**Evaluasi Pararlel Algoritma**

Tujuan analisis adalah untuk menunjukkan kebenaran algoritma dan untuk memeriksa sifat-sifatnya yang penting dalam hal ini kesesuaian untuk memecahkan masalah yang dihadapi, serta untuk membandingkan kinerja algoritma dan pilih yang terbaik.

Algoritma Asequential adalah urutan langkah-langkah yang didefinisikan secara tepat untuk menyelesaikan suatu masalah, yaitu menghitung data keluaran berdasarkan input data. Presisi dalam menggambarkan langkah-langkah ini diperlukan untuk dapat mengubah algoritma menjadi program komputeritu adalah implementasi dari algoritma.

Kriteria dasar untuk mengevaluasi algoritma sekuensial termasuk waktu berjalan dan kebutuhan memori. Waktu berjalan — atau lebih tepatnya, waktu menghitung prosesor — dan memori yang ditempati adalah jumlah sumber daya yang diperlukan untuk mengeksekusi suatu algoritma. Kriteria evaluasi lain dari algoritma sekuensial termasuk universalitas, dipahami sebagai kemungkinan penerapan suatu algoritma untuk menyelesaikan suatu kelas masalah, kesederhanaan, kemudahan implementasi, dan modifikasi. Mengevaluasi algoritma paralel lebih sulit daripada yang berurutan, karena faktor tambahan (kecuali ukuran data input) harus dipertimbangkan, yaitu jumlah prosesor dari komputer paralel di mana algoritma tersebut diterapkan jumlah kriteria evaluasi juga lebih besar. Kami memperhitungkannya, selain waktu berjalan dan kebutuhan memori, percepatan, biaya komputasi, efisiensi, biaya komunikasi, dan portabilitas. Kriteria yang terakhir ini penting karena sebagai ahasil keragaman arsitektur komputer, algoritma paralel dikembangkan untuk satu Arsitek mungkin sepenuhnya tidak cocok untuk komputer dengan arsitektur yang berbeda.

**Sistem Paralel yang Dapat diskalakan**

Yang dimaksud dengan sistem paralel adalah algoritma paralel yang dijalankan dalam paralel tertentu komputer. Secara informal, skalabilitas terkait dengan kemungkinan peningkatan percepatan atau, ekuivalen, pertahankan efisiensi konstan dengan semakin banyak prosesor yang digunakan untuk perhitungan. Skalabilitas dipengaruhi tidak hanya oleh desain algoritma paralel, tetapi juga oleh biaya komunikasi antara prosesor dan volume mereka kenangan utama, yaitu dengan sifat-sifat komputer paralel di mana suatu algoritma diimplementasikan.

**Fungsi Efisiensi Iso**

mendefinisikan tingkat pertumbuhan ukuran masalah, sehingga untuk menjaga efisiensi yang konstan. Jika fungsi ini tumbuh lambat, maka sistem paralelnya adalah mudah terukur. Sedikit peningkatan dalam ukuran masalah sudah cukup untuk penggunaan yang efisien kemampuan komputasi semakin banyak prosesor. Semakin besar peningkatan ukuran masalah yang menjamin efisiensi konstan, sistem kurang scalable.

Ada juga system yang tidak dapat dihilangkan, di mana efisiensi yang konstan tidak dapat dipertahankan dari tingkat pertumbuhan ukuran masalah.

Pada akhir bagian ini, perlu dicatat bahwa metrik yang didefinisikan sebelumnya, itu adalah waktu berjalan paralel, percepatan, biaya dan efisiensi, yang merupakan dasarnya untuk mengevaluasi kinerja algoritma paralel, adalah fungsi dari dua variabel — a jumlah p prosesor dan ukuran n dari masalah.

**HUKUM AMDAHL**

dikenal sebagai hukum Amdahl mengevaluasi batas atas speedup sebagai fungsi fraksi s dan jumlah prosesor p untuk masalah ukuran tetap n. Singkatnya, hukum Amdahl memungkinkan kita untuk menjawab pertanyaan sejauh mana algoritma sekuensial memecahkan masalah ukuran yang diberikan n dapat dipercepat. Hukum membuat dua asumsi penyederhanaan.

**METRIC KARP -FLATT**

mengusulkan perhitungan eksperimental berurutan fraksi f dari perhitungan algoritma paralel, yang memungkinkan evaluasi diperoleh speedup.

**ALGORITMA UNTUK MODEL MEMORY BERBAGI**

Terdiri dari beberapa :

1. Menemukan Jumlah Minimum dan Jumlah Unsur dalam Waktu O (log n)
2. Teorema Brent
3. Perhitungan Awalan
4. Menemukan Minimum dalam O (1) Waktu
5. Mengurutkan dalam O (log n) Waktu
6. Perkalian Matriks – Matriks
7. Perhitungan pada Daftar
8. Metode Siklus Euler

**ALGORITMA UNTUK MODEL JARINGAN**

Terdiri dari beberapa :

1. Multiplikasi Matriks – Vektor dalam Jaringan Torus Satu Dimensi
2. Perkalian Matriks – Matriks dalam Jaringan Torus Dua Dimensi
3. Pengurangan Operasi di Jaringan Cube
4. Siaran di Jaringan Cube
5. Komputasi Awalan dalam Jaringan Cube