

Penggunaan Aturan Prioritas dalam Penjadwalan Perakitan dan Pemesinan Untuk Mengurangi Makespan

Gianti Puspawardhani dan Rinto Yusriski
Jurusan Teknik Industri, Universitas Jenderal Achmad Yani
gianti.puspawardhani@gmail.com; yusarisaki@yahoo.com

Abstrak— Salah satu karakteristik sistem produksi yang cukup kompleks adalah sistem produksi yang melakukan proses pembuatan komponen serta perakitan yang memiliki aliran *job shop* serta strategi *make to order*. Karakteristik sistem produksi ini memungkinkan adanya beberapa jenis aliran material dalam sistemnya sehingga memerlukan kesesuaian jadwal perakitan dan pembuatan komponen. Untuk mencapai efisiensi sistem, diperlukan sinkronisasi jadwal antara bagian pembuatan komponen dan perakitan agar diperoleh waktu rentang pembuatan (*makespan*) yang minimum. Penjadwalan perakitan dan pemesinan agregat digunakan sebagai jadwal induk produksi dan metode *active scheduling* sebagai metode untuk membuat jadwal pemesinan secara rinci. Penggunaan aturan prioritas dapat mempengaruhi besarnya *lead time* pemesinan suatu kelompok pekerjaan. Aturan *Fewest Operation* membantu mengurangi antrian komponen setengah jadi yang juga mempengaruhi *makespan* pembuatan produk secara keseluruhan. Pembuatan jadwal agregat yang diturunkan menjadi jadwal rinci membantu manajemen dalam memperkirakan batas waktu penyelesaian sebuah pesanan, sehingga sesuai dengan sistem produksi yang menggunakan strategi *make to order*.

Kata kunci — perakitan dan pemesinan agregat, *job shop*, *active scheduling*

I. PENDAHULUAN

Sistem manufaktur dapat diklasifikasi berdasarkan karakteristiknya. Salah satu karakteristik yang membuat sistem cukup kompleks adalah *make to order* dengan variasi item produk yang cukup tinggi. Sistem ini akan mulai produksi jika ada permintaan yang jenis produknya tidak selalu sama, sehingga sistem produksinya harus mengelola sumber daya untuk mengerjakan produk yang beragam dengan konsumen berbeda dan waktu tenggat yang juga berbeda. Kompleksitas ini akan bertambah untuk sistem manufaktur yang melakukan kegiatan pemesinan (pembuatan komponen) serta kegiatan perakitan.

Permasalahan yang akan timbul dalam sistem seperti ini adalah bagaimana caranya mengatur sumber daya supaya sistem efisien tetapi dapat memenuhi waktu tenggat yang telah disepakati dengan konsumen. Hal ini berarti, sistem manufaktur memerlukan keputusan mengenai jadwal perakitan pembuatan komponen yang terpadu.

Kusiak [1] telah mengembangkan metode penjadwalan perakitan dan pemesinan agregat untuk meminimasi *makespan*. Metode ini cukup sederhana untuk menentukan jadwal urutan perakitan sebuah produk yang memiliki struktur kompleks (contoh struktur produk kompleks pada Gambar 1) dan digunakan pada sistem manufaktur fleksibel (*Flexible Manufacturing System, FMS*) yang di dalamnya terdapat kegiatan pemesinan dan perakitan, Karena dalam FMS satu komponen dikerjakan dalam satu mesin maka metode ini tidak membahas masalah penjadwalan mesin untuk pembuatan komponen yang memerlukan lebih dari satu mesin. Untuk sistem manufaktur konvensional, jadwal agregat ini harus diturunkan ke dalam jadwal yang lebih rinci yaitu jadwal pekerjaan yang dialokasikan ke dalam mesin.

Dalam kegiatan pemesinan yang terdiri dari banyak produk dan banyak mesin terdapat dua aliran material yang membentuk sifat berbeda sistem produksi, yaitu ada aliran *flow shop* dan aliran *job shop*. Dalam aliran *job shop*, setiap pekerjaan atau komponen mempunyai *routing* proses yang berbeda dan tidak searah. Kesalahan dalam melepaskan material ke lantai pabrik, dapat menyebabkan penumpukan *WIP* atau sebaliknya yaitu mesin menganggur karena material belum tersedia. Groover [2] menyimpulkan bahwa komponen *lead time* manufaktur yang paling dominan adalah waktu menunggu, sehingga diperlukan metode penjadwalan untuk mengurangi waktu menunggu sehingga dapat meminimumkan *makespan*.

Giffler dan Thompson dalam [3] menyusun algoritma untuk penjadwalan n job m mesin *job shop* untuk memperoleh jadwal aktif yang algoritmanya disebut *active scheduling*, dan Nascimento dan Mario [4] membuktikan bahwa algoritma ini masih sesuai untuk sistem produksi fleksibel.

Jadwal perakitan dan pemesinan agregat telah diterapkan untuk sistem produksi *job shop* oleh Puspawardhani, Yusriski, dan Wahyudi [5]. Penelitian

tersebut menurunkan jadwal perakitan dan pemesinan agregat ke dalam jadwal rinci dengan algoritma *active scheduling*, tetapi seluruh operasi yang diperlukan untuk memproses komponen diperlakukan sebagai *job* yang bersaing bersama-sama untuk dialokasikan (Gambar 2). Hal tersebut menimbulkan *lead time* pemesinan setiap komponen lebih panjang. Puspawardhani, Yusriski [6] membuat perbaikan prosedur penjadwalan dengan memisahkan kelompok pekerjaan pada cabang di level yang sama. Dalam penelitian tersebut dapat disimpulkan bahwa makespan tergantung pada dua faktor, yaitu urutan perakitan dan jadwal pemesinan. Pada jadwal pemesinan, kemungkinan terjadi komponen setengah jadi yang bersaing pada satu mesin sehingga memerlukan pemilihan aturan prioritas untuk menentukan komponen yang mana dulu yang harus diproses. Bedworth [7] membuktikan bahwa aturan prioritas penjadwalan dapat mempengaruhi performansi sistem produksi. Penelitian ini menganalisis pengaruh aturan prioritas penjadwalan terhadap *lead time* pemesinan dan *makespan* untuk jadwal keseluruhan.

II. METODE

A. Penjadwalan Perakitan dan Pemesinan Agregat

Penjadwalan agregat untuk pemesinan dan perakitan dilakukan dengan mengidentifikasi struktur produk yang kemudian digambarkan dalam bentuk digraph. Digraph ini menginformasikan level komponen dan semua pekerjaan perakitan. Untuk produk yang memiliki struktur sederhana, metoda ini menggunakan aturan *the maximum level of depth first* (MLDF) yaitu menjadwalkan pekerjaan pada level terendah sehingga dapat diketahui jadwal untuk perakitan dengan level yang lebih tinggi.

Kusiak (1) mengembangkan penjadwalan perakitan dan pemesinan agregat dengan tahap sebagai berikut :

1. Jika struktur produk sederhana, maka gunakan aturan MLDF dan stop. Jika produk memiliki struktur kompleks, lanjutkan ke langkah 2.
2. Pisahkan beberapa cabang kompleks dan tandai dengan v_0 dan tiap cabang baru dengan v_k (k =indeks nomor cabang) hingga muncul beberapa cabang struktur sederhana yang ditandai dengan g_{ik} (k = indeks turunan cabang pada level struktur berikutnya)
3. Gunakan aturan MDLF untuk cabang g_{ik} yang berhubungan dengan v_i .
4. Untuk setiap cabang sederhana, tentukan nilai *in-process idle time* dan *terminal time*
5. Membuat jadwal ke dalam dua bagian :
Jadwal kelompok 1 : jika *in-process idle time* \leq *terminal time*
Jadwal kelompok 2 : jika *in-process idle time* $>$ *terminal time*
6. Gunakan aturan Longest in-process idle time last untuk kelompok 1 dan aturan Longest terminal time first untuk kelompok 2.
7. Jika jadwal menghasilkan $v_k = v_0$ maka jadwal optimal

8. Untuk setiap jadwal parsial hitung *in-process idle time* dan *terminal time*
Set $k = n-1$ dan kembali ke langkah 3

B. Penjadwalan Pemesinan dengan Active Scheduling

Algoritma yang digunakan dalam penelitian ini yaitu algoritma heuristik Giffler Thompson (3) untuk memperoleh jadwal aktif. Pada saat pekerjaan dialokasikan pada setiap mesin, aka nada kemungkinan beberapa set jadwal bersaing di mesin yang sama. Kondisi ini memerlukan penggunaan aturan prioritas untuk memilih pekerjaan mana dulu yang didahulukan dan yang ditangguhkan. Berikut tahapan dan proses perhitungan dengan menggunakan algoritma Giffler dan Thompson

Step 1:

Set $t=0$ dan P_{st} (jadwal parsial yang mengandung t operasi terjadwal). Set S_t (yaitu kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.

Step 2 :

Tentukan $\varphi_j^* = \min(\sigma_j)$, dimana φ_j ialah saat paling awal operasi j dapat diselesaikan ($\varphi_j = \sigma_j + t_{ij}$).

Tentukan m^* , yaitu mesin dimana φ^* dapat direalisasikan.

Step 3 :

Untuk setiap operasi dalam P_{st} yang memerlukan mesin m^* dan memiliki $\sigma_j < \varphi^*$, buat suatu aturan prioritas tertentu. Tambahkan operasi yang prioritasnya paling besar dalam P_{st} sehingga terbentuk suatu jadwal parsial untuk tahap berikutnya.

Step 4 :

Buat suatu jadwal parsial baru P_{t+1} dan perbaiki kumpulan data dengan cara :

- a. Menghilangkan operasi j dari S_t .
- b. Buat S_{t+1} dengan cara menambah pengikut langsung operasi j yang telah dihilangkan.
- c. Menambahkan satu pada t .

Step 5 :

Kembali ke *step 2* sampai seluruh pekerjaan terjadwalkan.

Notasi matematis pada Algoritma GT set jadwal aktif adalah sebagai berikut [Baker: 1974] :

P_{st} = suatu jadwal parsial yang memiliki sejumlah t operasi yang telah dijadwalkan.

S_t = set operasi-operasi *schedulable* pada *stage* ke- t .

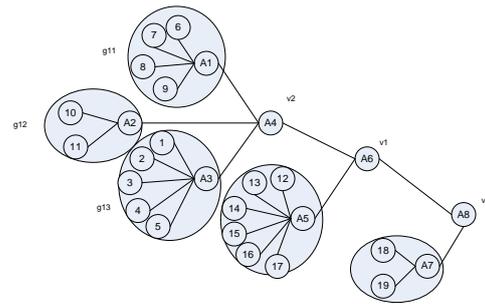
σ_t = saat paling awal dimana operasi $j \in S_t$ dapat mulai dikerjakan.

φ_j = saat paling awal operasi $j \in S_t$ dapat mulai diselesaikan.

t_{ij} = Waktu pemrosesan dari *job* i pada operasi ke- j .

C. Aturan Prioritas

Aturan prioritas yang digunakan dalam penelitian ini adalah *Most Operation Remaining* dan *Fewest Operation Remaining*. Penggunaan *Fewest Operation Remaining* disebabkan, jadwal pemesanan harus mengikuti jadwal perakitan yang sudah ditetapkan sebelumnya berupa urutan perakitan. Jadi untuk menghindari lead time yang panjang, dipilihlah pekerjaan yang sisa operasinya lebih sedikit.



Gambar 3 Pekerjaan pemesinan pada cabang A4 dikelompokkan menjadi satu set jadwal

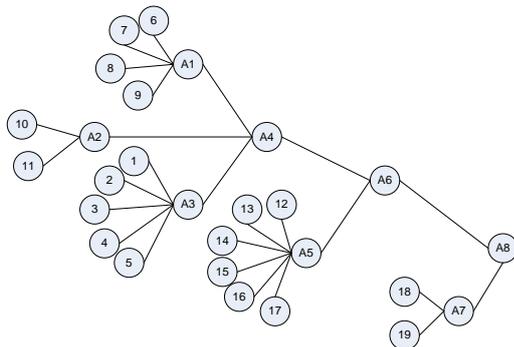
III. HASIL DAN DISKUSI

Pengujian metode ini diterapkan pada produk yang memiliki struktur kompleks (Gambar 1). Pada struktur produk dapat terlihat urutan perakitan berdasarkan level struktur dan terlihat pula terdapat cabang A4 yang memerlukan hasil pengurutan perakitannya yaitu untuk A1, A2, dan A3. Artinya selain diperlukan aturan *the maximum level of depth first* (MLDF), penjadwalan perakitan dan pemesinan agregat memerlukan algoritma Kusiak (1) untuk pengurutan di cabang A4.

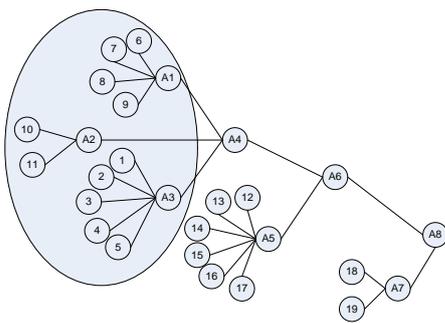
Jika penjadwalan perakitan dan pemesinan agregat sudah ditetapkan (Gambar 4), berikutnya adalah menurunkan jadwal ini menjadi jadwal pemesanan yang lebih rinci. Percobaan pertama, jadwal pemesinan disusun tanpa memisahkan cabang A1, A2, dan A3 pada cabang A4 (Gambar 2), berarti semua komponen yang membentuk A1, A2, dan A3 dikelompokkan dalam satu set jadwal dengan menggunakan aturan Random. Percobaan kedua menyusun jadwal pemesinan dengan memisahkan cabang A1, A2, dan A3 (Gambar 3) dan jika terjadi set jadwal yang bersaing, digunakan aturan *Most Operation Remaining*.

Tabel 1. Nomor part dan operasi

No Part	Operasi 1	Operasi 2	Operasi 3	Operasi 4	Operasi 5
1	BU (2) 0111	CNC(4) 0126	BO(2) 0133		
2	FR(3) 0212	BO(0.5) 0223			
3	FR(2) 0312	BO(0.5) 0323			
4	BU(2) 0411	FR(1) 0422	GS(1) 0435		
5	BU(4) 0511	FR(1) 0522			
6	FR(2.5) 0612	BU(1.5) 0621	BO(1.5) 0633	KB(1) 0648	GD(1) 0655
7	BU(3) 0711	FR(1.5) 0722	GS(1) 0735		
8	BU(2) 0811	BO(1) 0823	KB(0.5) 0838		
9	FR(0.5) 0912				
10	BU(2) 1011	BO(1) 1023	KB(0.5) 1038		
11	BU(3) 1111	BO(1) 1123	WCU(2) 11310	GD(1.5) 1144	KB(0.5) 1158
12	FR(1.5) 1212	BO(1) 1223			
13	BU(1) 1311	FR(0.5) 1322	GS(1) 1335		
14	FR(1) 1412	BU(1.5) 1421	BU(0.5) 1433	GD(1) 1444	
15	FR(1.5) 1512	BO(1) 1523			
16	BU(1.5) 1611	BO(1) 1623			
17	FR(1.5) 1712	BO(0.5) 1723			
18	BU(0.5) 1811				
19	FR(2.5) 1912	BO(1.5) 1923	KB(1) 1938		



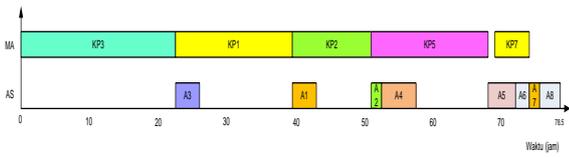
Gambar 1 Struktur Produk Kompleks



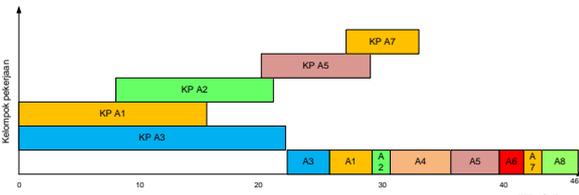
Gambar 2 Pekerjaan pemesinan pada cabang A4 dikelompokkan menjadi satu set jadwal

Tabel 2. Waktu Perakitan pada Setiap Cabang Struktur

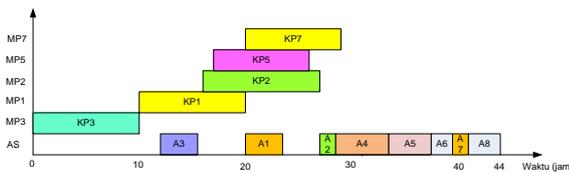
Operasi Perakitan	Waktu Perakitan	Operasi Perakitan	Waktu Perakitan
A1	3,5	A5	4
A2	1,5	A6	2
A3	3,5	A7	1,5
A4	5	A8	3



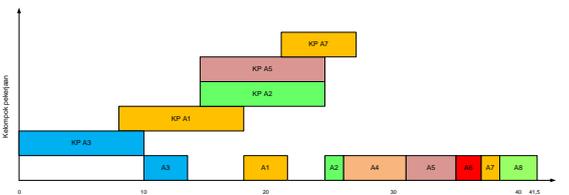
Gambar 4 Hasil penjadwalan perakitan dan pemesinan agregat



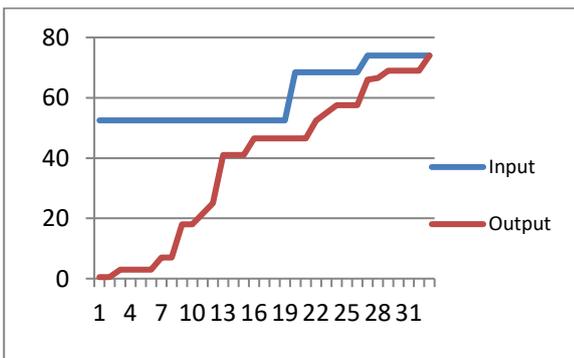
Gambar 5 Jadwal perakitan dan pemesinan setelah diturunkan menjadi jadwal rinci tanpa memisahkan cabang A4



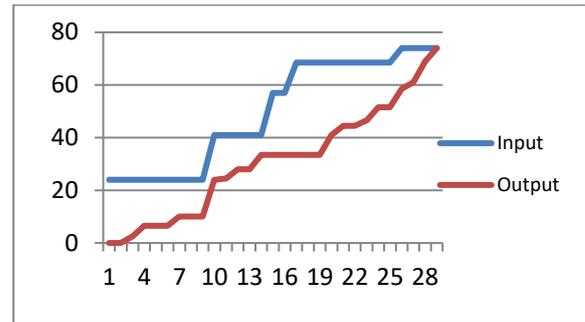
Gambar 6 Jadwal perakitan dan pemesinan setelah diturunkan menjadi jadwal rinci dengan memisahkan cabang A4 menggunakan aturan *Most Operation Remaining*



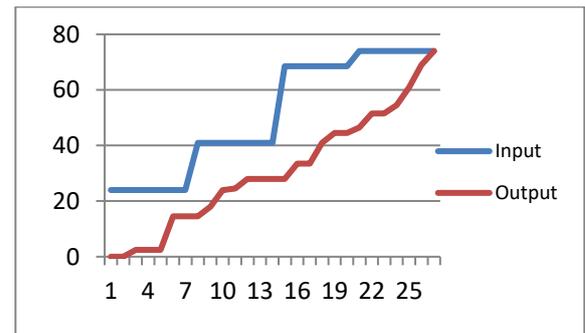
Gambar 7 Jadwal perakitan dan pemesinan setelah diturunkan menjadi jadwal rinci dengan memisahkan cabang A4 menggunakan aturan *Fewest Operation Remaining*



Gambar 8 Diagram Throughput bagian pemesinann tanpa memisahkan cabang A4 menggunakan aturan *Random*



Gambar 9 Diagram Throughput bagian pemesinann i dengan memisahkan cabang A4 menggunakan aturan *Most Operation Remaining*



Gambar 10 Diagram Throughput bagian pemesinann i dengan memisahkan cabang A4 menggunakan aturan *Fewest Operation Remaining*

Penjadwalan perakitan dan pemesinan agregat menghasilkan jadwal yang menyatakan urutan perakitan seluruh sub perakitan yang membentuk produk menggunakan aturan *the maximum level of depth first (MLDF)* yaitu menjadwalkan pekerjaan pada level terendah sampai tertinggi. Pada level yang sama, terjadi cabang yang terdiri dari beberapa komponen seperti yang terjadi pada data dalam penelitian ini, yaitu cabang A1, A2, dan A3 terdapat pada level yang sama. Jika semua komponen pada cabang tersebut *direlease* pada saat yang bersamaan, maka *makespan* pembuatan produk secara keseluruhan adalah 46 jam. Tetapi jika ketiga cabang tersebut *direlease* sesuai dengan urutan perakitannya, maka diperoleh nilai makespan sebesar 44 jam. Hal ini menunjukkan bahwa jumlah pekerjaan yang semakin banyak mengakibatkan banyaknya jumlah komponen setengah jadi yang mengantri. Kondisi ini dapat memperpanjang lead time manufaktur setiap kelompok pekerjaan dan pada akhirnya memperpanjang *makespan*

Aturan prioritas yang berbeda memberikan nilai makespan yang berbeda walaupun dilakukan pemisahan cabang pada struktur yang levelnya sama. Aturan prioritas *most operation remaining* menghasilkan makespan 44 jam dan aturan *fewest operation remaining* menghasilkan makespan sebesar 41,5 jam. Hal ini terjadi karena dengan *fewest operation remaining*, pekerjaan yang sudah dilakukan di lantai pabrik dipercepat pengerjaannya agar segera selesai sehingga dapat mengurangi pekerjaan yang tinggal di lantai pabrik.

Berdasarkan pengujian terlihat bahwa makespan berbanding lurus dengan *lead time* manufaktur. Untuk penjadwalan dengan cabang yang disatukan, menghasilkan *lead time* rata-rata untuk kelompok pekerjaan sebesar 13,1 jam. Penjadwalan dengan memisahkan cabang pada level yang sama dan menggunakan aturan *most operation remaining* menghasilkan *lead time* rata-rata untuk setiap kelompok pekerjaan sebesar 9,8 jam dan dengan aturan *fewest operation remaining* menghasilkan *lead time* 9,1 jam.

IV. KESIMPULAN

1. Penjadwalan perakitan dan pemesinan agregat harus diturunkan menjadi jadwal yang lebih rinci supaya layak diterapkan di lantai produksi. Untuk sistem produksi yang memiliki aliran job shop, metode *active scheduling* dapat membantu membuat jadwal pekerjaan untuk meminimasi *makespan*.
2. Penjadwalan rinci dengan memisahkan cabang kelompok pekerjaan pada level yang sama, dapat mengurangi *makespan* pembuatan produk rakitan yang disebabkan oleh pengurangan *lead time* untuk masing-masing kelompok pekerjaan, sehingga dapat mengurangi *makespan* pembuatan produk keseluruhan.
3. Penggunaan aturan prioritas dapat mempengaruhi besarnya *lead time* pemesinan suatu kelompok pekerjaan. Aturan *Fewest Operation* membantu

mengurangi antrian komponen setengah jadi yang juga mempengaruhi *makespan* pembuatan produk secara keseluruhan.

4. Pembuatan jadwal agregat yang diturunkan menjadi jadwal rinci membantu manajemen dalam memperkirakan batas waktu penyelesaian sebuah pesanan, sehingga sesuai dengan sistem produksi yang menggunakan strategi *make to order*.

DAFTAR PUSTAKA

- [1] Kusiak, Andrew. (1990) *Intelligent Manufacturing Systems*. Prentice Hall International Series In Industrial and System Engineering.
- [2] Groover, Mikel P. (2008) *Automation Production Systems and Computer Integrated Manufacturing*, Prentice Hall
- [3] Baker, Kenneth R (2009). *Principles of Sequencing and Scheduling*. John Wiley & Son, inc.
- [4] Nascimento, Mario (1993). *Giffler and Thompson Algorithms for Job Scheduling is Still Good For Flexible Manufacturing Systems*, *Operation Research Society*, 44, 521-524
- [5] Puspawardhani, G., Wahyudin, W., Yusriski, R., (2009) 'Penggunaan algoritma *active scheduling* dan aggregate machining assembly scheduling dalam pembuatan jadwal induk produksi', *Proceeding Seminar Nasional Sistem Produksi IX*
- [6] Puspawardhani, G., Wahyudin, W., Yusriski, R., (2015) 'Penjadwalan Job untuk Sistem Produksi *Make to Order Assembly Shop* dengan Tujuan Meminimasi *Makespan*', *Proceeding Seminar Nasional Sistem Produksi XI dan Manajemen Rekayasa Kualitas VI*
- [7] Bedworth, David D. James E .Bailey. (1999) *Integrated Productoin Control Systems*. John Wiley & Son, inc.