Performa (2002) Vol. 1, No.1: 20-25

Model Penjadwalan Pekerjaan pada *Flowshop* dengan Kriteria Minimasi Total Waktu Tinggal Aktual

Yuniaristanto*
Jurusan Teknik Industri, Universitas Sebelas Maret, Surakarta

Abstract
This paper proposes a scheduling model to determine a sequence of the jobs processed in flowshop. The objective for the model is to minimize total actual flow time accomodating the condition that jobs to be processed arrive at the shop at any times when needed and the conditon that the completed jobs be delivered at a common due-date. A heuristic algorithm searching the best solution from the resulting feasible solution has been developed. The numerical experience showing the characteristic of the problem is also presented.

***Keywords** : scheduling, heuristic, actual flowtime.*

1. Pendahuluan

Literatur-literatur yang membahas penjadwalan pekerjaan (*job*) pada sistem produksi *flowshop*, antara lain penelitian Chan dan Bedworth (1990), Rajendran dan Chaudhuri (1991) dan Woo dan Yin (1998). Penelitian-penelitian ini masih mengasumsikan bahwa seluruh pekerjaan telah siap pada saat t=0 dan belum mempertimbangkan *due-date* sebagai pembatas. Pada kenyataannya, pekerjaan tidak perlu datang seluruhnya pada saat t=0 tetapi dapat datang pada saat pekerjaan tersebut dibutuhkan. Demikian juga, pengiriman pekerjaan ke konsumen tidak dapat dilakukan setiap saat tetapi harus pada saat *due-date*.

Halim dan Ohta (1993, 1994) memperbaiki kekurangan kriteria waktu tinggal (*flow time*) dengan mengusulkan kriteria waktu tinggal aktual (*actual flow time*), yaitu lamanya pekerjaan berada di dalam pabrik dari saat mulai proses sampai *due-date*. Kriteria baru ini digunakan untuk model yang diusulkan agar dapat mengakomodasikan kondisi yang lebih realistik dan konsep tepat waktu, baik tepat waktu kedatangan material maupun tepat waktu pengiriman.

Memperhatikan uraian di atas, perlu dikembangkan suatu model penjadwalan pekerjaan pada *flowshop* statis m mesin untuk kasus *common due-date* dengan kriteria minimasi total waktu tinggal aktual. Asumsi-asumsi yang digunakan adalah :

1. Setiap *stage* diasumsikan hanya terdiri dari satu mesin.
2. Waktu proses pekerjaan pada mesin diketahui dan tetap.
3. Waktu *setup* mesin diketahui dan terpisah dari waktu proses.
4. Sistem yang diteliti bersifat deterministik.
5. Pemrosesan pekerjaan tidak dapat diganggu (interupsi).
6. Mesin selalu tersedia.

* *Correspondence*: E-mail: utan@uns.ac.id

7. Tidak ada mesin yang memproses lebih dari satu operasi pada saat sama.
8. Tidak ada pekerjaan yang diproses oleh lebih dari satu mesin pada saat sama.

2. Penjadwalan Pekerjaan dengan Waktu Tinggal Aktual

Halim (1993) mendefinisikan bahwa waktu tinggal aktual sebuah pekerjaan adalah lamanya pekerjaan tersebut berada di dalam pabrik mulai dari saat mulai proses sampai *due-date* pekerjaan tersebut dan dapat dirumuskan sebagai berikut :

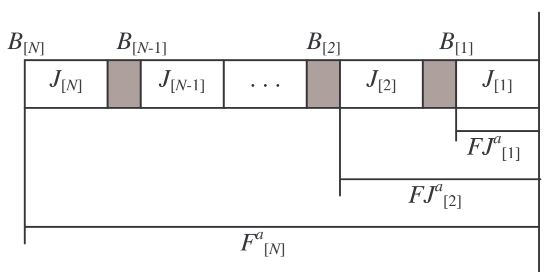
$$F^a_i = d - B_i \;\text{ untuk } i = 1, \ldots, n \qquad (1)$$

Dengan F^a_i, d, B_i adalah waktu tinggal aktual, *common due-date* dan saat mulai proses (*starting time*) dari pekerjaan yang dijadwalkan pada urutan ke-i dihitung dari posisi awal horison waktu. Definisi dari persamaan (2) adalah pekerjaan yang terlambat tidak diijinkan dan definisi ini memanfaatkan model penjadwalan mundur.

Dengan mengasumsikan bahwa waktu *setup* (s) konstan dan terpisah dari waktu proses pekerjaan (p_j), maka persamaan (1) dapat ditulis kembali sebagai berikut :

$$F_i^a = \sum_{j=1}^{i} \left(p_j + s\right) - s\,,\;\text{ untuk } i = 1, \ldots, n \qquad (2)$$

Perlu dicatat bahwa indeks i pada persamaan (2) dihitung dari posisi akhir pada horison waktu dengan penjadwalan mundur. Kondisi ini diperlihatkan pada gambar 1.



Gambar 1. *Gantt chart* penjadwalan mundur

3. Pengembangan Model

Pada model Chan dan Bedworth (1990) beberapa hal yang belum dipertimbangkan adalah waktu *setup* yang terpisah dari waktu proses, kendala *due-date, ready time* (r_i) ≠ 0. Model ini menggambarkan kondisi penjadwalan pekerjaan pada *flowshop* yang meminimasi total waktu tinggal aktual. Formulasi matematis model penjadwalan pekerjaan pada *common-due-date-flowshop* (*JCF*) dapat dirumuskan sebagai berikut :

$$\text{Minimasi} \quad F^a = \sum_{i=1}^{n} \left(d - B_{[i,1]}\right) \qquad (3)$$

Pembatas :

$$B_{[n,1]} \geq 0, \qquad (4)$$

$$B_{[i,m]} = B_{[i-1,m]} - t_{[i,m]} - s_{[i-1,m]},\;\; i = 2, \ldots, n, \qquad (5)$$

$$B_{[i,k]} = \min\{B_{[i-1,k]} - t_{[i,k]} - s_{[i-1,k]},\, B_{[i,k+1]} - t_{[i,k]}\},$$
$$i = 2, \ldots, n;\;\; k = 1, \ldots, m-1, \qquad (6)$$

$$B_{[1,k]} = B_{[1,k+1]} - t_{[1,k]}\,,\;\; k = 1, \ldots, m-1, \qquad (7)$$

$$B_{[1,m]} = d - t_{[1,m]}, \qquad (8)$$

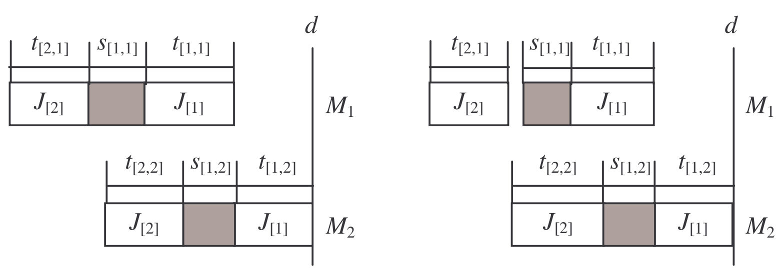
Persamaan (3) menyatakan minimasi total waktu tinggal aktual yang merupakan jumlah waktu tinggal aktual semua pekerjaan di mesin M_1. Pembatas (4) menyatakan bahwa saat mulai pekerjaan yang diproses pertama kali, J_n di mesin M_1 lebih besar atau sama dengan nol. Pembatas (5), (6) dan (7) digunakan untuk menyusun jadwal produksi dengan pendekatan penjadwalan mundur dan saat mulai proses stasiun kerja hilir mengendalikan saat mulai proses stasiun kerja hulu. Pembatas (8) menyatakan bahwa pekerjaan terakhir J_1 harus selesai tepat pada saat *due-date*.

Beberapa literatur yang membahas penjadwalan dengan kendala *due-date,* antara lain Baker dan Scudder (1990) dan Armentano dan Ronconi (1999). Ada beberapa kekhususan model *JCF* dibandingkan dengan model penjadwalan dengan kendala *due-date* lainnya, yaitu pekerjaan yang akan diproses dapat datang kapan saja saat diperlukan dan pekerjaan yang sudah diproses harus dikirim saat *common due-date*. Sehingga model ini tidak mengijinkan adanya pekerjaan yang *tardy.*

Model penjadwalan pekerjaan pada *flowshop* dengan *m* mesin merupakan masalah *NP-hard* (French, 1982)*,* sehingga perlu dikembangkan pendekatan heuristik untuk menjadwalkan semua pekerjaan. Formulasi matematis model *JCF* digunakan untuk menjadwalkan urutan pekerjaan terpilih yang dapat meminimasi total waktu tinggal aktual.

Untuk mengurutkan semua pekerjaan di sistem produksi *flowshop*, dikembangkan sebuah prosedur heuristik yang merupakan modifikasi dari prosedur yang dikembangkan Chan dan Bedworth (1990). Modifikasi dilakukan dengan dasar pendekatan penjadwalan mundur dimulai dari *due-date* ke arah *t*=0 dan menggunakan persamaan pembatas untuk memenuhi kondisi model tersebut. Penyelesaian model akan dimulai pada persoalan penjadwalan dua pekerjaan yang merupakan model dasar pada masalah *sequencing* dan mengamati variasi karakteristiknya. Pada saat dua pekerjaan diproses pada semua mesin dengan urutan sama, maka keputusan penjadwalannya adalah pekerjaan mana yang harus dijadwalkan pertama kali untuk meminimasi waktu tinggal aktual. Persoalan ini dapat diterapkan pada dua, tiga, empat sampai *m* mesin.

A. ($s_{[1,1]} + t_{[1,1]} > t_{[2,2]} + s_{[1,2]}$) B. ($s_{[1,1]} + t_{[1,1]} < t_{[2,2]} + s_{[1,2]}$)



Gambar 2. *Gantt chart* pasangan pekerjaan untuk 2 mesin

Misalkan dua pekerjaan, yaitu $J_{[1]}$ dan $J_{[2]}$ diproses di dua mesin, M_1 dan M_2. Dua alternatif keadaan yang dapat terjadi jika $J_{[2]}$ diproses sebelum $J_{[1]}$ dapat dilihat pada gambar 2. Jika dua pekerjaan tersebut diproses di tiga mesin, maka ada tiga alternatif keadaan yang dapat terjadi dan jika diproses di empat mesin, maka ada empat alternatif keadaan yang dapat terjadi. Waktu tinggal aktual pasangan pekerjaan dapat dilihat pada tabel 1. Tabel 1 juga menunjukkan waktu tinggal aktual dari pasangan pekerjaan yang diproses pada tiga mesin dan empat mesin.

Tabel 1. Penghitungan waktu tinggal aktual pasangan pekerjaan

M	c	F^a *job* 1	F^a *job* 2	F^a pasangan *job*
2	1	$t_{[1,1]}+t_{[1,2]}$	$t_{[2,1]}+s_{[1,1]}+t_{[1,1]}+t_{[1,2]}$	$t_{[1,1]}+2t_{[1,2]}+t_{[2,1]}+\max$
	2	$t_{[1,1]}+t_{[1,2]}$	$t_{[2,1]}+t_{[2,2]}+s_{[1,2]}+t_{[1,2]}$	$\{s_{[1,1]}+t_{[1,1]}, t_{[2,2]}+s_{[1,2]}\}$
3	1	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}$	$t_{[2,1]}+s_{[1,1]}+t_{[1,1]}+t_{[1,2]}+t_{[1,3]}$	$t_{[1,1]}+t_{[1,2]}+2t_{[1,3]}+t_{[2,1]}+\max$
	2	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}$	$t_{[2,1]}+t_{[2,2]}+s_{[1,2]}+t_{[1,2]}+t_{[1,3]}$	$[t_{[1,2]}+\max\{s_{[1,1]}+t_{[1,1]},$
	3	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}$	$t_{[2,1]}+t_{[2,2]}+t_{[2,3]}+s_{[1,3]}+t_{[1,3]}$	$t_{[2,2]}+s_{[1,2]}\}, t_{[2,2]}+t_{[2,3]}+s_{[1,3]}]$
4	1	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}+t_{[1,4]}$	$t_{[2,1]}+s_{[1,1]}+t_{[1,1]}+t_{[1,2]}+t_{[1,3]}+t_{[1,4]}$	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}+2t_{[1,4]}+t_{[2,1]}+$
	2	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}+t_{[1,4]}$	$t_{[2,1]}+t_{[2,2]}+s_{[1,2]}+t_{[1,2]}+t_{[1,3]}+t_{[1,4]}$	$\max < t_{[1,3]}+\max[t_{[1,2]}+\max$
	3	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}+t_{[1,4]}$	$t_{[2,1]}+t_{[2,2]}+t_{[2,3]}+s_{[1,3]}+t_{[1,3]}+t_{[1,4]}$	$\{s_{[1,1]}+t_{[1,1]}, t_{[2,2]}+ s_{[1,2]}\},$
	4	$t_{[1,1]}+t_{[1,2]}+t_{[1,3]}+t_{[1,4]}$	$t_{[2,1]}+t_{[2,2]}+t_{[2,3]}+t_{[2,4]}+s_{[1,4]}+t_{[1,4]}$	$t_{[2,2]}+t_{[2,3]}+s_{[1,3]}], t_{[2,2]}+t_{[2,3]}$
				$+t_{[2,4]}+s_{[1,4]}>$

Dari tabel 1 dapat diperoleh rumusan umum waktu tinggal aktual pasangan pekerjaan $J_{[1]}$ dan $J_{[2]}$ yang diproses pada m mesin yaitu

$$F^a = t_{[2,1]} + \sum_{k=1}^{m} t_{[1,k]} + t_{[1,m]} + R_m , \qquad m \geq 2, \tag{9}$$

dengan: $R_1 = s_{[1,1]}$

$$R_m = \max\{ t_{[1,m\text{-}1]} + R_{m\text{-}1} , \sum_{k=2}^{m} t_{[2,k]} + s_{[1,m]}\}$$

Jika $J_{[1]}$ diproses $J_{[2]}$, maka indeks 1 dan 2 dipertukarkan.

4. Algoritma Penjadwalan *JCF*

Untuk menjadwalkan n pekerjaan pada m mesin, maka digunakan persoalan penjadwalan dua pekerjaan pada m mesin sebagai dasar penjadwalan. Kemudian dikembangkan suatu algoritma agar persoalan penjadwalan n pekerjaan pada m mesin dapat diselesaikan sehingga dapat meminimasi total waktu tinggal aktual.

Untuk menentukan urutan pekerjaan digunakan persamaan (9) pada setiap pasangan pekerjaan yang mungkin. Kemudian pilih pekerjaan awal dari pasangan pekerjaan yang mempunyai waktu tinggal aktual terkecil. Jadwalkan secara mundur pekerjaan menurut *decreasing order* dari frekuensi suatu pekerjaan terpilih. Problem *JCF* layak jika $B_{[n,1]} \geq 0$. Algoritma penjadwalan *JCF* dapat dijelaskan sebagai berikut :

Langkah 0. Inisialisasi dengan menentukan $n, d, m, s_{[i,k]}, t_{[i,k]}$.

Langkah 1. Hitung jumlah pasangan pekerjaan $(R)=n(n\text{-}1)/2$. Tiap pasangan pekerjaan mempunyai dua urutan yang mungkin.

Langkah 2. Set $f = 1$ dan frekuensi pekerjaan terpilih $(a_i) = 0, \forall i$

Langkah 3. Untuk tiap pasangan f yang mungkin, hitung waktu tinggal aktual pasangan pekerjaan di dalam *shop* dengan menggunakan persamaan (9). Pilih urutan dengan waktu tinggal aktual terkecil. Pilih pekerjaan awal dari urutan yang terpilih. Set $a_i = a_i + 1$.

Langkah 4. Jika $f<R$ maka set $f = f+1$ dan kembali ke Langkah 3. Selainnya, lanjutkan ke Langkah 5.

Langkah 5. Jadwalkan secara mundur pekerjaan menurut *decreasing order* dari a_i, $\forall i$. Kemudian hitung total waktu tinggal aktual pekerjaan dengan menggunakan persamaan (3).

Langkah 6. Jika tidak ada solusi untuk $B_{[n,1]} \geq 0$, maka problem tak layak dan *stop*. Selainnya, pilih solusi dengan total waktu tinggal aktual minimum.

5. Contoh Numerik

Contoh numerik ini bertujuan untuk mengetahui karakteristik model penjadwalan *JCF*. Untuk melihat karakteristik model digunakan data yang meliputi data waktu proses, waktu *setup, due-date*, jumlah *job* dan jumlah mesin seperti pada tabel 2 dengan *due-date* = 200.

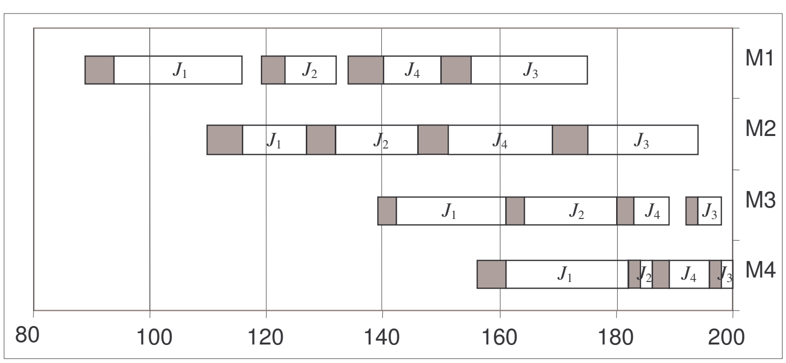
Tabel 2. Waktu proses dan waktu *setup* pekerjaan

M_k	$s_{[1,k]}$	$t_{[1,k]}$	$s_{[2,k]}$	$t_{[2,k]}$	$s_{[3,k]}$	$t_{[3,k]}$	$s_{[4,k]}$	$t_{[4,k]}$
1	5	22	4	9	5	20	6	10
2	6	11	5	14	6	19	5	18
3	3	19	3	16	2	4	3	6
4	5	21	2	2	2	2	3	7

Untuk mendapatkan solusi model penjadwalan *JCF* dijalankan algoritma penjadwalan *JCF*. Solusi model *JCF* menunjukkan bahwa urutan pekerjaan yang dihasilkan mulai dari *due-date* adalah 3 – 4 – 2 – 1 (jika mulai dari saat nol adalah 1 – 2 – 4 – 3). Kemudian urutan pekerjaan tersebut dijadwalkan dengan menggunakan pembatas (5), (6), (7), dan (8) dan hasil perhitungannya dapat dilihat pada tabel 3. Setelah semua pekerjaan dijadwalkan, maka diperoleh total waktu tinggal aktual = 288. Hasil penjadwalan model penjadwalan *JCF* dapat dilihat pada gambar 3.

Tabel 3. Hasil penghitungan $B_{[j,k]}$ model penjadwalan *JCF*

j \ *k*	1	2	3	4
1	155	175	194	198
2	140	151	183	189
3	123	132	164	184
4	94	116	142	161



Gambar 3. *Gantt chart* solusi model penjadwalan *JCF*

6. Kesimpulan

Pada model penjadwalan *JCF* tidak ada satupun pekerjaan yang *tardy* karena menggunakan pendekatan penjadwalan mundur. Algoritma penjadwalan *JCF* tidak menjamin bahwa solusi akan optimal karena urutan optimal untuk n-1 pekerjaan tidak selalu memberikan titik awal yang baik untuk masalah n pekerjaan. Pembandingan kualitas solusi tidak dapat dilakukan karena tidak adanya model pembanding.

Daftar Pustaka

Armentano, V. A. and Ronconi D. P. (1999). Tabu search for total tardiness minimization in flowshop scheduling problems. *Computers Ops. Res.,* 26, pp.219-235.

Baker, K. R. and Scudder, G. D. (1990). Sequencing with earliness and tardiness pelaties: Review. *Operation Res.*, 38(1), pp.22-36.

Chan, D. Y. and Bedworth, D. D. (1990). Design of a scheduling system for flexible manufacturing cell. *Int. J. Prod. Res.,* 28, pp.2037-2049.

French, S. (1982). *Sequencing and Scheduling: An Introdustion to the Mathematics of Job-Shop.* Chichester: Ellis Horwood Limited.

Halim, A. H. and Ohta, H. (1993). Batch-scheduling problems through flow shop with both receiving and delivery just in time. *Int. J. Prod. Res.,* 31 , pp.1943-1955.

Halim, A. H. & Ohta, H. (1994). Batch-scheduling problems to minimize inventory cost in the shop with both receiving and delivery just in time. *Int. J. Prod. Eco.,*33 , pp.185-195.

Rajendran, C. and Chaudhuri, D. (1991). An efficient heuristic approach to the scheduling of jobs in a flowshop. *Eur. J. Opl. Res.,* 61, pp.318-325.

Woo, H. S. and Yim, D. S. (1998). A heuristic algorithm for mean flowtime objective in flowshop scheduling. *Computers Ops. Res.,* 25(3), pp.175-182.