



**UNIVERSITAS INDONESIA**

**Studi Kinerja Algoritma Paralel dengan MPICH2 dan Cilk++  
pada Prosesor *Multicore***

**TESIS**


**Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister Teknik**

**DION OGI  
0806424314**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
KEKHUSUSAN JARINGAN INFORMASI DAN MULTIMEDIA  
DEPOK  
JULI 2010**

## HALAMAN PERNYATAAN ORISINALITAS

**Tesis ini adalah hasil karya saya sendiri,  
dan semua sumber yang dikutip maupun dirujuk  
telah saya nyatakan dengan benar.**

**Nama : Dion Ogi**  
**NPM : 0806424314**  
**Tanda Tangan : **  
**Tanggal : Juli 2010**

## HALAMAN PENGESAHAN

Tesis ini diajukan oleh :

Nama : Dion Ogi  
NPM : 0806424314  
Program Studi : Teknik Elektro  
Judul : Studi Kinerja Algoritma Paralel dengan MPICH2 dan Cilk++  
pada Prosesor *Multicore*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh Gelar Magister Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

### DEWAN PENGUJI

Pembimbing : Prof. Dr. Ir. Bagio Budiardjo, M.Sc. (.....)

Penguji : Prof. Dr. Ir. Kalamullah Ramli, M.Eng. (.....)

Penguji : Prof. Dr. Ir. Riri Fitri Sari, M.Sc., M.M. (.....)

Penguji : Dr. Ir. Anak Agung Putri Ratna, M.Eng. (.....)

Ditetapkan di : Depok

Tanggal : Juli 2010

## KATA PENGANTAR

*Alhamdulillah* *robbil 'alamin*, segala puji bagi Allah SWT. atas segala nikmat yang diberikan kepada penulis sehingga tesis ini dapat penulis selesaikan. Sholawat serta salam semoga selalu tercurah kepada Baginda Nabi Muhammad SAW., keluarga, para sahabat, dan ummatnya yang istiqomah hingga akhir zaman. Penulisan tesis ini dilakukan sebagai salah satu persyaratan untuk mendapatkan gelar Magister Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia. Penulis mendapatkan bimbingan dan dukungan yang tidak terduga dari berbagai pihak sehingga perkenankan penulis mengucapkan terima kasih kepada :

- (1) Prof. Dr. Ir. Bagio Budiardjo, M.Sc. selaku dosen pembimbing yang dengan penuh kesabaran memberikan ilmu pengetahuan, bimbingan, dan dukungan moril selama penulisan tesis ini;
- (2) Pimpinan Lembaga Sandi Negara yang telah memberikan kesempatan tugas belajar pada penulis serta jajaran dan stafnya yang telah memberikan dukungan fasilitas penelitian dan kesempatan berdiskusi dalam rangka penulisan tesis ini;
- (3) Orang tua, istri dan kedua buah hati penulis yang telah memberikan doa restu, dukungan moril serta materil yang tidak terduga;
- (4) Teman-teman JIM '08, rekan-rekan REN, sahabat-sahabat LQ dan FUI yang juga telah memberikan dukungan sehingga penulis dapat menyelesaikan tugas ini.

Semoga Allah SWT. membalas segala kebaikan semua pihak yang telah membantu penulis dengan kebaikan yang berlipat ganda. Penulis berharap tesis ini bermanfaat bagi pengembangan dan penerapan ilmu pengetahuan.

Depok, Juli 2010

Penulis

## HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini :

Nama : Dion Ogi  
NPM : 0806424314  
Program Studi : Kekhususan Jaringan Informasi dan Multimedia  
Departemen : Teknik Elektro  
Fakultas : Teknik  
Jenis karya : Tesis

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Non-eksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

### **Studi Kinerja Algoritma Paralel dengan MPICH2 dan Cilk++ pada Prosesor *Multicore***

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-eksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.  
Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : Juli 2010

Yang menyatakan



( Dion Ogi )

## ABSTRAK

Nama : Dion Ogi  
Program Studi : Teknik Elektro  
Judul : Studi Kinerja Algoritma Paralel dengan MPICH2 dan Cilk++  
pada Prosesor *Multicore*

Perkembangan teknologi prosesor seperti *multicore* menjadi bagian dari pengembangan HPC. Pengujian terhadap algoritma paralel dengan MPICH2 dan Cilk++ menggunakan variabel jumlah proses dan *core* diharapkan dapat menunjukkan perbedaan kinerja (waktu proses) yang signifikan.

Analisis yang dilakukan pada hasil eksperimen menunjukkan implementasi algoritma paralel dengan MPICH2 memberikan *speed-up* hingga 5,987 pada perkalian matriks dan 4,175 pada pengurutan data (*sorting*) serta efisiensi hingga 0,967 dan 0,721 pada masing-masing aplikasi.

Sedangkan implementasi algoritma paralel dengan *Cilk++* memberikan *speed-up* relatif hingga 25,126 pada perkalian matriks dan 9,105 pada pengurutan data (*sorting*) serta efisiensi relatif hingga 6,672 dan 2,593 pada masing-masing aplikasi.

Kata kunci :

Komputasi paralel, *cluster*, *multicore*, algoritma paralel, *multithreading*

## ABSTRACT

Name : Dion Ogi  
Study Program : Electrical Engineering  
Title : Performance Study of Parallel Algorithms using MPICH2 and Cilk++ on Multicore Processors

Developments of processors technology like multicore become apart of HPC developments. The experiments that apply the parallel algorithms using MPICH2 and Cilk++ with variables at process and core amount are expected can show significant differences of the performance (process time).

The analysis done to the experiment results showed that the parallel algorithms implementation using MPICH2 give speed-up until 5.987 at matrix multiplication and 4.175 at data sorting also give efficiency until 0.967 and 0.721 at each following applications.

While the parallel algorithms implementation using Cilk++ give relative speed-up until 25.126 at matrix multiplication and 9.105 at data sorting also give relative efficiency until 6.672 and 2.593 at each following applications.

Keywords:

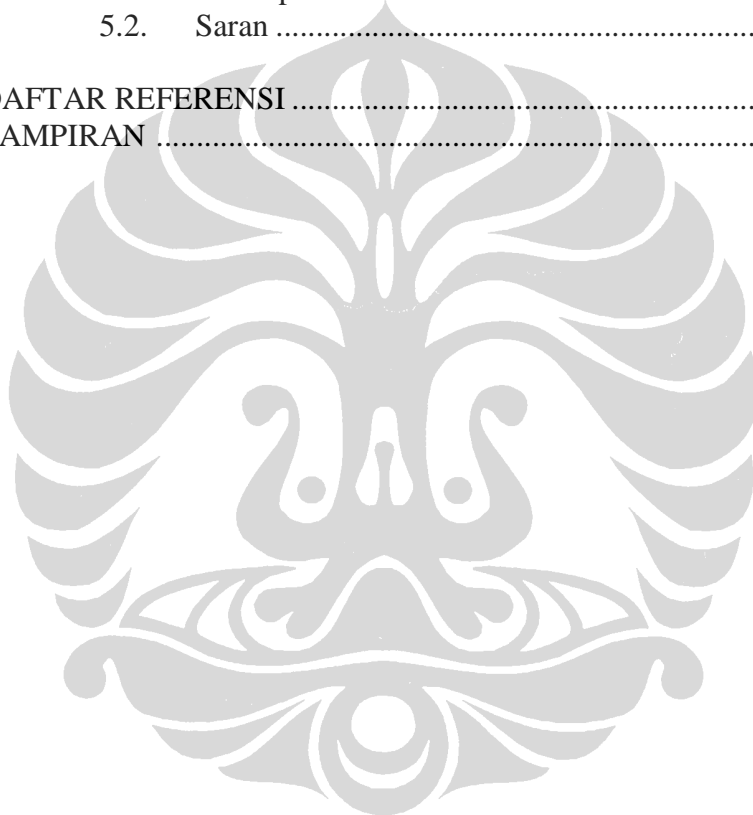
Parallel computation, cluster, multicore, parallel algorithm, multithreading

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PERNYATAAN ORISINALITAS .....	ii
HALAMAN PENGESAHAN .....	iii
KATA PENGANTAR .....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS .....	v
ABSTRAK.....	vi
DAFTAR ISI .....	vii
DAFTAR TABEL .....	ix
DAFTAR GAMBAR .....	x
DAFTAR LAMPIRAN .....	xi
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Perumusan Masalah .....	2
1.3. Tujuan Penelitian .....	2
1.4. Batasan Masalah .....	2
1.5. Metode Penelitian .....	3
1.6. Sistematika Penulisan .....	4
<b>BAB II LANDASAN TEORI.....</b>	<b>5</b>
2.1. Komputasi Paralel .....	5
2.1.1. Arsitektur Komputer Paralel .....	6
2.1.2. Algoritma Paralel .....	11
2.1.3. Algoritma <i>Multithreading</i> .....	13
2.2. Aplikasi Pengujian .....	15
2.2.1. Perkalian Matriks .....	15
2.2.2. Pengurutan Data ( <i>Sorting</i> ).....	15
2.2.3. Algoritma Serial Aplikasi Pengujian .....	16
<b>BAB III PERANCANGAN SISTEM.....</b>	<b>18</b>
3.1. Konfigurasi <i>Cluster PC Multicore</i> .....	18
3.1.1. Konfigurasi Hardware.....	18
3.1.2. Konfigurasi Software.....	19
3.2. Program Aplikasi Paralel .....	21
3.2.1. Algoritma Paralel dengan MPICH.....	22
3.2.2. Algoritma Paralel dengan Cilk++.....	23
<b>BAB IV PENGUJIAN DAN ANALISIS.....</b>	<b>24</b>
4.1. Skenario Pengujian .....	24
4.1.1. Percepatan ( <i>speed-up</i> ).....	24
4.1.2. Efisiensi .....	24
4.1.3. Sampel dan Variabel Pengujian .....	25



4.2.	Hasil yang Diperoleh .....	26
4.2.1.	Hasil Pengujian Algoritma Paralel untuk Aplikasi Perkalian Matriks .....	26
4.2.2.	Hasil Pengujian Algoritma Paralel untuk Aplikasi <i>Sorting</i> .....	27
4.3.	Analisis Hasil Pengujian .....	29
4.3.1.	Analisis Percepatan dan Efisiensi Algoritma Paralel untuk Aplikasi Perkalian Matriks.....	29
4.3.2.	Analisis Percepatan dan Efisiensi Algoritma Paralel untuk Aplikasi <i>Sorting</i> .....	34
<b>BAB V KESIMPULAN DAN SARAN.....</b>		<b>39</b>
5.1.	Kesimpulan .....	39
5.2.	Saran .....	40
DAFTAR REFERENSI .....		42
LAMPIRAN .....		44



## DAFTAR TABEL

Tabel 2.1.	Komposisi Arsitektur HPC Tahun 2009 .....	6
Tabel 4.1.	Waktu Proses Rata-rata Perkalian Matriks Algoritma Paralel MPICH2 dengan Jumlah Proses 4, 8, 12, dan 16.....	26
Tabel 4.2.	Waktu Proses Rata-rata Perkalian Matriks Algoritma Paralel MPICH2 dengan Jumlah Proses 1, 2, 3, dan 4.....	26
Tabel 4.3.	Waktu Proses Rata-rata Perkalian Matriks Algoritma Paralel Cilk++ dengan Jumlah <i>Thread</i> 1, 2, 3, dan 4.....	26
Tabel 4.4.	Waktu Proses Rata-rata <i>Sorting</i> Algoritma Paralel MPICH2 dengan Jumlah Proses 4, 8, 12, dan 16 .....	27
Tabel 4.5.	Waktu Proses Rata-rata <i>Sorting</i> Algoritma Paralel MPICH2 dengan Jumlah Proses 1, 2, 3, dan 4 .....	28
Tabel 4.6.	Waktu Proses Rata-rata <i>Sorting</i> Algoritma Paralel Cilk++ dengan Jumlah <i>Thread</i> 1, 2, 3, dan 4 .....	28
Tabel 4.7.	Percepatan Perkalian Matriks Algoritma Paralel MPICH2 pada <i>Cluster PC Quadcore</i> .....	30
Tabel 4.8.	Percepatan Perkalian Matriks Algoritma Paralel MPICH2 dan Cilk++ pada <i>PC Quadcore</i> .....	30
Tabel 4.9.	Percepatan Perkalian Matriks Algoritma Paralel Cilk++ pada <i>PC Quadcore</i> .....	30
Tabel 4.10.	Efisiensi Perkalian Matriks Algoritma Paralel MPICH2 pada <i>Cluster PC Quadcore</i> .....	31
Tabel 4.11.	Efisiensi Perkalian Matriks Algoritma Paralel pada <i>PC Quadcore</i> .....	31
Tabel 4.12.	Efisiensi Perkalian Matriks Algoritma Paralel Cilk++ pada <i>PC Quadcore</i> .....	32
Tabel 4.13.	Percepatan <i>Sorting</i> Algoritma Paralel MPICH2 pada <i>Cluster PC Quadcore</i> .....	34
Tabel 4.14.	Percepatan <i>Sorting</i> Algoritma Paralel MPICH2 dan Cilk++ pada <i>PC Quadcore</i> .....	35
Tabel 4.15.	Percepatan <i>Sorting</i> Algoritma Paralel Cilk++ pada <i>PC Quadcore</i> .....	35
Tabel 4.16.	Efisiensi <i>Sorting</i> Algoritma Paralel MPICH2 pada <i>Cluster PC Quadcore</i> .....	36
Tabel 4.17.	Efisiensi <i>Sorting</i> Algoritma Paralel MPICH2 dan Cilk++ pada <i>PC Quadcore</i> .....	36
Tabel 4.18.	Efisiensi <i>Sorting</i> Algoritma Paralel Cilk++ pada <i>PC Quadcore</i> .....	36

## DAFTAR GAMBAR

Gambar 2.1.	Tren Paralelisme Prosesor .....	5
Gambar 2.2.	Skema SISD .....	7
Gambar 2.3.	Skema SIMD.....	7
Gambar 2.4.	Skema MISD.....	8
Gambar 2.5.	Skema MIMD.....	8
Gambar 2.6.	Arsitektur HPC Tahun 1993 - 2009 .....	9
Gambar 2.7.	Arsitektur <i>Cluster</i> Komputer.....	10
Gambar 2.8.	Skema Prosesor <i>Multicore</i> .....	11
Gambar 2.9.	Ilustrasi <i>Sorting</i> .....	16
Gambar 2.10.	Algoritma Serial Perkalian Matriks.....	16
Gambar 2.11	Algoritma Serial Pengurutan Data .....	17
Gambar 3.1.	<i>Cluster</i> 4 PC <i>Multicore</i> .....	18
Gambar 3.2.	Skema Software pada <i>Cluster</i> PC <i>Multicore</i> .....	21
Gambar 3.3.	Algoritma Paralel Perkalian Matriks dengan MPICH2 .....	22
Gambar 3.4.	Algoritma Paralel <i>Sorting</i> dengan MPICH2 .....	22
Gambar 3.5.	Algoritma Paralel dengan Cilk++ untuk Perkalian Matriks...	23
Gambar 3.6.	Algoritma Paralel dengan Cilk++ untuk Pengurutan Data ....	23
Gambar 4.1.	Grafik Perbandingan Waktu Proses Algoritma Paralel MPICH2 dan Cilk++ untuk Aplikasi Perkalian Matriks.....	27
Gambar 4.2.	Grafik Perbandingan Waktu Proses Algoritma Paralel MPICH2 dan Cilk++ untuk Aplikasi <i>Sorting</i> .....	28
Gambar 4.3.	Grafik Percepatan Algoritma Paralel MPICH2 dan Cilk++ untuk Aplikasi Perkalian Matriks .....	32
Gambar 4.4.	Grafik Percepatan Algoritma Paralel Cilk++ untuk Aplikasi Perkalian Matriks .....	33
Gambar 4.5.	Grafik Efisiensi Algoritma Paralel Cilk++ untuk Aplikasi Perkalian Matriks .....	34
Gambar 4.6.	Grafik Percepatan Algoritma Paralel MPICH2 dan Cilk++ untuk Aplikasi <i>Sorting</i> .....	37
Gambar 4.7.	Grafik Percepatan Algoritma Paralel Cilk++ untuk Aplikasi <i>Sorting</i> .....	38
Gambar 4.8.	Grafik Efisiensi Algoritma Paralel MPICH2 dan Cilk++ untuk Aplikasi <i>Sorting</i> .....	38

## DAFTAR LAMPIRAN

Lampiran I	Listing Program Perkalian Matriks dengan MPICH2 .....	44
Lampiran II	Listing Program Perkalian Matriks dengan Cilk++ .....	49
Lampiran III	Listing Program <i>Sorting</i> dengan MPICH2 .....	52
Lampiran IV	Listing Program <i>Sorting</i> dengan Cilk++.....	58

