

## EFISIENSI WAKTU PROSES ALGORITMA NON PREEMPTIVE : HIGHEST RATIO NEXT DAN FIRST IN FIRST OUT

Sripurwani Hariningsih<sup>1</sup>, Erna Zuni Astuti<sup>2</sup>

<sup>1,2</sup>Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Jl. Nakula 5 - 11, Semarang, 50131, 024-3517261

E-mail : sp.hariningsih@dsn.dinus.ac.id<sup>1</sup>, erna.zuni.astuti@dsn.dinus.ac.id<sup>2</sup>

### **Abstrak**

Pengembangan ilmu dan teknologi menghasilkan terciptanya media kerja manusia yang sangat canggih dalam peradaban saat ini. Salah satu media kerja manusia yang paling utama adalah sistem komputer. Kecanggihan komputer akan diikuti oleh perkembangan sistem operasinya. Bahkan dari kecanggihan sistem operasi mempengaruhi kinerja layanan komputer terhadap pemakainya. Tidak disadari pula bagaimana pseudoparalellisme komputer dapat terjadi. Dalam pemakaian sistem komputer antar user, prosesor telah terjadi penjadwalan sistem. Algoritma maupun teknik penjadwalan proses ini berfungsi dengan tujuan antar job yang dalam status proses scheduler tidak akan terjadi crossing (tabrakan) maupun terjadinya deadlock. Hal ini juga proses-proses dituntut untuk bersifat adil terhadap setiap job dalam antrian proses, bersifat interleaved antar proses sehingga waktu tunda yang tersisa-sisa dapat diminimalkan. Teknik penjadwalan proses FIFO merupakan teknik penjadwalan yang dikenal dengan kesederhanaannya jika di banding teknik pejadwalan proses yang lainnya. Tetapi dilihat dari hasil pengujian dengan PP=0 untuk Teknik penjadwalan FIFO : HRN = 0 : 0; tetapi jika diumpamakan nilai PP=2 maka diperoleh perbandingan antara teknik penjadwalan antara FIFO : HRN adalah 7.6 : 8.4. Hal ini menunjukkan bahwa teknik yang paling sederhana belum tentu mempunyai nilai efisien yang tinggi. Hal ini akan mendukung bahwa nilai throughput dari HRN lebih besar pula.

**Kata Kunci :** Non Preemptive, FIFO, HRN, Throughput.

### **Abstract**

Development of science and technology resulted in the creation of advanced media for human activities nowadays. One of the media is the computer system. The sophistication of computer will be followed by the development of the operating system. Even from the sophistication of the operating system affects the performance of computer services to the user. Not realizing how computer pseudo parallel can also occur. In the use of computer between users, processor scheduling system has occurred. This algorithm and process scheduler status has the function so that there are no crossing and deadlock between jobs in scheduler process status. Processes are required to be fair to every job in the queue process, interleaved between processes so the wasted delay time can be minimized. FIFO scheduling technique is a process scheduling techniques known with its simplicity when compared with other scheduler. But when seen from the test results with PP = 0 for FIFO scheduling technique : HRN = 0 : 0 ; but if they are portrayed PP = 2 then the value for the comparison between the FIFO scheduling techniques : HRN is 7.6 : 8.4 . This suggests that the simplest techniques have not necessarily high efficiency. This would support that the throughput of HRN could be greater anyway.

**Keywords :** Non Preemptive, FIFO, HRN, Throughput.

## 1. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi saat ini sangat pesat sekali terutama teknologi komputer. Manusia yang pada umumnya sebagai pemakai komputer bersifat pasif dalam artian hanya “mampu” memakai fasilitas yang telah tersedia tanpa mau berfikir bagaimana komputer itu bisa digunakan untuk menyelesaikan masalah, bagaimana komputer dapat mengakses banyak masalah secara bersamaan. Pada dasarnya komputer hanya sebuah barang rongsokan apabila tidak tersedia *operating system* (SO). SO ini sebenarnya selalu dilakukan pengembangan seiring dengan perkembangan komputer.

Yang dikatakan dalam single processing adalah jaminan bahwa prosesor hanya mampu melakukan satu eksekusi saja dalam kurun waktu tertentu. Untuk era globalisasi seperti sekarang ini proses yang bersifat tunggal merupakan suatu masalah yang benar-benar harus dihindarkan. Bisa dibayangkan bagaimana dan berapa lama suatu pekerjaan dengan banyak antrian akan dapat di selesaikan? Masalah inilah manusia yang posisinya sebagai user tidak pernah dipikirkan, bagaimana komputer dapat bekerja secara bersamaan dengan berbagai macam masalah [1].

Sistem operasi dalam sistem komputer mempunyai tugas pokok sebagai pengendali sumber daya sistem komputer, terutama sebagai pengendali proses. Baik proses yang mempunyai status akan, telah, maupun sedang dieksekusi. Sistem operasi mengalokasikan sumber daya maupun mendealokasikan sumber daya. Sesuai dengan metode yang dipakai untuk

menentukan waktu eksekusi, selesai eksekusi dan nilai *turn around time*-nya. Untuk itu diperlukan algoritma dan teknik penjadwalan proses [1].

Terdapat delapan teknik penjadwalan proses yang memiliki keunggulan sendiri-sendiri. Dalam kesempatan ini akan dibahas penjadwalan proses algoritma non pre-emptive dengan teknik penjadwalan HRN dan FIFO [2].

Tujuan penulisan ini adalah untuk membahas dan menganalisis lebih rinci tentang teknik penjadwalan FIFO dan HRN dan perancangan simulasi untuk menggambarkan kinerja teknik penjadwalan proses tersebut, Walaupun kedua teknik ini termasuk kedalam algoritma teknik penjadwalan *non pre emptive*.

Sebelum membahas dengan membandingkan dua teknik penjadwalan proses diatas, ada baiknya kita bahas terlebih dahulu sifat dari proses-proses. Antara lain [3][4][5] :

### a. Fairness

Prosesor mempunyai sifat adil terhadap job-job dalam antrian proses. Dalam artian semua job yang terdapat dalam antrian proses akan mendapatkan jatah waktu eksekusi secara adil (sesuai dengan jatah waktu eksekusi). Semua job dalam antrian proses akan diberlakukan sama oleh processor. Tidak ada proses yang mengalami starvation maupun deadlock.

### b. Efisiensi

Yaitu proses membandingkan rasio waktu sibuk processor. Karena tujuan dari teknik penjadwalan proses adalah untuk menjaga supaya processor selalu dalam keadaan sibuk, selalu dalam

status proses, sehingga efisiensi waktu proses dapat tercapai secara maksimal.

#### c. Response time

Waktu yang digunakan untuk menunggu job dalam antrian proses sampai job tersebut mulai dilakukan eksekusi (proses). Semakin besar nilai RTnya maka akan mempengaruhi job yang berada di belakangnya menunggu dalam waktu yang lama. Begitu juga akan mempengaruhi nilai *turn around time*-nya. Sasaran dari teknik penjadwalan proses adalah meminimalkan waktu tunggu/waktu tanggap terhadap proses.

#### d. Turn Around Time

Besarnya waktu eksekusi dengan waktu tunggu, maksudnya besarnya waktu yang dihabiskan dari saat job mulai masuk ke sistem (*ready*) sampai proses tersebut selesai untuk di eksekusi (*completion*). Sasaran dari teknik penjadwalan proses adalah untuk meminimalkan *turn around time*.

#### e. Throughput

Banyaknya job yang dapat dilakukan eksekusi dengan kurun waktu tertentu. Semakin banyak waktu *throughput* yang tersedia maka semakin banyak pula job yang dapat dieksekusi.

Penjadwalan proses dalam sistem operasi terdapat 2 algoritma penjadwalan proses, antara lain :

1. Algoritma penjadwalan proses *pre-emptive*,
2. Algoritma penjadwalan proses *Non-Preemptive*

Keterangan :

1. Algoritma penjadwalan proses *pre-emptive*, yaitu Job yang sedang dilakukan proses atau dalam status

proses dan job-job tersebut dapat di sela, diberhentikan ataupun disisipi oleh proses lain walaupun job dalam processor tersebut belum selesai di eksekusi.

2. Algoritma penjadwalan proses *Non-Preemptive*, yaitu job yang sedang dilakukan proses atau dalam status proses dan job-job tersebut tidak dapat di sela, diberhentikan ataupun disisipi oleh proses lain walaupun job dalam prosesor tersebut belum selesai dieksekusi. Job-job dalam antrian proses Algoritma penjadwalan non-pre-emptive terdapat proses antrian yang panjang bagi job yang mempunyai lama eksekusi terdahulu besar.

## 2. METODE

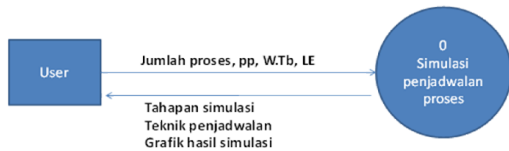
Dalam teknik penjadwalan proses HRN maupun FIFO ini sama-sama menggunakan algoritma penjadwalan proses non pre-emptive. Yaitu job yang sedang dilakukan eksekusi tidak dapat di sela, disisipi ataupun diberhentikan oleh proses lain [5]. Jadi algoritma ini menerapkan teknik satu proses akan tetap berada didalam dalam status process scheduler sampai proses tersebut selesai dieksekusi. Dan waktu selesainya eksekusi baru bisa digunakan oleh proses lain secara *interleave* dan seterusnya sampai job dalam antrian proses selesai dieksekusi semuanya.

Untuk menggambarkan proses analisis ini kami menggunakan alat bantu yang berupa *data flow diagram* (DFD), dengan tahapan pengembangannya adalah :

- a. Pembuatan Diagram Konteks

- b. Penurunan proses menjadi beberapa tingkatan yaitu DFD Level.

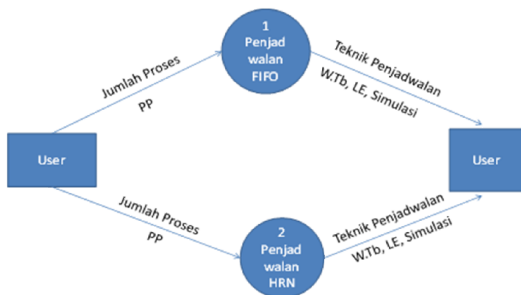
**2.1 Contex Diagram**



**Gambar 1.** Diagram Konteks Simulasi Penjadwalan Proses

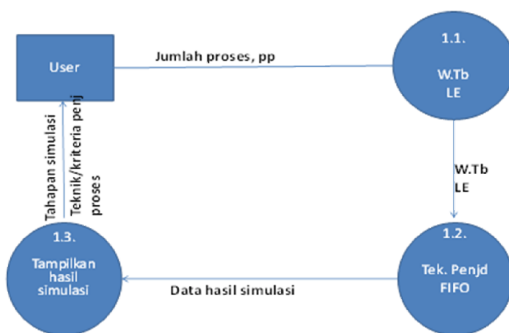
**2.2 DFD Level**

DFD level 1 simulasi penjadwalan proses digambarkan pada Gambar 2.



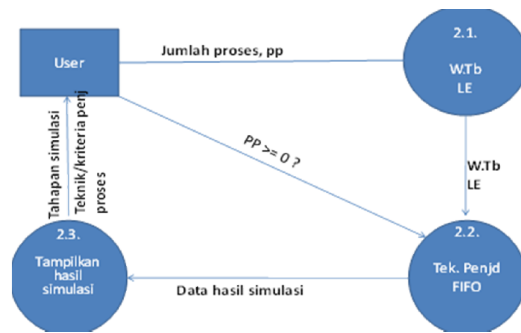
**Gambar 2.** DFD Level 1 Simulasi Penjadwalan Proses

DFD level 2 penjadwalan FIFO digambarkan pada Gambar 3 berikut.



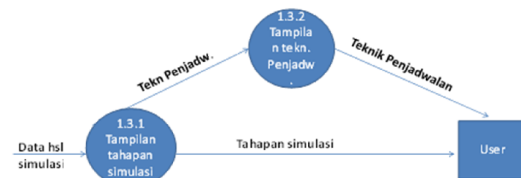
**Gambar 3.** DFD Level 2 Simulasi Penjadwalan Proses FIFO

DFD level 2 penjadwalan HRN digambarkan pada Gambar 4.



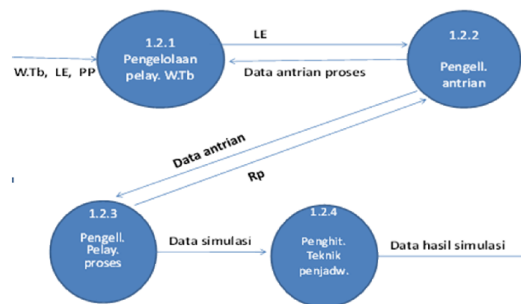
**Gambar 4.** DFD Level 2 Simulasi Penjadwalan Proses HRN

DFD level 3 proses tampilan hasil simulasi digambarkan pada Gambar 5 berikut.



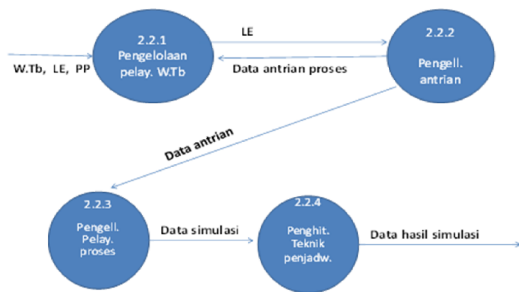
**Gambar 5.** DFD Level 3 Proses Tampilan Hasil Simulasi

DFD Level 3 Proses Pengelolaan Penjadwalan HRN digambarkan pada Gambar 6 berikut.



**Gambar 6.** DFD Level 3 Proses Pengelolaan Penjadwalan HRN

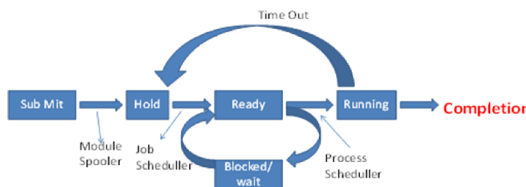
DFD Level 3 Proses Pengelolaan Penjadwalan FIFO digambarkan pada Gambar 7 berikut.



Gambar 7. DFD Level 3 Proses Pengelolaan Penjadwalan FIFO

### 3. HASIL DAN PEMBAHASAN

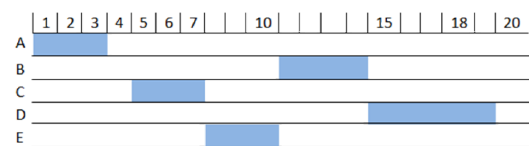
Dikatakan proses apabila terdapatnya job yang sedang di eksekusi. Atau job yang berada dalam status *ready* – *running* (*process scheduler*), maka jika digambarkan siklus dari proses adalah sebagai berikut :



Gambar 8. Siklus proses sistem komputer

Algoritma penjadwalan proses *non pre emptive* dengan teknik penjadwalan HRN mempunyai metode penyelesaian yang lebih rumit jika di dibandingkan dengan metode lainnya. Saya katakan *highest ratio next* lebih rumit jika dibandingkan dengan metode penjadwalan proses lainnya karena terdapatnya rumus yang bertingkat di dalam teknik penjadwalan tersebut, yaitu sebelum melakukan eksekusi harus dilihat terlebih dahulu status dan nilai prosesor, yang kedua dengan

mencari terlebih dahulu nilai *ratio penalty* dengan syarat proses jika job mempunyai nilai *ratio penalty* tertinggi maka job tersebut yang akan dieksekusi terlebih dahulu. Jadi harus membandingkan nilai *ratio penalty* diantara job-job antrian proses. Dengan rumusan *ratio penalty*  $(R_p) = (s+t)/t$ . Begitu seterusnya dilakukan penghitungan dan perbandingan *ratio penalty* sampai job dalam antrian proses berakhir. Dalam teknik penjadwalan *highest ratio next* dapat terjadi proses-proses dilakukan tidak secara berurutan sesuai kedatangan tetapi tergantung dari proses perbandingan nilai *ratio penalty* dari tiap antrian proses yang ada.



Gambar 9. Diagram Teknik Penjadwalan HRN

Contoh jika diketahui di Tabel 1 berikut:

Tabel 1: Tabel Antrian Proses

Antrian Proses	A	B	C	D	E
Waktu tiba	1	2	4	4	5
Lama eksekusi	2	4	3	5	3

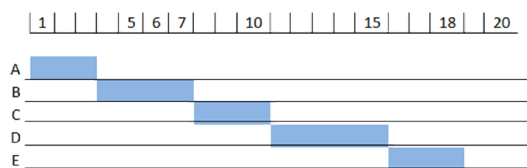
Hitunglah nilai rerata TA dengan menggunakan teknik penjadwalan proses HRN dengan  $PP = 0$

Tabel 2: Detail Antrian Proses

Antri an Proses	Wa ktu Tiba	Lama Eksek usi	Mulai Eksek usi	Selesa i Eksek usi	Wak tu Tung gu	T A
A	1	2	1	3	0	2
B	2	4	10	14	8	12
C	4	3	4	7	0	3
D	4	5	14	19	10	15
E	5	3	7	10	2	5

Rerata TA =  $37/5 = 7.4$

Pada teknik penjadwalan FIFO merupakan teknik penjadwalan yang sangat sederhana, bisa dibilang paling sederhana jika dibandingkan dengan teknik penjadwalan lainnya. Yaitu job yang pertama kali tiba dalam antrian proses maka job tersebut yang akan di eksekusi terlebih dahulu. Jadi job yang berada dalam antrian terakhir akan diproses oleh processor paling akhir juga.



Gambar 10. Diagram Teknik Penjadwalan FIFO

Contoh jika diketahui dari Tabel 3 berikut.

Tabel 3: Daftar Antrian Proses

Antrian Proses	A	B	C	D	E
Waktu tiba	1	2	4	4	5
Lama eksekusi	2	4	3	5	3

Hitunglah nilai rerata TA dengan menggunakan teknik penjadwalan proses FIFO dengan PP = 0

Tabel 4: Hasil Penjadwalan

Antrian Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	Waktu Tunggu	TA
A	1	2	1	3	0	2
B	2	4	3	7	1	5
C	4	3	7	10	3	6
D	4	5	10	15	6	11
E	5	3	15	18	10	15

Rerata TA = 37/5 = 7.4

Bagaimana dengan penjadwalan tersebut jika PP > 0

Tabel 4: Daftar Antrian Proses

Antrian Proses	A	B	C	D	E
Waktu tiba	1	2	4	4	5
Lama eksekusi	2	4	3	5	3

Hitunglah nilai rerata TA.....? dengan menggunakan teknik penjadwalan proses HRN dengan PP = 2.

Tabel 5: Hasil Penjadwalan

Antrian Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	Waktu Tunggu	TA
A	1	2	2	4	1	3
B	2	4	10	14	8	12
C	4	3	4	7	0	3
D	4	5	14	19	10	15
E	5	3	7	10	2	5

Rerata TA = 38/5 = 7.6

Dengan teknik FIFO maka diperoleh hasilnya :

Tabel 6: Hasil Penjadwalan

Antrian Proses	Waktu Tiba	Lama Eksekusi	Mulai Eksekusi	Selesai Eksekusi	Waktu Tunggu	TA
A	1	2	2	4	1	3
B	2	4	4	8	2	6
C	4	3	8	11	4	7
D	4	5	11	16	7	11
E	5	3	16	19	11	16

Rerata TA = 42 = 8.4

#### 4. KESIMPULAN

Hasil pengujian dengan simulasi DFD tersebut diatas mampu memberikan informasi teknik/kriteria penjadwalan proses. Pada penjadwalan proses teknik HRN jika PP = 0 maka mempunyai nilai rerata TA yang mendekati teknik penjadwalan proses FIFO. Tetapi jika PP > 0 misalnya 2 saja maka akan mempengaruhi nilai rerata TA yaitu 7.8 : 8.4. Hal ini menunjukkan bahwa teknik yang paling sederhana belum

tentu mempunyai nilai efisien yang tinggi. Hal ini akan mendukung bahwa nilai *throughput* dari HRN lebih besar pula. *Response time* pada teknik penjadwalan FIFO cenderung lebih besar jika dibanding dengan teknik penjadwalan HRN.

#### DAFTAR PUSTAKA

- [1] H M Deitek, 1990, *Operating Systems*, Massachusetts, Addison-Wesley, 2nd ed.
- [2] B Hariyanto, 1997, *Sistem Operasi*, Informatika, Bandung.
- [3] Avi Silberschatz, Peter Galvin, dan Grag Gagne, 2000, *Applied Operating Systems: First Edition*, Edisi Pertama, John Wiley & Sons.
- [4] William Stallings, 2001, *Operating Systems*, Fourth Edition, Prentice Hall.
- [5] Sri Kusuma Dewi, *Sistem Operasi*. Yogyakarta
- [6] Andrew S Tanenbaum, 1992, *Modern Operating Systems*, Englewood Cliffs, New Jersey.