



**UNIVERSITAS INDONESIA**

**PR 3 PEMROGRAMAN DI LINGKUNGAN GPU**

**LAPORAN TUGAS PEMROGRAMAN PARALEL**

**KELOMPOK III**

**Muhammad Fathurachman 1506706276**

**Otniel Yosi Viktorisa 1506706295**

**Yohanes Gultom 1506706345**

**FAKULTAS ILMU KOMPUTER  
PROGRAM STUDI MAGISTER ILMU KOMPUTER  
DEPOK  
MEI 2016**

# DAFTAR ISI

<b>Daftar Isi</b>	<b>ii</b>
<b>Daftar Gambar</b>	<b>iv</b>
<b>1 LINGKUNGAN PERCOBAAN</b>	<b>1</b>
1.1 Lingkungan Pengembangan . . . . .	1
1.2 Lingkungan Percobaan . . . . .	1
1.2.1 Server GPU Fasilkom Universitas Indonesia (UI) . . . . .	1
1.2.2 Personal Computer (Laptop) . . . . .	2
1.2.3 Cluster Rocks University of California San Diego (UCSD) . . . . .	2
<b>2 PENGENALAN CUDA</b>	<b>4</b>
2.1 Eksperimen . . . . .	4
2.1.1 Eksperimen Perilaku Thread dan Block . . . . .	4
2.1.1.1 Deskripsi Program . . . . .	4
2.1.1.2 Hasil Eksperimen . . . . .	4
2.1.2 Eksperimen Perilaku Thread dan Block . . . . .	4
2.1.2.1 Deskripsi Program . . . . .	4
2.1.2.2 Hasil Eksperimen . . . . .	5
2.2 Kesimpulan . . . . .	5
<b>3 PERKALIAN MATRIKS-VEKTOR &amp; MATRIKS BUJURSANGKAR</b>	<b>6</b>
3.1 Eksperimen . . . . .	6
3.1.1 Perbandingan Program Sekuensial dan CUDA . . . . .	6
3.1.1.1 Deskripsi Program . . . . .	6
3.1.1.2 Hasil Eksperimen . . . . .	6
3.1.2 Program CUDA dengan Variasi ukuran <i>Grid/Block</i> . . . . .	6
3.1.2.1 Deskripsi Program . . . . .	7
3.1.2.2 Hasil Eksperimen . . . . .	7
3.1.3 Program CUDA <i>Shared Memory</i> , CUBLAS dan MPI . . . . .	7
3.1.3.1 Deskripsi Program . . . . .	7
3.1.3.2 Hasil Eksperimen . . . . .	7
3.2 Kesimpulan . . . . .	7

<b>4</b>	<b>CONJUGATE GRADIENT METHOD</b>	<b>8</b>
4.1	Pendahuluan . . . . .	8
4.2	Eksperimen . . . . .	9
4.3	Kesimpulan . . . . .	9
<b>5</b>	<b>MOLECULAR DYNAMICS: AMBER</b>	<b>10</b>
5.1	Pendahuluan . . . . .	10
5.2	Eksperimen . . . . .	10
5.3	Kesimpulan . . . . .	11
<b>6</b>	<b>KONTRIBUSI</b>	<b>12</b>

## **DAFTAR GAMBAR**

4.1	Algoritma Conjugate Gradient Method. . . . .	8
-----	----------------------------------------------	---

# **BAB 1**

## **LINGKUNGAN PERCOBAAN**

### **1.1 Lingkungan Pengembangan**

Program paralel yang digunakan di dalam eksperimen ini dibuat menggunakan bahasa C yang di-*compile* menggunakan pustaka OpenMPI pada sistem operasi Linux Ubuntu dan Mint.

Kode program-program dan laporan eksperimen ini kami simpan menggunakan layanan GitHub di alamat <https://github.com/yohanesgultom/parallel-programming-assignment>. Hal ini kami lakukan untuk mempermudah kolaborasi dalam pembuatan program dan laporan.

### **1.2 Lingkungan Percobaan**

Dalam eksperimen yang dilakukan, kelompok kami menggunakan mesin-mesin dengan GPU NVIDIA dan *cluster* CPU Rocks University of California San Diego (UCSD).

#### **1.2.1 Server GPU Fasikom Universitas Indonesia (UI)**

*Server* dengan GPU NVIDIA merupakan lingkungan utama percobaan ini. Kelompok kami melakukan percobaan di 2 buah server Fasikom UI:

1. Server GTX 980 (152.118.31.27)
  - NVIDIA GTX 980 2048 CUDA Cores 4 GB GRAM
  - Intel(R) Core(TM) i7-3770 CPU 4 cores @ 3.40GHz
  - RAM 2x8 GB DDR3 1600 Mhz
  - SSD SAMSUNG MZ7TD128 128 GB
  - OS Debian 7 Wheezy x64
  - CUDA 7.0
  - Amber 14 dan Amber Tools 15
2. Server GTX 970 (152.118.31.34)

- NVIDIA GTX 970 1664 CUDA Cores 4 GB GRAM
- Intel(R) Core(TM) i7-3770 CPU 4 cores @ 3.40GHz
- RAM 2x8 GB DDR3 1600 Mhz
- SSD SAMSUNG MZ7TD128 128 GB
- OS Debian 7 Wheezy x64
- CUDA 7.0
- Amber 14 dan Amber Tools 15

Semua *server* ini dapat diakses dari jaringan Fasilkom menggunakan protokol *SSH* dan *credential Single Sign On* (SSO) UI. Sedangkan dari luar jaringan UI, semua *server* ini dapat diakses dengan masuk lebih dahulu ke [kawung.cs.ui.ac.id](http://kawung.cs.ui.ac.id) menggunakan protokol *SSH* juga.

### 1.2.2 Personal Computer (Laptop)

Sebagai bahan perbandingan, kami juga menggunakan PC (*notebook*) yang juga menggunakan GPU NVIDIA yang mendukung CUDA:

- NVIDIA 940M 384 CUDA Cores 2 GB GRAM
- Intel(R) Core(TM) i7-5500U CPU 2 cores @ 3.40GHz
- RAM 8 GB DDR3 1600 Mhz
- SSD Crucial 250 GB
- OS Ubuntu 15.10 x64
- CUDA 7.5
- Amber 14 dan Amber Tools 15

### 1.2.3 Cluster Rocks University of California San Diego (UCSD)

*Cluster Rocks*<sup>1</sup> milik University of California San Diego (UCSD) ini dapat diakses pada alamat [nbc-233.ucsd.edu](http://nbc-233.ucsd.edu) menggunakan protokol *SSH* dari komputer yang telah didaftarkan *public key* nya. Berdasarkan informasi dari aplikasi *monitoring* Ganglia<sup>2</sup>, cluster ini terdiri 10 *nodes* dengan total 80 prosesor.

---

<sup>1</sup>[www.rocksclusters.org](http://www.rocksclusters.org)

<sup>2</sup><http://nbc-233.ucsd.edu/ganglia>

Pada *cluster* ini sudah terpasang pustaka komputasi paralel OpenMPI<sup>3</sup> dan MPICH<sup>4</sup> serta paket dinamika molekular AMBER. Program paralel MPI dan eksperimen AMBER dijalankan mekanisme antrian *batch-jobs* untuk menjamin ketersediaan sumberdaya komputasi (*computing nodes*) saat program dieksekusi. Pengaturan eksekusi program paralel ini ditangani oleh *Sun Grid Engine*<sup>5</sup> yang juga tersedia dalam paket Rocks.

---

<sup>3</sup><https://www.open-mpi.org/>

<sup>4</sup><https://www.mpich.org/>

<sup>5</sup><http://www.rocksclusters.org/roll-documentation/sge/5.4/>

## BAB 2

### PENGENALAN CUDA

#### 2.1 Eksperimen

Pada bagian ini kami melakukan 2 eksperimen untuk mengamati perilaku *block* dan *thread* pada kernel dan mengamati pengaruh ukuran data terhadap eksekusi *kernel*

##### 2.1.1 Eksperimen Perilaku Thread dan Block

@todo

###### 2.1.1.1 Deskripsi Program

@todo

###### 2.1.1.2 Hasil Eksperimen

@todo

##### 2.1.2 Eksperimen Perilaku Thread dan Block

@todo

###### 2.1.2.1 Deskripsi Program

@todo



### 2.1.2.2 Hasil Eksperimen

@todo

## 2.2 Kesimpulan

@todo

## BAB 3

### PERKALIAN MATRIKS-VEKTOR & MATRIKS BUJURSANGKAR

#### 3.1 Eksperimen

Pada bagian ini kami melakukan eksperimen perkalian matriks-vektor dan matriks bujursangkar pada CPU (sekuensial), CUDA, CUDA dengan *shared memory*, CUBLAS dan CPU (*cluster MPI*) untuk membandingkan kinerjanya.

##### 3.1.1 Perbandingan Program Sekuensial dan CUDA

@todo

###### 3.1.1.1 Deskripsi Program

@todo

###### 3.1.1.2 Hasil Eksperimen

@todo

##### 3.1.2 Program CUDA dengan Variasi ukuran *Grid/Block*

@todo

#### 3.1.2.1 Deskripsi Program

@todo

#### 3.1.2.2 Hasil Eksperimen

@todo

### 3.1.3 Program CUDA *Shared Memory*, CUBLAS dan MPI

@todo

#### 3.1.3.1 Deskripsi Program

@todo

#### 3.1.3.2 Hasil Eksperimen

@todo

## 3.2 Kesimpulan

@todo

## BAB 4

### CONJUGATE GRADIENT METHOD

#### 4.1 Pendahuluan

Conjugate gradient method digunakan untuk menyelesaikan persamaan linear  $Ax = b$  di mana matriks koefisiennya bersifat simetris dan definit positif. Matriks  $A$   $n \times n$  dikatakan simetris jika  $a_{ij} = a_{ji}$  untuk  $i, j = 1, \dots, n$ . Matriks  $A$  dikatakan definit positif jika untuk setiap vektor  $x$  bukan nol, perkalian skalar  $x \cdot Ax$  menghasilkan nilai lebih besar dari nol. Algoritma conjugate gradient method ditunjukkan pada Gambar 4.1.  $r_k$  merupakan sisa atau selisih antara  $b$  dengan  $Ax_k$ , sedangkan  $p_k$  merupakan *search direction*.

```

k = 0; x0 = 0; r0 = b
while (||rk||2 > tolerance) and (k < max_iter)
    k++
    if k = 1
        p1 = r0
    else
        βk =  $\frac{r_{k-1} \cdot r_{k-1}}{r_{k-2} \cdot r_{k-2}}$  // minimize ||pk - rk-1||
        pk = rk-1 + βkpk-1
    endif
    sk = Apk
    αk =  $\frac{r_{k-1} \cdot r_{k-1}}{p_k \cdot s_k}$  // minimize q(xk-1 + αpk)
    xk = xk-1 + αkpk
    rk = rk-1 - αksk
endwhile
x = xk

```

**Gambar 4.1:** Algoritma Conjugate Gradient Method.

Pada implementasi paralel CGM, matriks  $A$  akan didistribusikan menggunakan fungsi `MPI_scatter` kepada setiap proses dan masing-masing proses mendapat sebanyak  $n \times n / p$  data. Algoritma tersebut dijalankan di setiap proses untuk setiap bagian distribusi matriks  $A$ .

## 4.2 Eksperimen

@todo

## 4.3 Kesimpulan

@todo

## BAB 5

### MOLECULAR DYNAMICS: AMBER

#### 5.1 Pendahuluan

AMBER (*Assisted Model Building with Energy Refinement*) adalah paket program untuk menjalankan simulasi dinamika molekular (*molecular dynamics*) yang dikembangkan oleh *University of California, San Fransico*. Alamat *website* resmi dari AMBER adalah <http://ambermd.org>.

AMBER digunakan untuk berbagai eksperimen dinamika molekular seperti simulasi pergerakan fisik dari atom dan molekular, pemodelan protein dan eksperimen yang terkait dengan perancangan obat (*drug discovery*).

AMBER didistribusikan dalam dua bagian:

1. AMBER (berbayar, versi terakhir 14)
2. AMBERTool (*opensource* GPL, versi terakhir 15)

AMBER memiliki dua mode instalasi, yaitu berbasis CPU (dengan OpenMPI) dan berbasis GPU (dengan CUDA). Petunjuk instalasi dapat dilihat di <http://jswails.wikidot.com/installing-amber14-and-ambertools14>.

#### 5.2 Eksperimen

Percobaan dilakukan dengan menjalankan 6 buah eksperimen yang disediakan pada AMBER GPU *Benchmark Suite* yang diperoleh dari *website* resmi AMBER [http://ambermd.org/Amber14\\_Benchmark\\_Suite.tar.bz2](http://ambermd.org/Amber14_Benchmark_Suite.tar.bz2). Enam buah eksperimen yang kami jalankan adalah:

1. TRPCAGE Production (304 atoms, 250.000 nsteps)
2. Myoglobin Production (2,492 atoms, 25.000 nsteps)
3. JAC Production NVE (23,558 atoms, 25.000 nsteps)
4. Nucleosome Production (25,095 atoms, 600 nsteps)
5. Factor IX Production NVE (90,906 atoms, 10.000 nsteps)

## 6. Cellulose Production NVE (408,609 atoms, 5.000 nsteps)

Pada eksperimen ini kami mengamati kecepatan proses (ns/day) pada mesin yang berbeda, yaitu:

1. UCSD cluster 32 CPU + OpenMPI (nbc-233.ucsd.edu)
2. GTX 980 + CUDA 7.0 (Fasilkom UI)
3. GTX 970 + CUDA 7.0 (Fasilkom UI)
4. GT 940M + CUDA 7.5 (PC/notebook)

**@todo**

## 5.3 Kesimpulan

**@todo**

## **BAB 6**

### **KONTRIBUSI**

Kontribusi tiap anggota kelompok pada tugas ini adalah sebagai berikut:

**Otniel Yosi Viktorisa:**

1. Topik 1: eksperiman & presentasi perilaku *block* dan *grid* pada CUDA
2. Topik 3: eksperimen & presentasi *Conjugate Gradient Method*

**Yohanes Gultom:**

1. Topik 2: eksperiman, presentasi & laporan perkalian matriks-vektor dan matriks bujursangkar dengan CPU (sekuensial), CUDA, CUDA dengan *shared memory*, CUBLAS dan CPU (*cluster MPI*)
2. Instalasi AMBER 14 dan AMBER Tools 15 pada server GTX 980 dan GTX 970 Fasilkom UI
3. Topik 4: eksperimen, presentasi & laporan AMBER dengan enam jenis *benchmarking set* dari *website* resmi AMBER pada 3 GPU dan 1 cluster (UCSD)

**Muhammad Fathurachman:**

1. Topik 2: mencoba eksperiman matriks-vektor dan matriks bujursangkar dengan CUDA