni lineari $Ax = b$. A deve essere una matrice quadrata sparsa, b deve essere un vettore colonna avente lo stesso numero di righe di A . La soluzione è corretta a meno di un errore dovuto al malcondizionamento della matrice A .
2. $x = \text{Jacobi}(A,b,TOL)$ usa TOL per determinare l'accuratezza della soluzione. Se non specificato $TOL = 10^{-6}$
3. $x = \text{Jacobi}(A,b,TOL,MAXITER)$ usa TOL per determinare l'accuratezza della soluzione e $MAXITER$ per individuare il numero massimo di iterazioni che l'algoritmo può compiere. Se non specificati, $TOL = 10^{-6}$ e $MAXITER=500$.

Esempi di Utilizzo

Sistema Lineare con A Matrice Sparsa Casuale

- Command Line Matlab

```
A = sprand(4);  
b = rand(length(A),1);  
x = Jacobi(A,b)
```

Warning: Attenzione: TOL e MAXITER non specificati, uso TOL=10^-6 e MAXITER=500 valori di default
Specificato solo soluzione x come uscita

```
x =  
(1,1)      36.7416
```

Sistema Lineare con A Matrice Sparsa di Poisson

- **Command Line Matlab**

```
A = gallery('poisson',10);  
b = rand(length(A),1);  
x = Jacobi(A,b)
```

Warning: Attenzione: TOL e MAXITER non specificati, uso TOL=10⁻⁶ e MAXITER=500 valori di default
Specificato solo soluzione x come uscita

```
x = 100x1  
    0.6200  
    1.1301  
    1.1987  
    1.3175  
    1.5674  
    1.5243  
    1.4176  
    1.2946  
    1.0473  
    0.7137  
      ⋮  
      ⋮
```

Sistema Lineare con A Matrice Sparsa, specificazione *TOL* e *NMAX* come parametri di ingresso e visualizzazione degli ulteriori parametri di uscita: *niter* e *resrel*

- **Command Line Matlab**

```
A = sprand(4);  
b = rand(length(A),1);  
TOL = 10-14;  
MAXITER = 900;  
[x,niter,resrel] = Jacobi(A,b,TOL,MAXITER)
```

Specificati tutti i parametri come uscita

```
x =  
    (1,1)      0.4064  
niter = 2  
resrel = 1.4079e-16
```

Esempio da Command Line Matlab

```
[x,niter,resrel] = Jacobi(sprand(4),rand(length(A),1),eps,700)
```

Specificati tutti i parametri come uscita

```
x =  
    (1,1)      0.8911  
niter = 2  
resrel = 1.5422e-16
```

Esecuzione da Interfaccia Grafica

In aggiunta all'esecuzione testuale dell'algoritmo, si può utilizzare un interfaccia grafica che esegue l'algoritmo, appositamente implementata. Il comando per l'esecuzione della stessa è **MainGUI()**

Riferimenti

Docenti.unina.it D'alessio Alessandra

[Wikipedia](#) : Metodo di Jacobi

Autori

Giuseppe Napolano M63000856 Raffaele Formisano M63000912 Giuseppe Romito M63000936