

Programma del corso di
Algoritmi Avanzati
Prof.ssa Rossella Petreschi

Il corso è incentrato sul progetto di algoritmi che operano su architetture parallele o distribuite e si propone di spiegare come l'approccio a tale progettazione sia diverso da quello usato per gli algoritmi sequenziali e dipenda strettamente dal tipo di architettura utilizzata.

Programma in dettaglio AA.2010/2011

Concetti generali

Pensare in parallelo: computazione parallela, computazione distribuita. Differenza fra sistemi concorrenti e sistemi sequenziali e fra sistemi distribuiti e sistemi paralleli. Sistemi sincroni e asincroni. Algoritmi paralleli e algoritmi distribuiti. Parallelismo nelle macchine parallele: pipeling, VLIW, parallelismo dei processori: memoria condivisa, rete di interconnessione. Complessità parallela: tempo, costo, speed-up ed efficienza. Complessità distribuita: tempo e trasmissione dell'informazione.

Modelli di macchine parallele

Modello PRAM (Parallel Random Access Machine), modello WD (Work and Depth), modello BSP (Bulk Synchronous Parallel). Analisi delle caratteristiche dei tre modelli presentati e confronto fra gli stessi. Modelli a lettura e scrittura esclusiva e/o concorrente. Macchine SIMD a reti di interconnessione: vettore, matrice, albero, ipercubo. Algoritmi elementari su diversi modelli di macchine parallele: ricerca di un elemento in un vettore, calcolo delle somme prefisse su n elementi.

La trasmissione dell'informazione

Broadcast su PRAM EREW. Broadcast su rete a vettore, su albero binario, mesh e ipercubo.. Il problema del caricamento dei dati su rete a vettore, su albero binario, su mesh. e su ipercubo. Broadcast in un sistema distribuito ad anello. Broadcast con eco su rete generica a connessione fissa. Broadcast asincrono. Specifica del Broadcast basico. Broadcast e ordinamento: approccio FIFO, totale e casuale. Affidabilità della trasmissione dei messaggi: tempo di vita, integrità, mancanza di duplicazioni.

La trasportabilità degli algoritmi

Simulazione di lettura e/o scrittura concorrente su modelli a lettura e/o scrittura esclusiva. Reti combinatorie: fan in e fan out. Teorema di Brent. Trasportabilità fra PRAM con diverso numero di processori. Sistemi sincroni e asincroni. Orologi fisici e logici. Complessità temporale in un sistema asincrono: la relazione “accade prima” e il problema della sincronizzazione dell’orologio. Lo strumento sincronizzatore per simulare sistemi asincroni su sistemi sincroni.

Minimo albero ricoprente

Unicità del minimo albero ricoprente. Proprietà del minimo albero ricoprente. Algoritmo di Prim, Kruskal e Sollin per il calcolo del minimo albero ricoprente. Differenza fra meganodi e frammenti. Algoritmo parallelo per il calcolo dell'albero ricoprente di costo minimo. Calcolo del minimo albero ricoprente in un sistema distribuito. Dimostrazione della correttezza dei due algoritmi presentati e calcolo dei rispettivi costi.

Tecniche parallele di base

Iterazione della prima metà: ricerca del minimo. Tecnica del salto del puntatore: calcolo del rango in una lista. Costruzione del Tour di Eulero (TDE). Tecnica del TDE: ordinamento dei vertici di un albero rispetto alla radice, numerazione in preordine e postordine. Calcolo in un albero dei livelli e dei discendenti di ogni nodo. Introduzione degli operatori left e right per calcolare la numerazione in modo simmetrico e il minimo antecedente comune. Accelerated Cascading: somme, somme prefisse e somme con numero di processori $p < n$.

Algoritmi paralleli su grafi

Partizione degli spigoli di un grafo in cammini semplici: cicli fondamentali, ear. Algoritmo di costo $O(m \log n)$ su PRAM-CRCW per la determinazione di una partizione semplice: analisi della correttezza, calcolo della complessità. Contrazione e decontrazione di alberi binari tramite l'operazione di Rake. Trasformazione di alberi qualsiasi in alberi binari 0/2. Valutazione di espressioni tramite contrazione di alberi. Matching su grafi. Teorema di Tutte, Verifica di identità polinomiali, lemma di isolamento. Algoritmo probabilistico per la ricerca di un matching perfetto di peso minimo. Dal matching perfetto di peso minimo al matching massimo.

Ordinamento nel Parallelo

Ordinamento per inserzione tramite circuiti di comparatori. Monotonicità e principio 0/1. Sequenze bitoniche e pulite. Circuiti di fusione e di ordinamento. Algoritmo di ordinamento in tempo parallelo costante su PRAM CRCW. Algoritmo di ordinamento pari/dispari su PRAM EREW. Le proprietà della K-copertura. Applicazione della “2-copertura” per ottenere un algoritmo di merge di tempo parallelo $\log(n)$ e un algoritmo di ordinamento di costo $O(n \log n)$. Algoritmo di

Cole per l'ordinamento in tempo $O(\log n)$ su PRAM-CREW: algoritmo, prova di correttezza e calcolo della complessità.

Algoritmi per la computazione distribuita

Il problema dell'elezione del leader. Impossibilità di eleggere un leader in processi anonimi. Algoritmo probabilistico per l'elezione del leader in reti anonime. L'elezione del leader in reti ad anello sincrone: algoritmi di tempo $O(n^2)$ e $O(n \log n)$. Calcolo dell'upper e del lower bound per l'elezione del leader in reti asincrone. Mutua esclusione: algoritmo randomizzato con poche variabili condivise. Algoritmi probabilistici per il problema del consenso.

Riferimenti bibliografici

1. Akl S.G. *Progettazione ed analisi degli algoritmi paralleli*, Gruppo Editoriale Jackson.
2. Attiya H., Welch J. *Distributed Computing*, McGraw-Hill.
3. Cormen T.H., Leiserson C.E., Rivest R.L. *Introduzione agli algoritmi*, Jackson Libri.
4. Gallager R.G., Humblet P.A., and Spira P.M., A Distributed Algorithm for Minimum-Weight Spanning Trees. *ACM Transactions on Programming Languages and Systems*, 1(5), 1983, pages 66-77. [p66-gallager.pdf](#)
5. Grama A., Gupta A., Karypis G., Kumar V. *Introduction to parallel Computing*, Pearson-Addison-Wesley.
6. Jaja J. *An introduction to parallel algorithms*, Addison-Wesley.
7. Johnsonbaugh R., Schaefer M. *Algorithms*, Pearson-Addison-Wesley.
8. Reif J.H. *Synthesis of Parallel Algorithms*, Morgan Kaufmann Publishers.
9. Skillicorn D.B., Hill J.M.D., McColl W.F. *Question and answers about BSP*, ftp://ftp.comlab.ox.ac.uk/pub/Packages/BSP/papers/SkillHillMcColl_QA.ps.gz.
10. Valiant L. G. *A bridging model for parallel computation*, *Communication of the ACM*, 33 (8): 101 – 111, August 1990
11. Vishkin U. *Thinking in parallel: some basic data-parallel algorithms and techniques*, <http://www.umiacs.umd.edu/~vishkin/PUBLICATIONS/classnotes.pdf>
12. Zomaya A.Y.H. *Parallel and distributed computing handbook*, McGraw-Hill.

Per ulteriori dettagli, si rimanda alla pagina del corso all'elenco contenente i riferimenti bibliografici relativi alle singole lezioni.