

## BAB 6

### KESIMPULAN DAN SARAN

Pada bab ini akan dibahas mengenai kesimpulan dari hasil analisis, perancangan, dan pengujian perangkat lunak simulator penjadwalan *flow shop*.

#### 6.1 Kesimpulan

Kesimpulan dari penulisan skripsi ini adalah algoritma penjadwalan *flow shop* untuk meminimalisir nilai *make span* dengan metode NEH, Palmer, dan Gupta telah berhasil diterapkan dan diimplementasikan pada perangkat lunak simulator penjadwalan *flow shop*. Perangkat lunak simulator penjadwalan *flow shop* juga mampu melakukan simulasi pemrosesan urutan antrian *job* hasil penjadwalan dengan algoritma NEH, Palmer dan Gupta. Perangkat lunak simulator penjadwalan *flow shop* juga mampu memperlihatkan urutan antrian hasil penjadwalan beserta dengan nilai *make span* dan *tardiness* dari urutan antrian hasil penjadwalan dengan algoritma NEH, Palmer, dan Gupta. Analisis mengenai hasil penjadwalan *flow shop* dari algoritma NEH, Palmer, dan Gupta dilakukan dengan cara membandingkan nilai hasil *make span* dari masing-masing algoritma. Dari hasil pengujian perangkat lunak, dapat disimpulkan bahwa algoritma penjadwalan NEH merupakan algoritma penjadwalan yang menghasilkan nilai *make span* terbaik untuk setiap pengujian dengan data yang berbeda. Sedangkan algoritma Palmer dan Gupta menghasilkan nilai *make span* yang bervariasi. Dalam beberapa kasus algoritma Palmer menghasilkan nilai *make span* yang lebih baik ketimbang algoritma Gupta. Sebaliknya, di kasus lain algoritma Gupta menghasilkan nilai *make span* yang lebih baik dibandingkan algoritma Palmer. Variasi dari nilai *make span* pada algoritma Palmer dan Gupta dikarenakan kedua algoritma tersebut memiliki metode menghitung **slope index** yang cukup berbeda sehingga menghasilkan urutan penjadwalan yang berbeda pula.

Hasil analisis terhadap perangkat lunak simulator penjadwalan *flow shop* yang sudah dibangun antara lain :

1. Perangkat lunak simulator penjadwalan *flow shop* dapat melakukan simulasi proses *job* hasil dari penjadwalan dengan algoritma NEH, Palmer, Gupta dengan berbagai macam *input* dalam bentuk *file* maupun dengan data yang dibangkitkan secara *random*. Sementara untuk proses simulasi masih belum sempurna karena dibatasi oleh ukuran antarmuka yang belum dapat menampilkan simulasi dengan skala data besar dengan baik.
2. Secara teori, perangkat lunak simulator penjadwalan *flow shop* dapat melakukan penjadwalan dengan data tanpa batasan jumlah *job* maupun jumlah *machine*, namun jumlah *job* dan *machine* dibatasi hingga 200 *job* dan 20 *machine* untuk mendapatkan tampilan simulasi yang optimal.
3. Penjadwalan dengan algoritma NEH masih belum optimal karena durasi untuk penjadwalan *flow shop* dengan algoritma NEH menghabiskan waktu cukup lama dibandingkan dengan algoritma lain untuk data dengan skala cukup besar.

## 6.2 Saran

Adapun saran untuk pengembangan dari perangkat lunak simulator penjadwalan *flow shop*, adalah sebagai berikut :

1. Membuat rancangan antarmuka yang dapat menampilkan simulasi dengan baik untuk data skala besar. Hal ini dikarenakan dalam pembuatan perangkat lunak simulator penjadwalan *flow shop* jumlah *job* dan jumlah *machine* dibatasi agar dapat menampilkan simulasi secara baik.
2. Mengoptimalkan algoritma penjadwalan agar durasi proses penjadwalan lebih optimal. Pengembangan ini dibutuhkan didasari oleh proses penjadwalan yang menggunakan waktu cukup lama untuk melakukan penjadwalan dengan data skala besar. Sehingga diperlukan optimasi dari algoritma penjadwalan *flow shop* yang digunakan pada perangkat lunak (NEH, Palmer, dan Gupta) agar dapat melakukan penjadwalan dengan waktu yang lebih singkat.
3. Melengkapi parameter hasil simulasi dengan parameter-parameter lain seperti, tingkat optimasi *machine*, rata-rata waktu *idle* dari setiap *machine*, rata-rata waktu tunggu setiap *job*, dan lain-lain.
4. Menambah fitur untuk menyimpan data hasil simulasi ke dalam basis data. Fitur ini diperlukan karena nilai hasil simulasi untuk tampilan statistik masih disimpan secara *hard code*.

## DAFTAR REFERENSI

- [1] Baker, K. (1974) *Introduction to sequencing and scheduling*. Wiley, Canada.
- [2] Parveen, S. dan Ullah, H. (2010) Review on job-shop and flow-shop scheduling using multi criteria decision making. *Journal of Mechanical Engineering*, **ME 41, No.2 (2010)**, 130–146.
- [3] Masudin, I., Marsetya Utama, D., dan Susastro, F. (2014) Penjadwalan flow shop menggunakan algoritma nawaz enscore ham. *Jurnal Ilmiah Teknik Industri*, **Vol. 13, No. 1, Juni 2014**.
- [4] Hong, T.-P., Huang, P.-Y., dan Horng, G. (2006) Using the lpt and the palmer approaches to solve group flexible flow-shop problems. *IJCSNS International Journal of Computer Science and Network Security*, **Vol 6, No.3A**, 98–104.
- [5] Sugino, W. dan Abdullah, H. (2006) Penjadwalan produksi menggunakan metode fcfs, cds, dan gupta. *Jurnal Ilmu Komputer dan Sistem Informasi*, **Vol 1, No 1 (2013)**, 225–230.
- [6] Kuswandi, I. (2010) Minimasi makespan dengan penjadwalan produksi pada tipe berulang. *Jurnal Teknik Industri*, **Vol 11, No.1**, 84–93.
- [7] Juniar, A. (2015) Penerapan algoritma greedy pada penjadwalan produksi single-stage dengan parallel machine di industri konveksi. *Jurnal SIFO Mikroskil*, **Vol 16, No 2 (2015)**, 175–184.
- [8] G.Srinivasan, P. (2012) Flow shop scheduling heuristics palmer, campbell dudek smith algorithm. Operations and Supply Chain Management Department of Management Studies Indian Institute of Technology, Madras, India.
- [9] Leemis, L. dan Park, S. (2006) *Discrete-event Simulation: A First Course*, 3rd edition. Pearson Prentice Hall, Canada.