

Parallel and Distributed Computing

Prof. G. Laccetti – a.a. 2023-2024

Università degli Studi di Napoli Federico II

PROGRAMMA 2023/24

1. Concetti introduttivi: Motivazioni. Tipi di parallelismo: temporale, spaziale e asincrono. Parallelismo on chip. Classificazione di Flynn. Calcolatori MIMD a memoria condivisa e a memoria distribuita. Reti di interconnessione. Problematiche di base del calcolo parallelo e del calcolo distribuito.

[1] Cap. 1,2 e 6; [1bis] cap 1; [4] (vedi Bibliografia)

2. Il paradigma dello scambio di messaggi (message passing): Modello message-passing: caratteristiche e funzioni di base. Lo standard MPI (Message Passing Interface). Principali routines per la gestione dei processi e loro comunicazione. Griglia di processori.

[1] Cap. 3; [1bis] Cap. 2, Par. 4.5, 6.2; [2] Cap. 1, 3, 3, 6, 7, 8 e 11; [4] (vedi Bibliografia)

3. Parametri di valutazione di un software parallelo: Efficienza e speed-up di un algoritmo parallelo. Legge di Ware-Amdahl e legge di Ware-Amdahl generalizzata. Efficienza scalata. Isoefficienza. Legge di Gustafson.

[1] Cap. 4; [1bis] Par. 3.4, 3.6, 3.7; [4] (vedi Bibliografia)

4. OpenMp - Open specifications for Multi Processing: Caratteristiche di un sistema Shared-Memory. Processi e threads. Modello d'esecuzione parallela fork-join. Direttive al compilatore. Clausole. Runtime library routines. Variabili d'ambiente.

[1] Cap. 3; [1bis] Cap. 2, Par 4.1, 4.2, 4.3, 4.6, 6.1; [3] Cap. 1, 2, 3 e 4; [4] (vedi Bibliografia)

5. Graphics Processing Units e CUDA: Genesi e motivazioni del GP-GPU. Le GPU rispetto alla tassonomia di Flynn. Livelli di parallelismo di una GP-GPU. Cenni all'architettura CUDA.

[1bis] Par 1.5, 6.5; [4] (vedi Bibliografia)

6. La documentazione del software parallelo. Linee guida.

[4] (vedi Bibliografia)

PRINCIPALI ALGORITMI DISCUSSI

1. Algoritmo per la somma di n numeri

Decomposizione del problema. Algoritmo in ambiente MIMD Shared-Memory. Algoritmo in ambiente MIMD Distributed-Memory: I, II, e III strategia; complessità di tempo, speed-up, efficienza. [1] Cap. 3 ; [4]

2. Algoritmo per il prodotto matrice-vettore

Richiamo dell'algoritmo sequenziale. Algoritmi a blocchi: I strategia, II strategia. Parallelizzazione delle due strategie su ambiente MIMD Distributed-Memory: distribuzione dei dati e comunicazioni. III strategia. Speed-up ed efficienza dei vari algoritmi. Algoritmo in ambiente MIMD Shared-Memory. [1] Cap. 3 ; [4]

3. Algoritmo per il prodotto matrice-matrice

Richiamo dell'algoritmo sequenziale. Algoritmi a blocchi: I strategia, II strategia. Parallelizzazione delle due strategie su ambiente MIMD Distributed-Memory: distribuzione dei dati e comunicazioni. III strategia. Strategia Broadcast Multiply Rolling: dettagli. Speed-up ed efficienza dei vari algoritmi. [1] Cap. 3 ; [4]

ESERCITAZIONI IN LABORATORIO

1. Introduzione all'uso del cluster. I nodi biprocessore dual/quad-core.
2. Sviluppo e implementazione di un algoritmo per il calcolo della somma di N numeri reali, in ambiente di calcolo parallelo su architettura MIMD a memoria distribuita, mediante la libreria MPI (implementazione II e III strategia per la comunicazione dei risultati finali come discusso a lezione). **ELABORATO RICHIESTO**
3. Sviluppo e implementazione di un algoritmo per il calcolo della somma di N numeri reali, in ambiente di calcolo parallelo su architettura MIMD a memoria condivisa, mediante la libreria OpenMp.
4. Creazione di una griglia di pxq processi, dove $pxq = P$. Distribuzione degli elementi di un vettore tra i processi di una griglia di pxq processi. Distribuzione dei sottoblocchi di una matrice tra i processi di una griglia di pxq processi.
5. Sviluppo e implementazione di un algoritmo per il calcolo del prodotto matrice-vettore, in ambiente di calcolo parallelo su architettura MIMD a memoria distribuita, mediante la libreria MPI.
La matrice $A \in \mathbb{R}^{m \times n}$ e' distribuita a pxq processi, disposti secondo una topologia a griglia bidimensionale.
6. Sviluppo e implementazione di un algoritmo per il calcolo del prodotto matrice-vettore, in ambiente di calcolo parallelo su architettura MIMD a memoria condivisa, mediante la libreria OpenMp. **ELABORATO RICHIESTO**
7. Sviluppo e implementazione di un algoritmo per il calcolo del prodotto matrice-matrice, in ambiente di calcolo parallelo su architettura MIMD a memoria distribuita, mediante la libreria MPI.
Le matrici $A \in \mathbb{R}^{n \times n}$ e $B \in \mathbb{R}^{n \times n}$ sono distribuite a pxp processi, disposti secondo una topologia a griglia bidimensionale. L'algoritmo implementa la strategia di comunicazione BMR. **ELABORATO RICHIESTO**

BIBLIOGRAFIA

1. A. Murli, Lezioni di Calcolo Parallelo, Liguori Editore.
- 1bis. S. Kumar, Introduction to Parallel Programming, Cambridge University Press
2. Manuale di MPI <https://www.mpi-forum.org/docs/mpi-4.0/mpi40-report.pdf>
3. B. Chapman, G. Jost, R. van der Pas, Using OpenMP. Portable Shared Memory Parallel Programming", The MIT Press, 2007
- 3bis. Il sito di OpenMp <http://openmp.org/>
- 3ter. un manuale di OpenMp <https://computing.llnl.gov/tutorials/openMP/>
4. Appunti e copie di slide scaricabili dall'area *Materiale Didattico* nella pagina del corso sul sito web docenti <http://www.docenti.unina.it> e nel canale Microsoft Teams del corso