

## **Penerapan Algoritma Genetika Dalam Memperoleh Average Waiting Time Pada Penjadwalan Round Robin**

**Harni Seven Adinata<sup>1</sup>, Monang Juanda Tua Sihombing<sup>2</sup>**

<sup>1</sup> Teknik Informatika, Universitas Sumatera Utara

<sup>2</sup> Komputerisasi Akuntansi, AMIK Imelda

[seven.adinata@gmail.com](mailto:seven.adinata@gmail.com) , [juanda\\_hs18@yahoo.com](mailto:juanda_hs18@yahoo.com)

### ***Abstract***

*The Average Waiting Time value in round robin scheduling is influenced by the quantum time value given randomly, so that the CPU scheduling is not maximal. In this study, Genetic Algorithms are applied to determine the quantum time value in order to determine the minimum Average Waiting Time value, so that the optimal solution is achieved in solving classic Round Robin scheduling problems. In this study the application of Genetic Algorithms using parameter values consists of population: 2, number of generations: 2. In this study obtained AWT 12 milliseconds with fitness value 0.18388.*

**Keywords:** *Round Robin, Genetic Algorithm, JITA*

### **1. PENDAHULUAN**

Salah satu misi yang dijalankan oleh sistem operasi adalah efisiensi penggunaan waktu ketika terjadi multiprogramming. Sistem operasi dapat menjalankan beberapa program pada saat yang hampir bersamaan atau bersamaan mengakibatkan terjadinya multiprosesing yang membuat CPU harus melaksanakan eksekusi terhadap sejumlah proses [2].

Banyaknya jumlah proses yang harus dieksekusi oleh CPU mengakibatkan kerumitan tersendiri dikarenakan alokasi waktu yang sangat terbatas. Performa dari sebuah sistem operasi sangat tergantung kepada algoritma penjadwalan CPU. Fungsi utama dari algoritma penjadwalan CPU adalah meminimalisasi waktu tunda (waiting time), *response time* dan *context switching*, serta memaksimalkan utilitas CPU [6].

Untuk menangani penjadwalan proses pada sistem operasi terdapat banyak algoritma yang telah diciptakan dan digunakan diantaranya adalah First Come First Served (FCFS), Shortest Job First (SJF) dan

Round Robin (RR). Diantara algoritma tersebut, algoritma yang sering digunakan adalah algoritma round robin karena berbasis waktu quantum, dimana setiap proses akan mendapatkan jumlah waktu proses yang sama [1].

Pada penjadwalan round robin, jumlah *quantum time* (waktu quantum) yang diberikan kepada tiap proses dapat mempengaruhi hasil kinerja penjadwalan round robin. Jika nilai waktu quantum yang diterapkan nilainya lebih besar daripada burst time proses, akan menyebabkan waktu tanggap menjadi lebih lama (besar) dan akan sama persis seperti penjadwalan FCFS, sebaliknya jika nilai waktu quantum yang lebih kecil dari burst time proses akan mengakibatkan banyak *context switching* yang dapat menyebabkan penggunaan CPU tidak efisien [3].

Penentuan nilai waktu quantum yang proporsional dapat dilakukan dengan menyisipkan algoritma genetika di dalam penjadwalan *round robin*. Nilai waktu quantum proporsional dapat menghasilkan waktu tunggu rata - rata yang rendah, sehingga dapat meningkatkan kinerja utlilitas CPU[8].

## 2. METODE

Metode merupakan suatu cara atau teknik yang sistematis untuk memecahkan suatu kasus sehingga memberikan hasil sesuai dengan yang diharapkan. Penulis menggunakan studi kepustakaan (*library research*), yaitu menggunakan sumber-sumber melalui buku, jurnal, tugas akhir, tesis maupun disertasi, *browsing* melalui internet, serta sumber-sumber lain yang relevan untuk digunakan dalam penelitian ini. Studi kepustakaan dalam penelitian ini adalah hal-hal yang berkaitan dengan ilmu Sistem Operasi dan Algoritma Genetika.

### 2.1 Sistem Operasi

Sistem operasi merupakan penghubung perangkat keras (*hardware*) dengan perangkat lunak (*software*). Sistem operasi menggunakan algoritma penjadwalan untuk memberikan layanan kepada pengguna dalam melakukan tugas yang berbeda pada saat terjadi *multi programming*[6]. Untuk mengukur kinerja dari sebuah penjadwalan CPU ada beberapa hal yang perlu kriteria di perhatikan :

1. Adil (*fairness*)

Penjadwalan yang dikategorikan adil apabila semua proses yang mengantri diperlakukan sama, yaitu semua proses yang mengantri dapat di eksekusi oleh CPU.

2. Efisiensi (*eficiency*)

Efisiensi atau utilisasi pemroses dihitung dengan perbandingan (rasio) waktu sibuk CPU sebanyak 100 %, dimana tidak ada jeda CPU dalam mengeksekusi proses yang sedang mengantri.

3. Waktu tanggap (*response time*)

Waktu tanggap yang minimal diperoleh dari nilai *average waiting time* (AWT) pada penjadwalan CPU.

4. Throughput

Banyaknya jumlah proses yang dapat di eksekusi oleh CPU dalam satuan interval waktu.

Algoritma penjadwalan CPU ada enam jenis, yaitu :

1. First Come First Serve (FCFS) scheduling.

Penjadwalan yang memprioritaskan proses yang pertama mengantri untuk pertama kali di eksekusi

2. Short Job First (SJF) scheduling.

Penjadwalan yang memprioritaskan proses yang terakhir mengantri merupakan proses yang pertama kali di eksekusi.

3. Priority Scheduling.

Penjadwalan CPU yang mengeksekusi proses yang memiliki urutan burst terkecil. Penjadwalan priority scheduling akan mengurutkan sejumlah proses yang mengantri pada ready queue terlebih dahulu, sebelum mengeksekusi burst time terkecil.

4. Round Robin Scheduling.

Round robin scheduling merupakan penjadwalan yang memberikan alokasi waktu quantum pada tiap proses dengan nilai burst time yang sama. Penjadwalan round robin akan mengeksekusi proses yang pertama mengantri terlebih dahulu dengan jumlah waktu quantum yang diberikan kemudian dilanjutkan kepada proses berikutnya

5. Multilevel Queue Scheduling

Penjadwalan CPU yang berorientasikan lebih dari satu antrian untuk mengeksekusi kumpulan proses yang akan dieksekusi. Umum nya penjadwalan ini menggunakan dua buah penjadwalan dalam mengeksekusi proses.

6. Multilevel Feedback Queue Scheduling (MFQS)

Penjadwalan MFQS hampir sama seperti penjadwalan Multilevel Quee Scheduling, hanya saja ketika proses memasuki sistem, akan di kategorikan

## 2.2 Round Robin

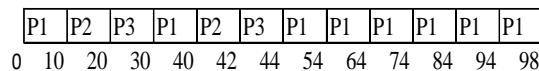
Penjadwalan round robin memiliki persamaan dengan penjadwalan FCFS, hanya saja bersifat *preemptive*, dimana setiap proses yang sedang di eksekusi dapat di tunda, jika waktu yang diberikan CPU sudah habis. Setiap proses pada penjadwlan round robin mendapatkan jumlah waktu yang sama dari CPU, yang disebut dengan waktu quantum (*quantum time*), dimana waktu quantum yang diberikan berkisar antara 10 - 100 milidetik[4].

Performansi algoritma round robin dapat dijelaskan sebagai berikut, jika nilai waktu quantum (q) besar, penjadwalan maka penjadwalan akan tampak sama sperti algoritma penjadwalan FIFO, tetapi jika q kecil maka sering terjadi *context switch*. Sebagai contoh: misalkan waktu quantum yang tentukan adalah 10 mili detik pada proses *P1*, *P2*, dan *P3* dengan varian burst time pada tabel 1

Tabel 1. Antrian Proses dengan waktu quantum 10

Proses	Burst Time
P1	74
P2	12
P3	12

Penjadualan proses dengan algoritma *round robin* dapat dilihat pada gantt chart berikut:

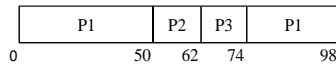


Gambar 1. Ganncart Round Robin quantum time= 10

Waktu tunggu untuk *P1* adalah  $(20 + 4) = 24$ , *P2* adalah  $(10 + 20) = 32$ , dan *P3* adalah  $(20 + 12) = 32$  sehingga rata-rata waktu tunggu adalah  $(24 + 32 + 32) / 3 = 29$  milidetik. Algoritma *round-robin* ini di satu sisi memiliki keuntungan, yaitu adanya keseragaman waktu. Namun di sisi lain, algoritma ini akan terlalu sering melakukan *switching* seperti yang terlihat pada Gambar 1.

Semakin besar waktu quantum pada penjadwalan round robin, maka *switching* yang terjadi akan semakin sedikit. Penjadualan proses dengan

algoritma *round robin* dengan nilai waktu quantum 50 mili detik dapat dilihat pada gambar 2.



Gambar 2. Gancart Round Robin quantum time= 50

Waktu tunggu untuk *P1* adalah  $(0 + (74 - 50) = 24$  , *P2* adalah 50, dan *P3* adalah 62 sehingga rata-rata waktu tunggu adalah  $(24 + 50 + 62) = 45,33$  milidetik. Penggunaan nilai waktu quantum dengan nilai yang besar pada algoritma round robin mengakibatkan nilai AWT semakin besar.

### 2.3 Algoritma Genetika

Algoritma genetika adalah sebuah teknik pencarian yang digunakan dalam komputasi untuk mencari penyelesaian - penyelesaian eksak atau aproksimasi terhadap masalah-masalah optimisasi dan pencarian [5]. Sifat algoritma genetika mencari kemungkinan– kemungkinan dari kandidat solusi untuk mendapatkan yang optimal untuk penyelesaian masalah. Susunan kedatangan proses disimpan sementara pada memori sebelum dieksekusi. Tiap susunan kedatangan proses dapat mempengaruhi nilai *average waiting time* (AWT). Pada penearpan algoritma genetika terhadap penjadwalan round robin adalah untuk mengubah susunan antrian sebelum dilakukan penjadwalan round robin dan menentukan bobot quantum time berdasarkan ketetapan nilai waktu quantum.

Sebelum algoritma genetika dapat dijalankan, maka terlebih dahulu harus merancang sebuah kode yang sesuai untuk permasalahan penjadwalan. Untuk ini, maka tiap solusi yang layak di dalam *search space* di kodekan dalam bentuk kromosom / string yang terdiri atas komponen genetic terkecil yaitu gen. Pembentukan kromosom tampak pada tabel 2.

Tabel 2. Pembentukan Gen

Kromosom	Nilai Gen
1	123
2	312
3	231

Kromosom berevolusi melalui iterasi berkelanjutan, yang disebut generasi. Pada setiap generasi, kromosom dievaluasi berdasarkan suatu

fungsi evaluasi. Pendefinisian kromosom adalah proses mengkodekan nilai gen pada tiap-tiap kromosom menggunakan kromosom permutasi, dimana isi dari satu kromosom mewakili dari sebuah CPU yang akan mengeksekusi sekumpulan proses yang dibentuk akan dijadikan sebagai bagian dari populasi awal yang tampak pada tabel 3.

Tabel 3. Pembentukan Populasi Awal

Populasi	Kromosom	Nilai Gen	Solusi Quantum
1	1	123	10
	2	312	50
	3	231	43

Penghitungan nilai fitness dihitung berdasarkan nilai *Average Waiting Time* terkecil sementara, untuk memperoleh nilai *fitness tiap kromosom*, rumusnya dapat diperoleh sebagai berikut :

$$\text{Fitness (Ki)} = \frac{1}{\text{awt}(q)} \quad (1)$$

Keterangan :

q = Nilai waktu quantum

awt = *Average Waiting Time*

Ki = Nilai *Fitness* Pada Kromosom 1 sampai kromosom n

Proses seleksi kromosom dilakukan dengan cara menyeleksi antara kromosom yang mempunyai fungsi objektif kecil untuk kemungkinan terpilih atau mempunyai nilai probabilitas yang tinggi. Metode seleksi yang digunakan adalah metode *Roulette Wheel Selection* yang terdiri dari.

1. Hitung Total Fitness

$$\text{Total fitness} = \sum Ki \quad (2)$$

Dimana Ki= Jumlah fitness tiap kromosom

2. Hitung fitness relative tiap individu (kromosom)

$$Pk = \frac{Fq}{\text{TotalFitness}} \quad (3)$$

Keterangan :

Pk= Fitness relative tiap – tiap kromosom

Fq= Fitness tiap kromosom

### 3. Hitung fitness kumulatif

$$C[i] = \sum_{k=1}^i P[k] \quad (4)$$

Keterangan:

$P[i]$  = probabilitas *fitness*[ $i$ ]

$C[i]$  = nilai kumulatif indeks ke- $i$

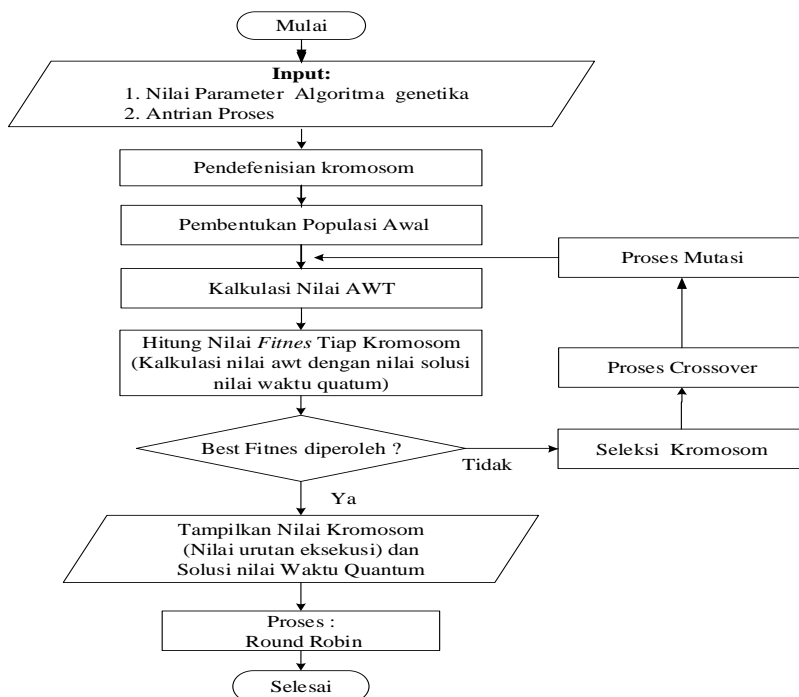
$I$  = indeks kromosom (1,2,3,... $n$ )

$K$  = *counter* (1,2,3,... $n$ )

Proses crossover (kawin silang) adalah prosedur untuk mengkawinkan dua induk yang telah dipilih. Dalam penelitian ini metode kawin silang yang digunakan adalah teknis crossover satu titik. Tujuan dari proses crossover dalam penelitian ini adalah untuk mengetahui kromosom mana dengan urutan proses dan burst time yang memiliki nilai Average Waiting Time yang minimal.

Mutasi kromosom kromosom yang digunakan dalam penelitian ini adalah metode swapping mutaton, dimana mutasi ini dilakukan dengan cara menukarkan nilai gen yang dipilih dari kromosom yang memiliki nilai fitness yang rendah.

Kondisi selesai dicapai jika setelah beberapa generasi/iterasi berturut-turut nilai *fitness* terbaik tidak mengalami perubahan kembali. Secara keseluruhan, diagram alir penerapan Algoritma terhadap penjadualan round robin tampak pada gambar 3.



Gambar 3. Diagram Alir penerapan Algoritma Genetika

### 3 HASIL DAN PEMBAHASAN

Penerpapan algoritma genetika terhadap penjadualan round robin dengan menggunakan sebanyak 3 populasi terlihat pada tabel 4.

Tabel 4. Pembentukan 3 buah populasi

Populasi	Kromosom	Nilai Gen	Solusi Quantum	AWT	Total AWT
1	1	123	10	30.667	36.88888889
	2	312	50	28.667	
	3	231	40	51.333	
2	1	123	30	24	20.33333333
	2	312	45	23	
	3	231	18	14	
3	1	312	16	22.667	17.44444444
	2	231	17	17.667	
	3	123	40	12	

Perolehan total average waiting time terkecil secara keseluruhan terdapat pada populasi ketiga sebesar 17,44 milidetik. Nilai average waiting time terkecil yang ada pada populasi ketiga terletak di kromosom tiga sebesar 12 milidetik dengan urutan proses nya adalah P1-P2-P3 dengan nilai quantum time sebesar 40.

Pada regenerasi yang dilakukan menggunakan nilai generasi: 2 generasi, maka diperoleh nilai fintnes yang tampak pada tabel 5.

Tabel 5. Proses regenaerasi terhadap 3 populasi

Generasi	Populasi	Nilai Fitnes	Total Fitnes
1	1	0.15082575	0.49145
	2	0.156573499	
	3	0.184054754	
2	1	0.042857143	0.18388
	2	0.057692308	
	3	0.083333333	



Perolehan nilai *fitness* yang digunakan adalah nilai *fitness* terkecil dengan tujuan semakin nilai *fitness*, maka nilai average waiting time semakin kecil. Total *fitness* terkecil diperoleh dari regenerasi kedua dengan nilai *fitness* 0.18388. Adapun susunan dan nilai waktu kuantum yang diperoleh pada tabel 6.

Tabel 6. Pembentukan populasi pada generasi kedua.

Generasi	Populasi	Kromosom	Nilai Gen	Solusi Quantum	AWT	Total AWT
2	1	1	123	17	23.333	
		2	312	16	17.333	17.55555556
		3	231	40	12	
	2	1	123	17	23.333	
		2	312	18	23	20.11111111
		3	231	16	14	
	3	1	312	16	22.667	
		2	231	17	17.667	17.44444444
		3	123	40	12	

#### 4 SIMPULAN

1. Algoritma Genetika dapat diterapkan untuk memberikan solusi berupa susunan proses dan pemberian nilai waktu quantum dalam memperoleh *average waiting time* minimum.
2. Besarnya parameter jumlah generasi yang diterapkan memiliki pengaruh yang signifikan terhadap pencapaian solusi akhir.

## 5 REFERENSI

- [1] Ahad, M. A. 2012. *Modifying round robin algorithm for process scheduling using dynamic quantum precision*. International Journal of Computer Applications on Issues and Challenges in Networking, Intelligence and Computing Technologies (ICNICT), pp. 5 -10.
- [2] D'Souza, M., Caiero, F., dan Surlakar, S. 2012. *Optimal round robin CPU scheduling algorithm using euclidean distance*. Journal of Computer Applications Vol. 96(18): 8-11
- [3] Kundargi, N. dan Bandekar, S.2013. *CPU scheduling algorithm using dynamic time quantum for batch system*. Internatioan Journal of latest Trends in Engineering and Technology. 82-84
- [4] Mishra, M. K., dan Rashid. F, 2014. *An improved round robin CPU scheduling algorithm with varying time quantum*. International Journal of computer science, Engineering and Applications. Vol. 4(4):1-8
- [5] Kumar, R. 2012. *Novel encoding scheme in genetic algorithms for better fitness*. International Journal of Engineering and Advanced Technology (IJEAT). Vol.1(6) : 2249 – 8958.
- [6] Silberschatz, A., Galvin. B. G., dan Gagn. G. 2012, *Operating Systems concepts*, John Wiley and Sons. 9<sup>th</sup> Edition. Yale Univeristy.