

Metodi Iterativi per la Risoluzione di un Sistema Lineare : Jacobi

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

Sintassi

- `x = Jacobi(A,b)`
- `x = Jacobi(A,b,TOL)`
- `x = Jacobi(A,b,TOL,MAXITER)`
- `[x,niter] = Jacobi(A,b)`
- `[x,niter] = Jacobi(A,b,TOL)`
- `[x,niter] = Jacobi(A,b,TOL,MAXITER)`
- `[x,niter,resrel] = Jacobi(A,b)`
- `[x,niter,resrel] = Jacobi(A,b,TOL)`
- `[x,niter,resrel] = Jacobi(A,b,TOL,MAXITER)`

Descrizione

1. `x = 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 = Jacobi(A,b,TOL)` usa **TOL** per determinare l'accuratezza della soluzione. Se non specificato **TOL** = 10^{-6} .
3. `x = 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.
4. `[x,niter] = Jacobi(A,b,TOL,MAXITER)` ritorna oltre alla soluzione il numero di iterazioni che sono state necessarie per determinarla.
5. `[x,niter,resrel] = Jacobi(A,b,TOL,MAXITER)` viene restituito anche il residuo relativo.

Parametri di input

A - Matrice dei coefficienti

Matrice dei coefficienti quadrata sparsa.

Data Types: single | double

b - Vettore dei termini noti

Vettore dei termini noti, specificato come un vettore colonna avente lo stesso numero di righe della matrice A.

Data Types: single | double

TOL - Tolleranza

Facoltativo, valore di accuratezza desiderato, specificato come uno scalare. Se omissso TOL=10⁻⁶.

Data Types: single | double

MAXITER - Numero massimo di iterazioni

Facoltativo, numero massimo di iterazioni che l'agoritmo deve eseguire, specificato come uno scalare. Se omissso MAXITER=500.

Data Types: single | double

Parametri di output

-x :Soluzione del sistema

Soluzione del sistema, restituita come un vettore colonna.

-NITER

Numero di iterazioni effettuate dall'agoritmo per determinare la soluzione

-RESREL

Residuo relativo : $\frac{\|b - Ax\|}{\|b\|}$

Esempi di Utilizzo

Sistema Lineare con A Matrice Sparsa

```
A=sparse([1 3 2 3 4 1], [1 2 2 3 4 3],[1 2 3 4 5 8],4,4);  
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

```
x =  
-0.9078  
0.0109  
0.1349  
0.1764
```

Sistema Lineare con A Matrice Sparsa di Poisson

```
A = gallery('poisson',3);  
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

```
x =  
0.3902  
0.3879  
0.2795  
0.5038  
0.6915  
0.3612
```

```
0.4727
0.5316
0.3173
```

Sistema Lineare con A Matrice Sparsa, avente *TOL* e *NMAX* come parametri di ingresso e *niter* e *resrel* come parametri di uscita

```
A=sparse([2 0 0 1; 4 -1 -1 0; 0 0 1 0; -2 0 0 2]);
b = [8; -1; -18; 8];
TOL = 10^-5;
MAXITER = 900;
[x,niter,resrel] = Jacobi(A,b,TOL,MAXITER)
```

```
x =
    1.3333
   24.3334
  -18.0000
    5.3333
niter = 35
resrel = 6.5707e-06
```

Sistema Lineare con A Matrice Sparsa avente *TOL=eps* e *NMAX=700*

```
A=sparse([2 0 0 1; 4 -1 -1 0; 0 0 1 0; -2 0 0 2]);
b = [8; -1; -18; 8];
[x,niter,resrel] = Jacobi(A,b,eps,700)
```

```
x =
    1.3333
   24.3333
  -18.0000
    5.3333
niter = 105
resrel = 1.8662e-16
```

Esecuzione da Interfaccia Grafica

Si può inoltre eseguire la funzione Jacobi attraverso l'interfaccia grafica appositamente implementata. Il comando per utilizzare l'interfaccia è `run MainGui()`.

Riferimenti

[Docenti.unina.it D'alessio Alessandra](http://Docenti.unina.it/D'alessio/Alessandra)

[Wikipedia : Metodo di Jacobi](#)

Autori

Giuseppe Napolano M63000856 Raffaele Formisano M63000912 Giuseppe Romito M63000936