

ANALISIS PERFORMANSI ALGORITMA GENETIKA PADA PENJADWALAN PROSES SISTEM OPERASI

Muhammad Adam Graphiandana¹, Endro Ariyanto ², Tri Brotoharsono³

¹Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

Abstrak

Penjadwalan pada Sistem Operasi adalah salah satu faktor penting pada komputer secara umum. Proses penjadwalan pada sebuah sistem operasi dapat dikatakan sebagai proses yang dialokasikan ke processor sehingga average response time, turnaround time dan normalized turnaround time dari sistem menjadi seminimal mungkin. Permasalahan penjadwalan termasuk dalam permasalahan NP-Complete dimana tidak ada algoritma yang dapat menghasilkan hasil yang optimal dalam waktu polinomial. Oleh karena itu, untuk masalah penjadwalan yang dapat diukur dapat ditemukan solusi yang optimal. Ada cara untuk menemukan solusi yang adaptif dan dapat diandalkan untuk permasalahan asli proses penjadwalan. Algoritma Genetika merupakan teknik pencarian secara heuristik dimana mensimulasikan proses dengan cara seleksi alam dan evolusi. Algoritma Genetika bekerja dengan cara menyamai proses natural dari evolusi yang artinya maju dari solusi optimal. Algoritma Genetika memiliki prinsip utama untuk menciptakan populasi dari individual yang dapat beradaptasi pada lingkungan dimana lingkungan tersebut mempunyai perilaku seperti sistem natural.

Algoritma Genetika memiliki kualitas proses yang lebih baik dibandingkan dengan algoritma Multilevel Feedback Queue jika bekerja pada penjadwalan dengan jumlah proses yang tidak banyak dan memiliki variasi yang sedikit. Secara keseluruhan algoritma Genetika memiliki nilai respon yang kurang baik dari algoritma Multilevel Feedback Queue.

Kata Kunci : Penjadwalan, Sistem Operasi, Average Response time, Turnaround Time, Normalized Turnaround Time, Algoritma Genetika.

Abstract

Scheduling on Operating System is one of important factor in general. Scheduling Process on Operating System can be described as Process which can allocate in processor that average response time, turnaround time dan normalized turnaround time from system can be minimal. Problem in scheduling categorize in NP-Complete Problem that there is no algorithm can produce optimal output in polynomial. Therefore, in scheduling problem which can be measured the optimal solution can be found. There is a way to find adaptive solution and can be certifiable to solve real problem in scheduling. Genetic Algorithm is a searching technique that can simulate process with natural selection and evolution. Genetic Algorithm work like a natural process from evolution which means forward to optimal solution. Principe Genetic algorithm is to create population from individual that can adaptation in natural environment.

Genetic Algorithm has a better quality process than Multilevel Feedback Queue Algorithm if it works with a little bit process and has not many variant. In general, Genetic algorithm has a not good response for process than Multilevel Feedback Queue algorithm.

Keywords : Scheduling, Operating System, Average Response time, Turnaround Time, Normalized Turnaround Time, Genetic Algorithm.

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Sistem Operasi merupakan jembatan penghubung antara *user* dengan perangkat keras komputer. Dimana, salah satu fungsi Sistem Operasi adalah mengatur waktu eksekusi proses yang akan dilaksanakan oleh sistem dan dapat diatur oleh user atau secara otomatis oleh sistem itu sendiri. Penjadwalan pada Sistem Operasi adalah salah satu faktor pada kriteria sistem yang baik pada umumnya dan faktor yang berpengaruh besar pada eksekusi program. [4]. Permasalahan penjadwalan termasuk dalam permasalahan *NP-Complete* dimana tidak ada algoritma yang dapat menghasilkan hasil yang optimal dalam waktu polinomial. Oleh karena itu, untuk masalah penjadwalan yang dapat diukur dapat ditemukan solusi yang optimal [2]. Salah satu kriteria optimal pada penjadwalan adalah waktu respon yang cepat untuk setiap proses. Terdapat banyak algoritma yang telah diciptakan dan digunakan untuk menangani penjadwalan proses pada sebuah sistem operasi seperti FCFS (*First Come First Served*), RR (*Round Robin*), SJF (*Shortest Job First*) dan *priority scheduling* [4]. Salah satu algoritma yang optimal terhadap waktu respon adalah algoritma Genetika.

Algoritma Genetika mempunyai basis utama yaitu prinsip dari Genetika dan evolusi. Algoritma Genetika dimulai dari ruang pencarian atau populasi dimana terdiri dari semua solusi yang layak untuk masalah optimasi. Sebuah anggota pada populasi disebut sekuens atau *chromosome*. Nilai kecocokan (*fitness value*) akan menjadi elemen dari setiap sekuens yang akan dipilih nilai rata-rata respon terkecil [5]. Diharapkan dengan memilih sekuens yang memiliki waktu respon terkecil, maka performansi akan semakin baik. Performansi algoritma Genetika dapat dibandingkan dengan algoritma yang telah direalisasikan pada komputer, yaitu algoritma *Multilevel Feedback Queue*. Algoritma *Multilevel Feedback Queue* memiliki kelemahan dapat terjadinya *context switch* yang sering jika *quantum* yang diinput terlalu kecil yang dibandingkan dengan nilai *service time* dari proses. Sehingga, algoritma *Multilevel Feedback Queue* memiliki waktu penyelesaian proses yang buruk namun algoritma *Multilevel Feedback Queue* memiliki waktu respon yang baik. Nilai performansi algoritma Genetika yang dibandingkan dengan algoritma *Multilevel Feedback Queue* dapat dilihat dengan berbagai atribut seperti *average response time*, *average*

turnaround time dan *average normalized turnaround time* untuk mengukur performansi, serta menguji beberapa kasus untuk mengukur kinerja algoritma Genetika.

1.2 Perumusan Masalah

Permasalahan yang akan dibahas dalam Tugas Akhir ini antara lain:

1. Bagaimana mengimplementasikan Algoritma Genetika pada penjadwalan proses?
2. Bagaimana performansi Algoritma Genetika pada penjadwalan proses bila dibandingkan dengan algoritma *Multilevel Feedback Queue*?

1.3 Batasan Masalah

Adapun batasan masalah dalam pengerjaan Tugas Akhir ini. Diantaranya adalah sebagai berikut:

1. Tersedianya area untuk alokasi proses dalam CPU, dimana semua proses sudah berada dalam memory.
2. Semua proses independen satu sama lain.
3. Implementasi simulator pada kedua algoritma menggunakan bahasa pemrograman Java.
4. Analisa performansi algoritma ditujukan untuk proses *non real-time*.

1.4 Tujuan

Tujuan dari Tugas Akhir ini adalah:

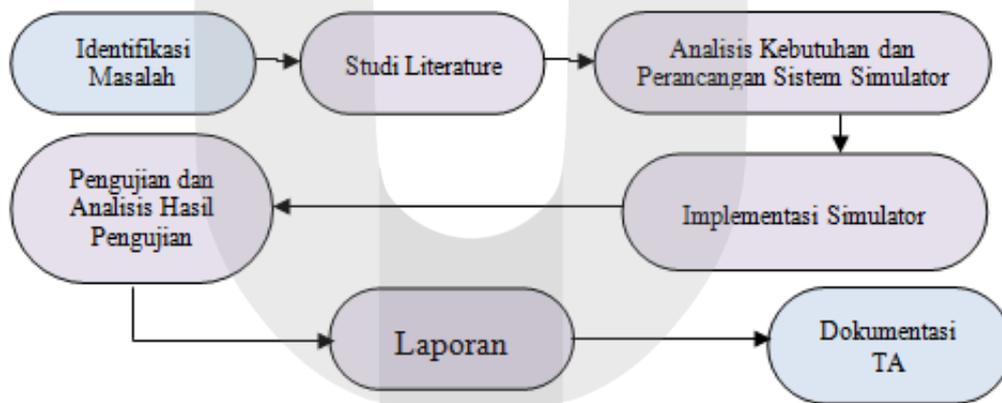
1. Mengimplementasikan algoritma Genetika untuk proses penjadwalan sistem operasi.
2. Menganalisis jumlah proses yang optimal serta variasi ukuran proses yang sesuai untuk algoritma Genetika.
3. Menganalisis performansi algoritma Genetika terhadap algoritma *Multilevel Feedback Queue* berdasarkan *average response time*, *average turnaround time* dan *average normalized turnaround time*.

1.5 Hipotesis

Penggunaan Algoritma Genetika pada proses penjadwalan sistem operasi mempunyai karakteristik yang prediktif terhadap berbagai proses dimana setiap proses akan dikerjakan hingga selesai dan algoritma Genetika mencari sekuens yang terbaik dari *average waiting time*, sehingga Algoritma Genetika dapat mendekati solusi permasalahan pekerjaan secara sekuens yang terbaik untuk mengurangi *response time* dan *turnaround time* dari proses-proses yang ada pada skenario.[4] Algoritma Genetika mencari sekuens proses yang terbaik dari karakteristik *average response time* sekuens yang terkecil dari permasalahan.[1]

1.6 Metodologi Penyelesaian Masalah

Metodologi yang digunakan untuk memecahkan permasalahan-permasalahan yang ada dalam Tugas Akhir ini terdiri dari 6 tahap, yaitu:



Gambar 1.1 Kerangka pengerjaan Tugas Akhir

1. Tahap Identifikasi Masalah
Pada tahap ini akan dilakukan identifikasi terhadap permasalahan yang ada dan melakukan identifikasi sistem yang akan dikembangkan.
2. Tahap Studi Literature
Pada tahap ini akan dipelajari mengenai Algoritma Genetika, *Multilevel Feedback Queue*, *Round Robin* dan *First Come First Serve (FCFS)*.
3. Tahap Analisis kebutuhan dan Perancangan Sistem Simulator
Pada tahap ini akan dilakukan analisis kebutuhan dan merancang sistem dari algoritma yang digunakan.

4. Tahap Implementasi Simulator

Pada tahap ini dilakukan implementasi simulator dari algoritma dengan batasan-batasan yang ada.

5. Tahap Pengujian dan Analisis Hasil Pengujian

Pada tahap ini dilakukan pengujian sistem kemudian analisis terhadap hasil pengujian yang diperoleh.

6. Tahap Laporan

Pembuatan laporan dalam bentuk dokumentasi dari tahap-tahap yang telah dilakukan selama pengerjaan Tugas Akhir berlangsung.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil simulasi Algoritma Genetika maka dapat disimpulkan:

1. Algoritma Genetika dipengaruhi oleh banyaknya jumlah proses, dimana semakin banyak proses yang dikerjakan oleh algoritma Genetika maka hasil performansi algoritma Genetika semakin buruk. Hal ini dapat ditunjukkan dari gambar 4.12, 4.13 dan 4.14 yang menjelaskan pengaruh proses pada nilai *response time*, *turnaround time* dan *normalized turnaround time*.
2. Algoritma Genetika dipengaruhi oleh variasi proses *service time*, dimana variasi yang ideal pada algoritma Genetika adalah 30-50 ms. Hal ini disebabkan pengaruh persebaran data pada lampiran A dimana pada variasi *service time* 30-50 ms memiliki jumlah *service time* yang lebih tersebar dibandingkan dengan kasus variasi *service time* lainnya. Hal ini dapat ditunjukkan dari gambar 4.16 dan 4.17 yang menjelaskan pengaruh proses pada nilai *turnaround time* dan *normalized turnaround time*.
3. Secara penanganan respon proses, algoritma Genetika memiliki nilai respon yang lebih rendah dibandingkan dengan algoritma *Multilevel Feedback Queue* dapat dilihat nilai *response time* dari gambar 4.12 dan 4.15. Hal ini disebabkan perbedaan sifat pada algoritma Genetika yang memiliki sifat non-preemptive dan algoritma *Multilevel Feedback Queue* pada prioritas antrian 1, 2 dan 3 memiliki sifat preemptive dari algoritma Round Robin.
4. Secara kualitas setiap proses, algoritma Genetika lebih baik daripada algoritma *Multilevel Feedback* kecuali pada kasus 80 proses Rincian hasil pengujian dapat dilihat dari parameter *normalized turnaround time* pada gambar 4.14. Hal ini disebabkan waktu eksekusi kasus 80 proses pada algoritma Genetika yang terlampaui besar, sehingga mempengaruhi parameter *response time*. Hal ini dapat dilihat dari parameter *response time* pada gambar 4.12.

5.2 Saran

Beberapa hal yang menjadi saran dari penulis untuk pengembangan tugas akhir ini ke depannya, yaitu

1. Perlu dianalisis jika algoritma Genetika menggunakan nilai bobot berdasarkan parameter *turnaround time*.
2. Perlu dibentuk aturan-aturan yang dinamis untuk parameter internal algoritma Genetika seperti jumlah populasi, probabilitas terjadinya *crossover*, probabilitas mutasi dan jumlah generasi, sehingga algoritma Genetika bekerja efektif yang sesuai dengan lingkungan sistem, seperti input banyaknya generasi yang dihasilkan akan dipengaruhi oleh jumlah proses di sistem pada algoritma Genetika.
3. Perlu dibandingkan nilai performansi algoritma Genetika dengan algoritma yang lainnya, misalkan algoritma Linux Time Sharing.
4. Penggabungan algoritma Genetika dengan algoritma lainnya, misalkan algoritma Round Robin: Genetika dimana algoritma Genetika berfungsi untuk mencari nilai *quantum* yang sesuai dengan lingkungan sistem.

5.

Daftar Pustaka

- [1] Davis, Lawrence. 2008. *Applying Adaptive Algorithms to Epistatic Domains*. America: Texas Instruments Inc.
- [2] Gaba, Vikas and Anshu Prashar. 2012. *Comparison of processor scheduling algorithms using Genetic Approach*. India: International Journal of Advanced Research in Computer Science and Software engineering Volume 2, Issue 8, August 2012
- [3] Hariyanto, Bambang. 2010. *Sistem Operasi*. Informatika, Bandung Indonesia
- [4] Kumar, Rakesh and Rajiv Kumar. 2010. *Genetic Algorithm approach to Operating System process scheduling problem*. India: International Journal of Engineering Science and Technology 2010
- [5] Sindhwani, Preeti and Vaishali Wadhwa. 2011. *Genetic Algorithm approach for Optimal CPU Scheduling*. India: IJCST Vol 2, Issue 2, June 2011
- [6] Stallings, William. 2005. *Operating Systems Internals and Design Principles*. 7th Edition. John Wiley & Sons. Inc, Massachusetts United States of America.
- [7] Suyanto. 2008. *Soft Computing: membangun mesin ber-IQ tinggi*. Informatika, Bandung Indonesia.
- [8] Tanenbaum, Andrew S. 2001. *Modern Operating System 3rd Edition*. Prentice Hall.
- [9] Russinovich, Mark E and David Solomon. 2005. *Microsoft Windows Internal 4th edition*. Microsoft Press. Washington, Amerika.
- [10] Suyanto. 2008. *Evolutionary Computation: Komputasi berbasis Evolusi dan Genetika*. Informatika, Bandung Indonesia.