BAB V
KESIMPULAN

5.1 Kesimpulan

Setelah dilakukan pengujian, pengambilan data, perhitungan waktu baik secara ujicoba maupun berdasarkan teori, serta dilakukan analisis dari masing-masing algoritma, Maka dapat diambil kesimpulan-kesimpulan sebagai berikut:

1. Dekomposisi data yang dilakukan dari kompleksitas matriks paralel mulai dari $\frac{N^3}{P}$, $\frac{N^3}{P^2}$ sampai dengan $\frac{N^3}{P\sqrt{P}}$ hanya ditujukan untuk mengefisiensikan besar data yang akan ditransfer melalui jaringan dan besarnya memori yang dibutuhkan untuk menyimpan besarnya matriks masing-masing *node*, namun tidak dapat menghasilkan perbedaan waktu paralel yang signifikan dari masing-masing algoritma, dengan perbandingan Algoritma *Row-Column Striped Block* dihitung berdasarkan teori yaitu 938 detik (Algoritma *Broadcast* Matriks-B), 944 detik (Algoritma *Row-Column Striped Block)*, 915 detik (Algoritma *Checkboard-Blok*) dengan nilai-N 2048 dan jumlah prosesor (P) adalah 4-Prosesor.
2. Dengan adanya penambahan jumlah prosesor berdampak sangat besar untuk mendapatkan dekomposisi yang lebih kecil dan peningkatan kecepatan waktu paralel yang dihasilkan serta percepatan (*speedup*) yang mendekati besar nilai penambahan jumlah prosesor (P), dengan perbandingan waktu 1-Prosesor sebesar 3720 detik dengan waktu paralel yang didapat tertera pada poin satu dengan kondisi yang sama.
3. Sedangkan berdasarkan uji coba agoritma *Row-Column Striped Block* lebih baik dari algoritma lainnya, hal ini disebabkan karena proses pertukaran posisi dari Matriks B yang di tujukan untuk memenuhi kekurangan dari kemampuan bahasa-C dalam melakukan proses penyimpanan data berdasarkan urutan baris (*row-major*) dan kemampuan fungsi *Scatter* dari MPI hanya bisa berdasarkan baris, Sedangkan kebutuhan dari algoritma ini yaitu melakukan *Scatter* berdasarkan kolom dari Matriks B, Proses pertukaran posisi dari urutan baris (*row-major*)

menjadi urutan kolom (*column-major*) ini menyebabkan perubahan dari perkalian antara baris dari Matriks A dengan kolom-kolom dari Matriks B menjadi perkalian antara baris dari Matriks A dengan baris dari Matriks B.

4. Pemilihan nilai N (banyaknya baris dan kolom matriks) terbaik untuk perkalian matriks paralel dalam penelitian ini adalah nilai N yang lebih besar dari 128, dikarenakan waktu komputasi lebih besar daripada waktu komunikasi. Sedangkan untuk nilai N lebih kecil dari 256, waktu komunikasi melebihi waktu komputasi sebesar dua kalinya.
5. Jika waktu komunikasi lebih besar dari waktu komputasi, kejadian ini menyebabkan *overhead*, dimana prosesor lebih banyak tidak bekerja, menunggu data yang akan diproses tiba.

5.2 Saran

Dikarenakan kekurangan dari kemampuan bahasa-C dan MPI, sebaiknya perancangan algoritma perkalian matriks paralel pada tugas akhir ini dapat dikembangkan dengan metode-metode lain yang terdapat dalam fitur-fitur MPI.

Penambahan jumlah memori internal dan kecepatan prosesor yang berbeda dapat dilakukan untuk kedepannya. Sebagai bagian perubahan dari homogenitas ke heterogenitas serta dekomposisi dan kemampuan prosesor mempengaruhi FLOPS dalam perkalian matriks paralel dengan tingkatan perhitungan waktu yang lebih kompleks.