

Cholesky Decomposition

Structuri Multiprocesor

Indrumator: Conf.univ.dr.ing. Bogdan Țigănoaia

Student: Margineanu Nicolae-Vladut

Seria si grupa: 341C2

Factorizarea Cholesky este un tip de factorizare ce consta in obtinerea unei matrici inferior triunghiulare care inmultita cu simetrica sa, va rezulta in matricea originala.

Pentru ca o matrice sa poata fi supusa acestei descompuneri ea trebuie sa fie Hermitica si pozitiv definita.

Cand A este o matrice de numere reale (hence symmetric positive-definite), factorizarea mai poate fi scrisa si astfel:

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T,$$

$$\begin{aligned} A &= \begin{pmatrix} a_{11} & a_{21} & a_{31} \\ a_{21} & a_{22} & a_{32} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \\ &= \begin{pmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{pmatrix} \begin{pmatrix} l_{11} & l_{21} & l_{31} \\ 0 & l_{22} & l_{32} \\ 0 & 0 & l_{33} \end{pmatrix} \equiv LL^T \\ &= \begin{pmatrix} l_{11}^2 & l_{21}l_{11} & l_{31}l_{11} \\ l_{21}l_{11} & l_{21}^2 + l_{22}^2 & l_{31}l_{21} + l_{32}l_{22} \\ l_{31}l_{11} & l_{31}l_{21} + l_{32}l_{22} & l_{31}^2 + l_{32}^2 + l_{33}^2 \end{pmatrix} \end{aligned}$$

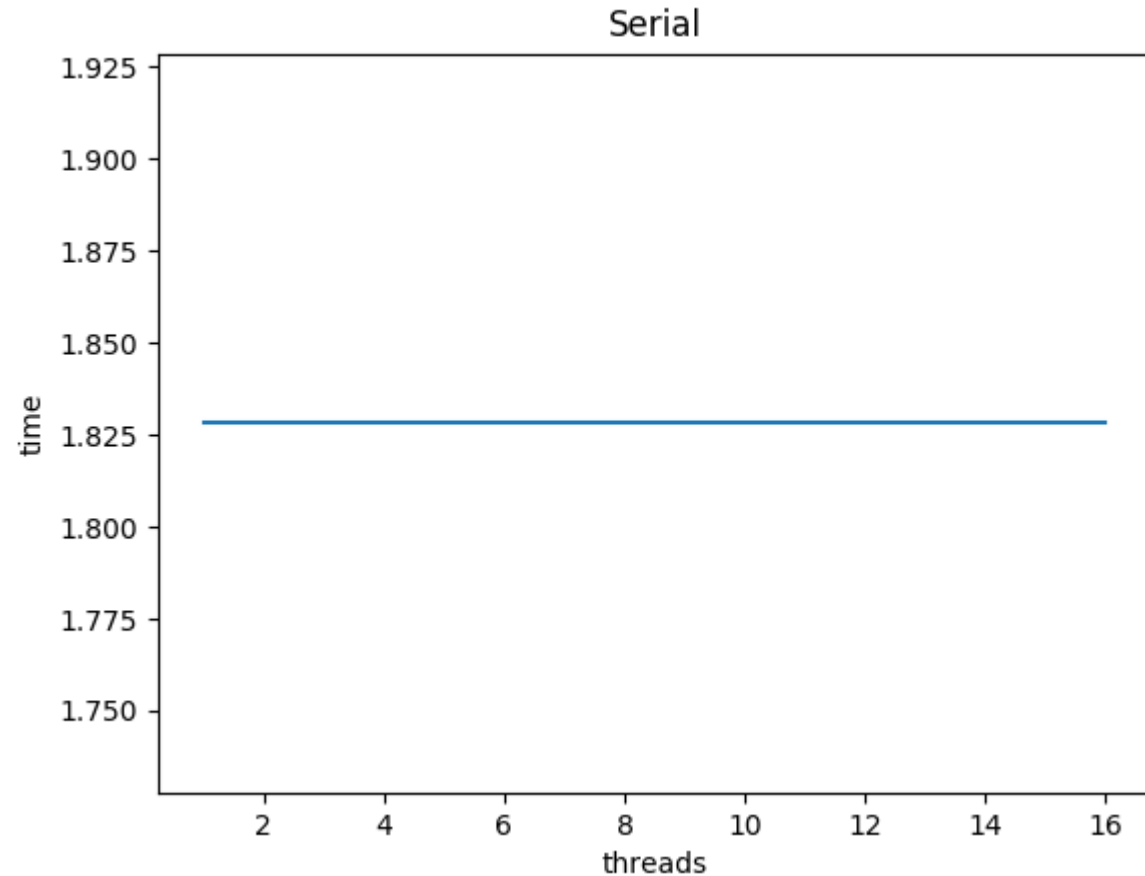
$$l_{kk} = \sqrt{a_{kk} - \sum_{j=1}^{k-1} l_{kj}^2}$$

$$l_{ik} = \frac{1}{l_{kk}} \left(a_{ik} - \sum_{j=1}^{k-1} l_{ij} l_{kj} \right)$$

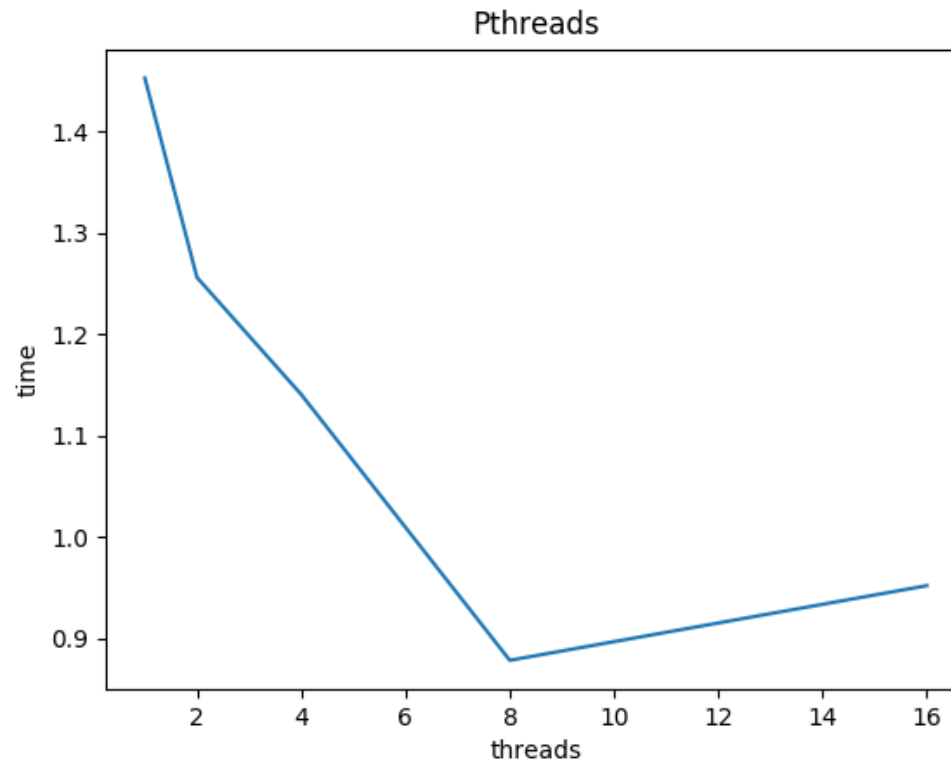
-
- Pentru calcularea elementelor matricei inferior triunghiulare se pot folosi urmatoarele formule:

Serial

- Se poate observa din grafic, urmatoarele:
- Implementarea seriala este constanta in timp
- Implementarea seriala este pe locul trei la viteza de executie fata de celelalte metode folosite



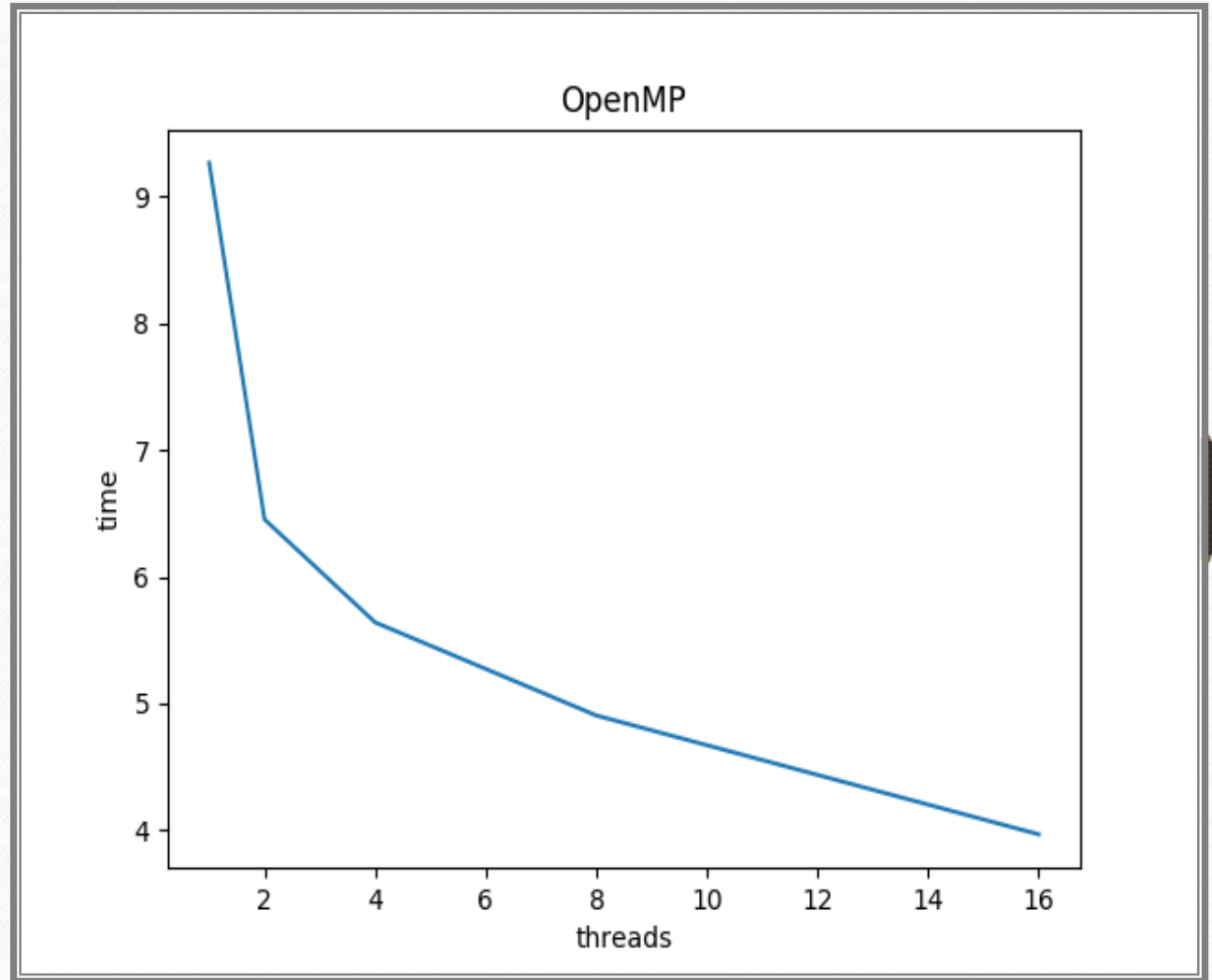
Pthreads



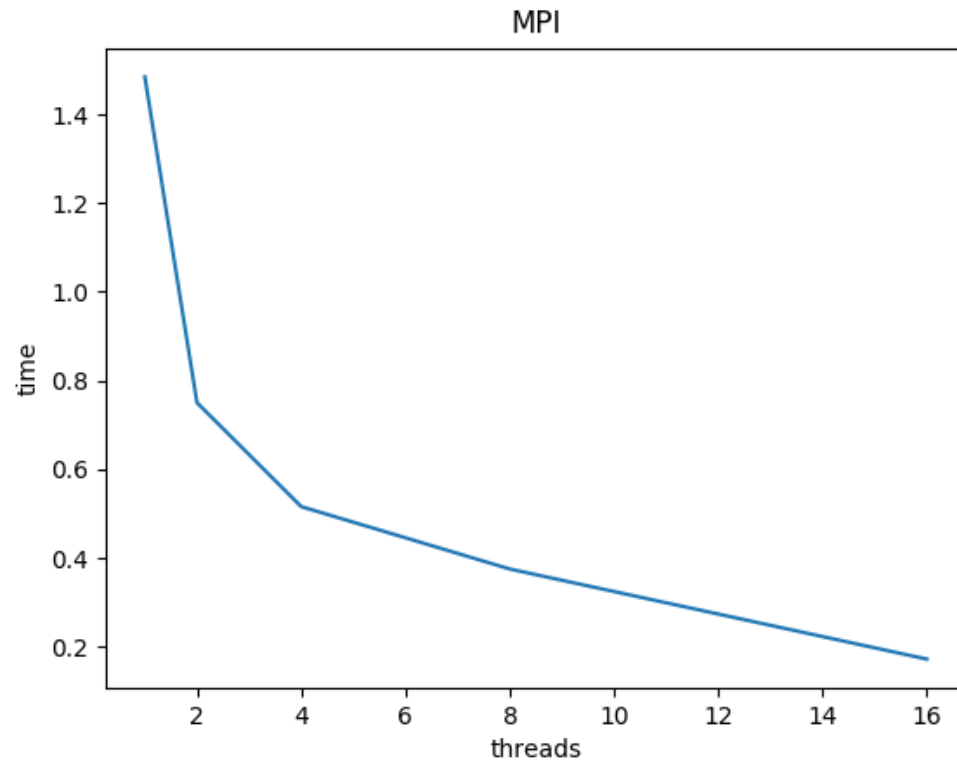
- Se poate observa din grafic, urmatoarele:
- Numarul de thread-uri folosite in implementare influenteaza viteza de executie a programului
- Se poate observa ca atunci cand se folosesc 8 thread-uri, este cea mai rapida executie
- Metoda Pthreads este pe locul doi la viteza de executie fata de celelalte metode implementate

OpenMP

- Se poate observa din grafic, urmatoarele:
 - Numarul de thread-uri folosite in implementare influenteaza viteza de executie a programului
 - Viteza de executiei a programului creste direct proportional cu numarul de thread-uri folosite
 - Metoda OpenMP este implementarea pe locul patru in viteza de executie fata de celelalte metode folosite



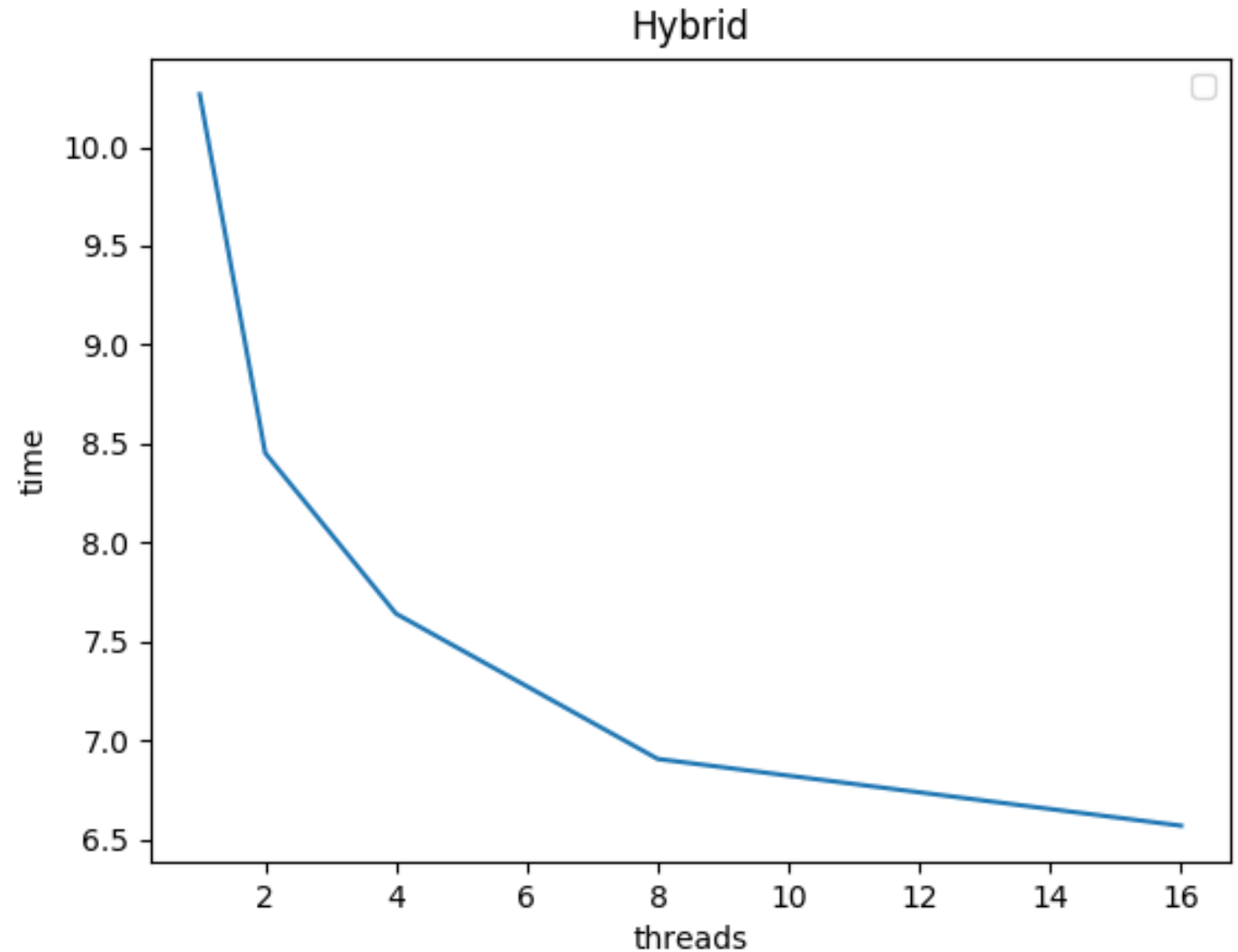
MPI



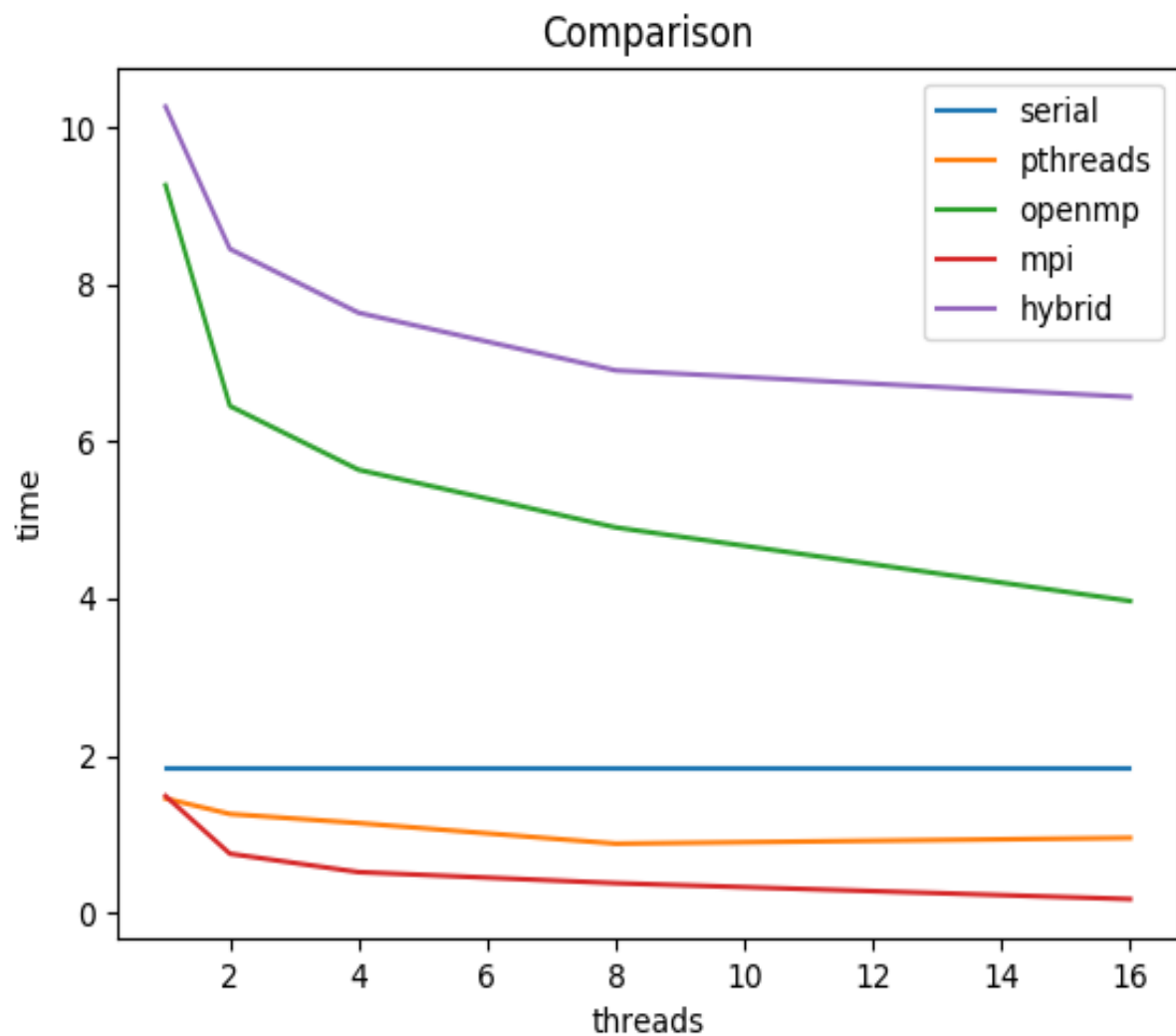
- Se poate observa din grafic, urmatoarele:
- Numarul de thread-uri folosite in implementare influenteaza viteza de executie a programului
- Viteza de executiei a programului creste direct proportional cu numarul de thread-uri folosite
- Metoda MPI este cea mai rapida implementare dintre cele cinci metode folosite in functie de timp

Hybrid : MPI - OpenMP

- Se poate observa din grafic, urmatoarele:
- Numarul de thread-uri folosite in implementare influenteaza viteza de executie a programului
- Viteza de executiei a programului creste direct proportional cu numarul de thread-uri folosite
- Medoda hibrida MPI-OpenMP este cea mai lenta metoda din cele cinci in functie de timp



Comparatie



- In urma implementarii celor cinci metode, se poate observa evolutia celor cinci executii de programe in antiteza, in functie de timp.

Concluzii

1. Pe baza rezultatelor, MPI este cea mai rapida varianta pentru descompunerea Cholesky, urmata de Pthreads.
2. Varianta Hybrid : MPI – OpenMP este cea mai lenta dintre cele 5 variante analizate.
3. De asemenea, se observa ca atat varianta OpenMP, cat si varianta Hybrid MPI – OpenMP sunt mai lente decat implementarea seriala.

Bibliografie:

- https://en.wikipedia.org/wiki/Cholesky_decomposition
- <https://www.youtube.com/watch?v=NppyUqgQqd0>
- <https://www.sciencedirect.com/topics/engineering/cholesky-decomposition>

Va multumesc pentru
atentie!