

Metodi del Calcolo Scientifico

Risoluzione di sistemi lineari tramite il metodo di Cholesky

SILVA EDOARDO 816560

ZHIGUI BRYAN 816335

MARCHETTI DAVIDE 815990

A.A. 2019/2020

Abstract

Lo scopo di questo progetto è di studiare l'implementazione del metodo di Choleski per la risoluzione sistemi lineari per matrici sparse, simmetriche e definite positive in ambienti di programmazione open source e di compararli con l'implementazione di MATLAB.

Il confronto avverrà in termini di tempo, accuratezza, impiego della memoria e anche facilità d'uso sia in ambiente Linux che Windows, eseguendo il codice su diverse matrici sparse derivate da problemi reali e raccolte nella **SuiteSparse Matrix Collection**.

1 Analisi dell'implementazione

1.1 MATLAB

Per la decomposizione di cholesky, MATLAB mette a disposizione il modulo **cholesky.matlab** contenente tutto il necessario. In particolare è stata utilizzata la funzione **chol**

Utilizzo

`R = chol(A, [triangle])`: Fattorizza la matrice A simmetrica definita positiva in una matrice triangolare superiore R tale che $A = R^{-1}R$. Il pa-

rametro **triangle** permette di scegliere se attuare la decomposizione in una matrice triangolare superiore (opzione di default) o traiangolare inferiore. In quest'ultimo caso, la matrice R risultante dall'equazione soddisferà l'uguaglianza $A = RR^{-1}$.

Manutenzione

La libreria è stata rilasciata per la prima volta nell'aggiornamento R2013a MATLAB. Attualmente, è ancora supportata e non presenta lacune o problemi che sono stati riscontrati durante il suo l'utilizzo.

Licenza

Essendo MATLAB un software closed-source, non è possibile accedere al codice sorgente del modulo.

1.2 Open-Source (C++)

Dopo un'attenta analisi e comparazione di diverse opzioni, l'implementazione in C++ è stata costruita utilizzando **Eigen**, libreria che si pone l'obiettivo di essere leggera ed offrire supporto alle operazioni su vettori e matrici dense e sparse.

Utilizzo

Eigen::loadMarket(A, filename): Importa i valori di una matrice sparsa memorizzata in un file **.mtx** nella matrice fornita come primo argomento. Nel nostro programma, **A** è definita come **Eigen::SparseMatrix<Type>**.

Il modulo **unsupported/Eigen/SparseExtra** che contiene queste funzionalità è attualmente deprecato.

- **Eigen::VectorXd::Ones(A.rows()):** dichiara matrice di dimensioni fissate (prese dalle dimensioni della matrice **A**), package 'VectorXd' usato per le operazioni su matrici dinamiche di double.
- **Eigen::SimplicialCholesky<SpMat> chol(A):** Pacchetto creato per gestire matrici di grandi dimensioni con pochi elementi diversi da 0. Implementa uno schema di rappresentazione e gestione dei valori

diversi da 0 con uso di poca memoria e alte prestazioni.

Il metodo `chol(A)` implementa la fattorizzazione di Cholesky della matrice `A`.

- **Eigen::VectorXd `x_ap` = `chol.solve(b)`:** Applicazione del risolutore iterativo per risolvere la fattorizzazione.

1.2.1 Manutenzione

Eigen è in sviluppo attivo, tuttavia, alcuni moduli sono marcati come deprecati e non ne è garantito il loro pieno funzionamento. Un esempio di questi è il modulo `MarketIO`, che permette di effettuare operazioni di Input e Output con file in formato Matrix Market (`.mtx`).

1.2.2 Problemi

Durante lo sviluppo, l'utilizzo di una classe deprecata ha inizialmente rallentato lo sviluppo. Infatti, delle matrici importate tramite `MarketIO` veniva ignorato il fatto che fossero salvate come simmetriche o meno.

La soluzione a questo problema è stata messa in pratica modificando lo script matlab `mmwrite` di conversione per file `.mat` in `.mtx` e rigenerando le matrici a partire dai file `.mat`, assicurandosi che venissero salvati correttamente tutti gli elementi della matrice.

1.2.3 Licenza

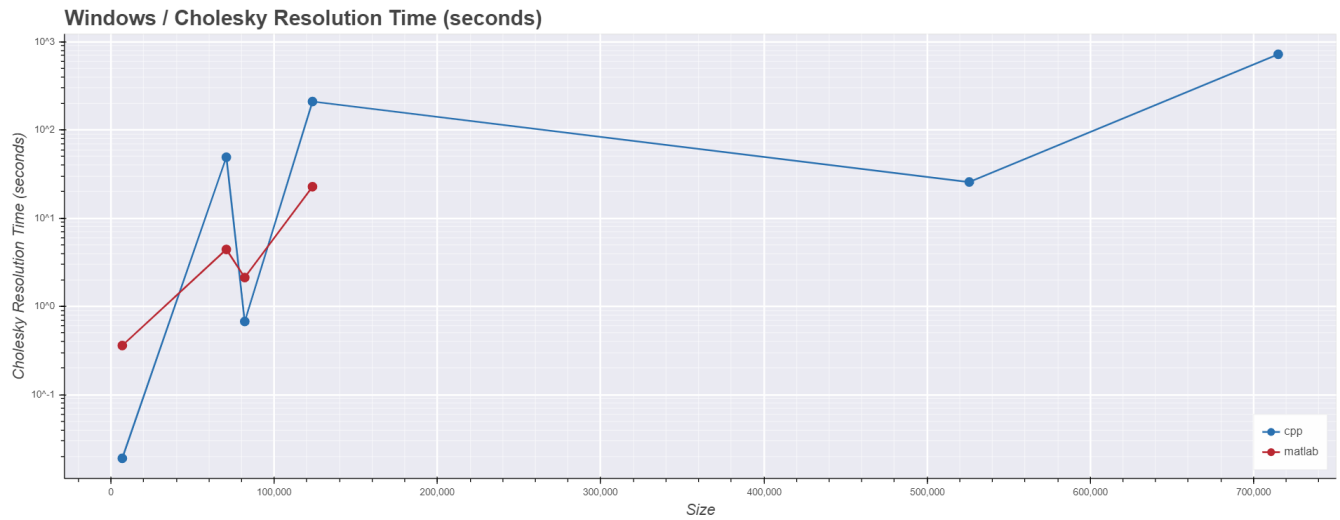
Eigen è un software gratuito ed open-source rilasciato con licenza Mozilla Public License 2.0 (MPL2: simple weak copyleft license) dalla versione 3.1.1.

2 Specifiche hardware

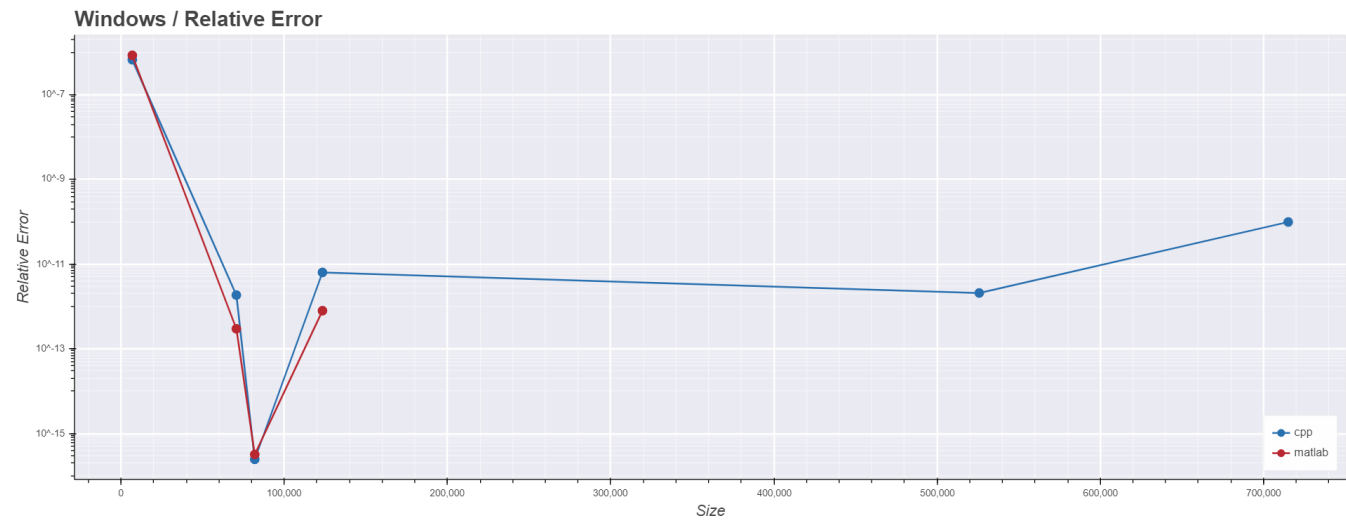
3 Risultati

3.1 Windows

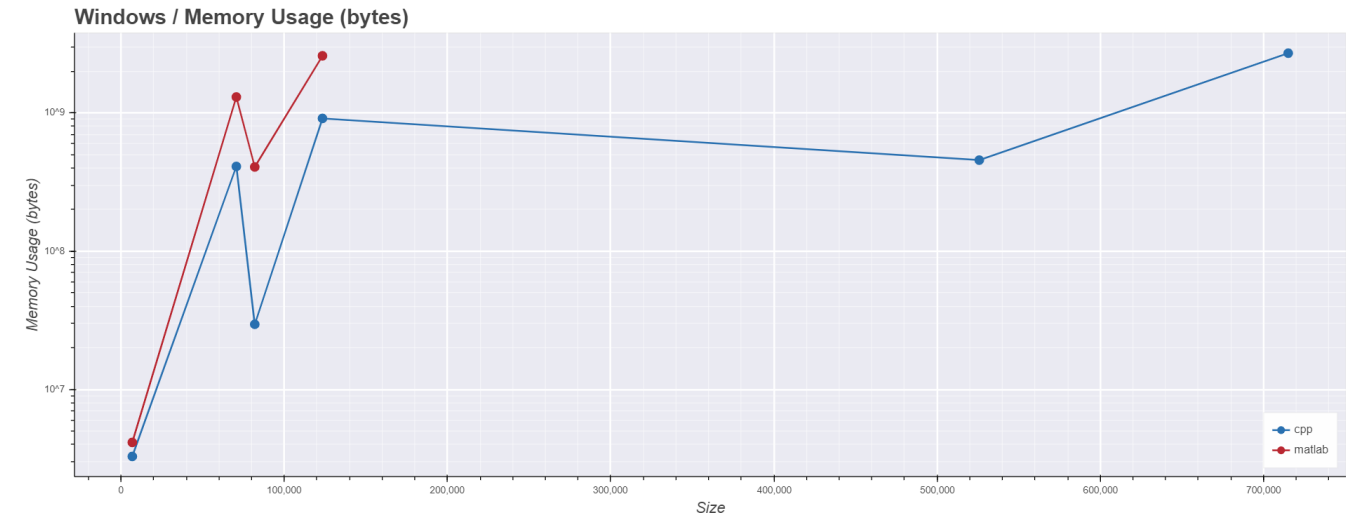
Tempo



Errore relativo

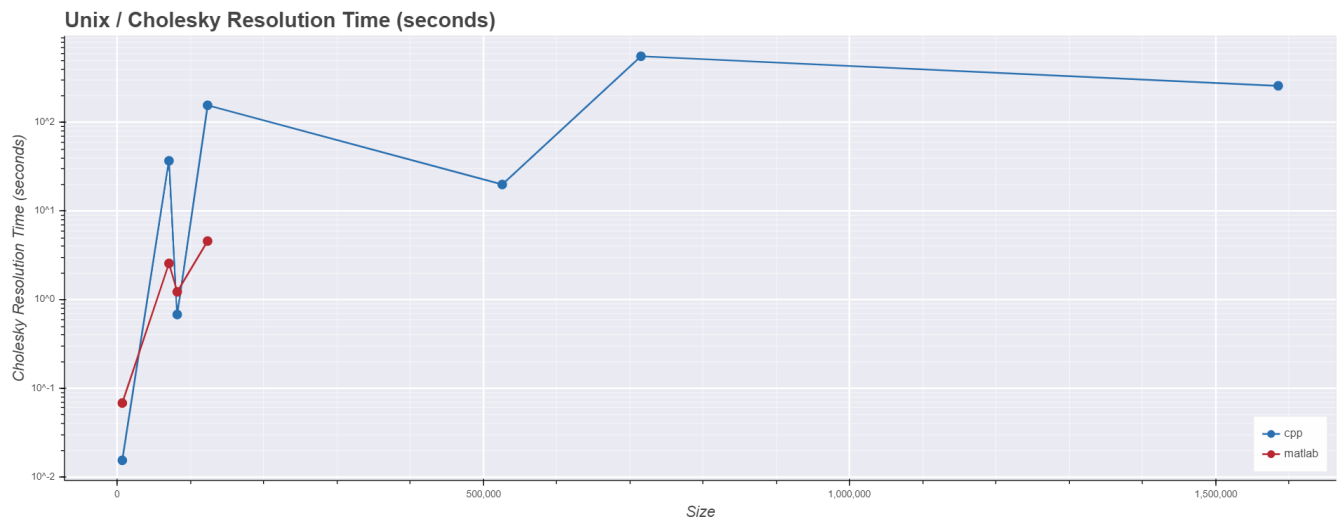


Memoria

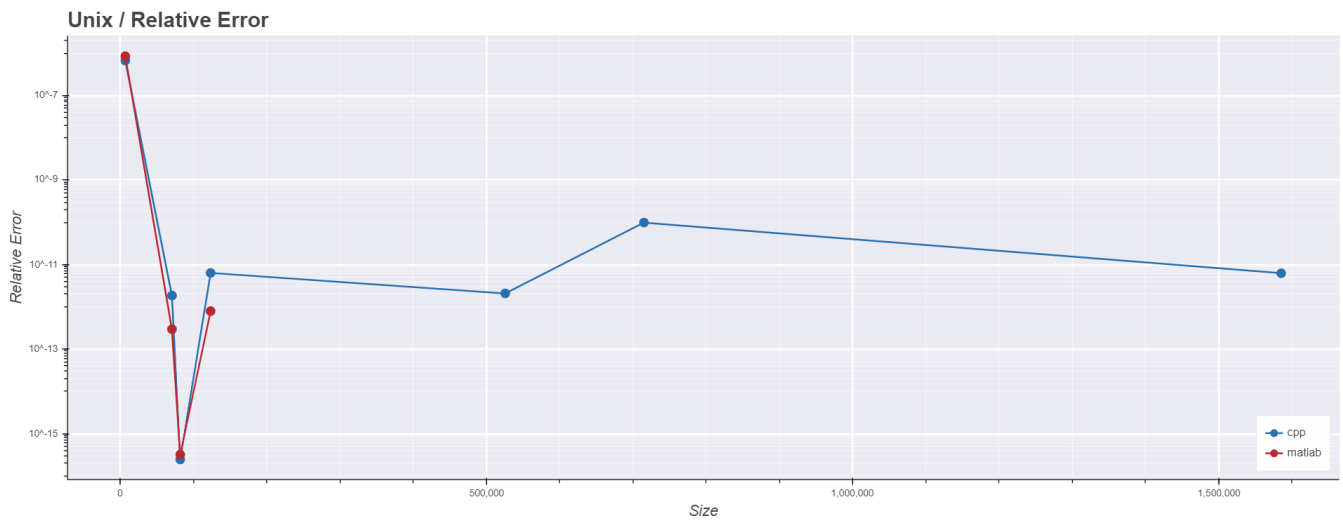


3.2 Linux

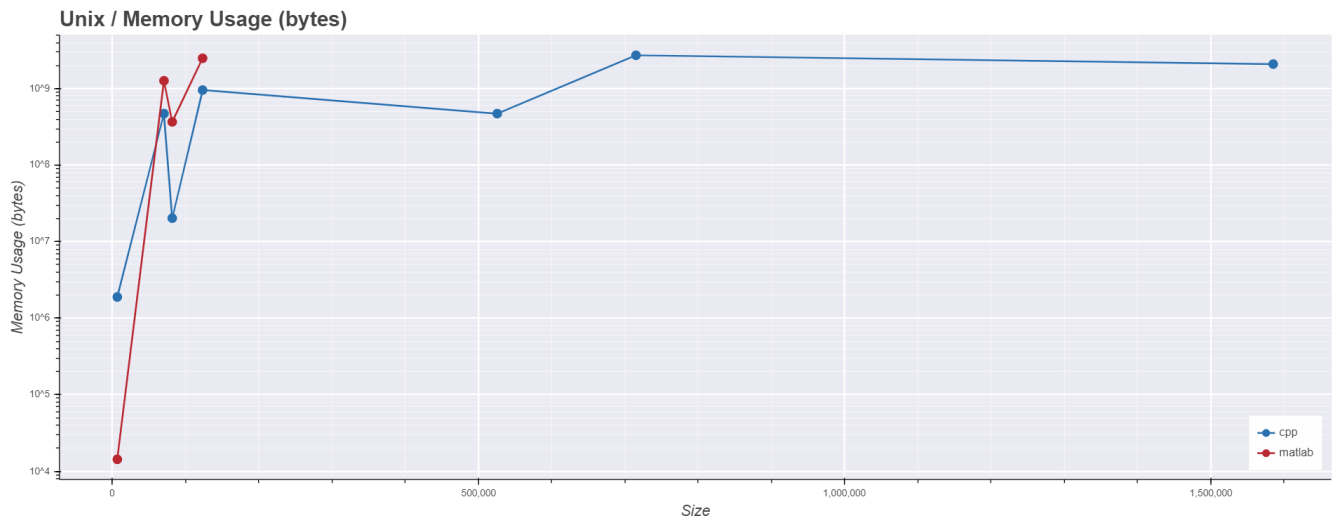
Tempo



Errore relativo



Memoria



4 Conclusioni

5 Code