

# MODEL PENJADWALAN PADA *BATCH PROCESSOR* TUNGGAL DENGAN WAKTU PROSES YANG TIDAK KONSTAN UNTUK MEMINIMASI TOTAL WAKTU TINGGAL AKTUAL

## A SCHEDULING MODEL FOR A SINGLE BATCH PROCESSOR WITH UNFIXED PROCESSING TIME TO MINIMIZE TOTAL ACTUAL FLOW TIME

<sup>1</sup>Murni Dwi Astuti, <sup>2</sup>Abdul Hakim Halim

<sup>1</sup>Program Studi Teknik Industri, Fakultas Rekayasa Industri, Telkom University

<sup>2</sup>Program Studi Teknik Industri, Fakultas Teknologi Industri, Institut Teknologi Bandung

<sup>1</sup>mumidwiasuti@telkomuniversity.ac.id, <sup>2</sup>ahakimhalim@ispitb.org

**Abstrak**— Penjadwalan *batch* sudah didiskusikan di beberapa penelitian. Terdapat dua jenis penjadwalan *batch* yaitu *job processor* dan *batch processor*. Pada *job processor*, *part* diproses secara bergantian sampai semua *part* dalam satu *batch* selesai diproses, sedangkan pada *batch processor* *part* diproses bersama dalam satu waktu sehingga waktu untuk memproses *batch* besarnya sama dengan waktu untuk memproses satu *part*. Berbeda dengan penelitian sebelumnya, pada penelitian ini waktu untuk memproses *batch* bukan hanya waktu untuk memproses satu *part* namun gabungan dari waktu yang diperlukan untuk memproses satu *part* dan waktu untuk memasang *tools* pada setiap *part* dalam satu *batch*. Penelitian ini merupakan tahap ke dua, setelah sebelumnya dikembangkan model yang sama namun untuk masalah *single item* dengan *due date* yang sama maka pada penelitian ini *due date* dari item berbeda – beda. Model penjadwalan untuk masalah *multi due date* diselesaikan dengan menganggap satu interval waktu yang merupakan selisih antar dua *due date* yang berurutan sebagai masalah *common due date*. Jumlah minimum *batch* yang terbentuk untuk setiap interval waktu ini adalah satu dan waktu minimal untuk memproses *batch* besarnya sama dengan waktu untuk memproses satu *part* ditambah dengan waktu yang diperlukan untuk memasang *tools* untuk seluruh *part* pada satu *batch*. Jika waktu pada interval waktu tidak cukup untuk memproses *part* yang diminta pada interval waktu tersebut maka *part* diproses pada interval waktu setelahnya. Model diuji dengan beberapa set data hipotetik. Variabel keputusan adalah jumlah *batch* dan ukuran *batch* untuk setiap interval waktu, dan urutan pengerjaan *batch* sehingga akan meminimasi total waktu tinggal aktual.

**Kata kunci:** *batch*, *batch processor*, *single item*, *common due date*, *multi due date*, waktu tinggal aktual

**Abstract**— Batch scheduling has been discussed in several studies. There are two types of batch scheduling, job processor and batch processor. On job processor, parts are processed alternately until all parts are processed, while in batch processor parts in batch are processed together in one time. Hence the processing time for

batches is equal to one part processing time. Unlike the previous study, in this study batch processing time not only time to process one part but it is the sum of processing time of one part and time to install tools on each part in one batch. This paper is continuation of previously developed batch scheduling model for single item common due date problem, hence for multi due date. The scheduling model for single item multi due date is solved by considering one time interval which is the difference between two successive due dates as a common due date issue. The minimum number of batches generated for each time interval is one and the minimum time required to process this batch is the one-part processing time plus the time required to install tools on all parts in one batch. If the time at the time interval is not sufficient to process the requested part at that time interval then the part is processed at the time interval thereafter. The resulting model was tested using several sets of hypothetical data. The decision variables in this scheduling model are the number of batches and batch size for each batch at each time interval, and the order of batch work so as to minimize the total actual flow time.

**Keywords:** *batch*, *batch processor*, *single item*, *common due date*, *multi due date*, actual flow time

### I. PENDAHULUAN

Sukoyo dkk [1] menguraikan penjadwalan *batch*. Karakteristik permasalahan yang diselesaikan dengan penjadwalan *batch* tidak sama dengan penjadwalan *job*. Untuk penjadwalan *batch*, *job* yang akan dikerjakan dibagi ke dalam beberapa kelompok. Kelompok – kelompok ini disebut sebagai *batch*, sehingga jumlah *job* yang akan dikerjakan berubah menjadi jumlah *batch* dan waktu yang diperlukan untuk memproses *job* ini sangat tergantung kepada bagaimana *batch* dibentuk. Hal ini menyebabkan permasalahan penjadwalan menjadi tidak sederhana karena masalah yang harus diselesaikan tidak hanya menjadwalkan pengerjaan

*batch* saja namun juga harus ditentukan bagaimana membagi *job* yang ada ke dalam *batch*, menentukan berapa ukuran dari *batch* yang terbentuk, dan menentukan urutan pemrosesan *batch* yang sudah dibentuk.

Penjadwalan *batch* dilakukan karena adanya pertimbangan efisiensi [2]. Dengan menerapkan penjadwalan *batch*, pemrosesan *job* dapat dilakukan lebih efisien dibanding memproses *job* secara bergantian dan saat pemrosesan *job* satu dengan *job* yang lain dibutuhkan waktu untuk *set up*. Kegiatan *set up* ini diperlukan untuk mengubah peralatan yang digunakan maupun membersihkan mesin yang digunakan.

Dalam industri manufaktur, penjadwalan *batch* merupakan sistem yang sangat banyak diaplikasikan [3]. Dengan dilakukannya penjadwalan *batch* akan menghindari *set up*. Jenis *batch* terdiri dari dua yaitu *batch* serial dan *batch* paralel. *Batch* serial, merupakan *batch* dengan kondisi *job* yang berada pada satu *batch* adalah *job* dengan *set up* yang sama dan *job* diproses secara bergantian sampai seluruh *job* yang ada dalam satu *batch* selesai diproses. *Batch* paralel, adalah *batch* dengan kondisi beberapa *job* digabungkan menjadi satu *batch* dengan memperhatikan batasan kapasitas mesin dan *job* yang berada pada satu *batch* diproses bersamaan pada satu waktu. Pada saat *part* dalam *batch* mulai diproses, mesin tidak dapat diberhentikan sampai mesin selesai memproses *part* dalam *batch*. Pada istilah penjadwalan yang sering digunakan, sistem *batch* serial sering disebut sebagai *job processor* dan sistem *batch* paralel sering disebut dengan *batch processor*.

*Batch* merupakan sekumpulan *part* yang diproses dengan *set up* yang sama [4]. *Batch* yang dihasilkan dari sejumlah *part* yang diminta dapat dikerjakan menggunakan *job processor* dan *batch processor*. Pada *job processor*, *part* dalam *batch* diproses secara bergantian, sehingga waktu yang dibutuhkan untuk memproses seluruh *part* yang terdapat dalam satu *batch* adalah waktu untuk memproses satu *part* dikalikan dengan jumlah *part* yang berada pada satu *batch*. Pada *batch processor*, *part* yang terdapat pada satu *batch* diproses secara bersamaan, sehingga waktu yang dibutuhkan untuk memproses seluruh *part* dalam *batch* nilainya sama dengan waktu untuk memproses satu *part*.

Masalah penjadwalan *batch* yang paling sederhana adalah penjadwalan *batch* pada mesin tunggal. Penelitian yang telah dilakukan untuk masalah penjadwalan *batch* pada mesin tunggal diantaranya Halim dkk [5], Dobson dan Nambimadom [6], Sukoyo dkk [7], Hidayat dkk [4], dan Astuti dkk [8]. Halim dkk [5] mengusulkan model untuk menyelesaikan permasalahan penjadwalan *batch* pada *job processor* untuk *single item* dengan *due date* yang berbeda – beda (*multi due date*) yang akan meminimasi total waktu tinggal aktual. Waktu yang diperlukan untuk memproses *part* dalam *batch* adalah  $tQ$  dengan  $t$  adalah waktu yang dibutuhkan untuk memproses satu *part* dan  $Q$  merupakan jumlah *part* yang berada pada satu *batch*. Dobson dan Nambimadom [6] membahas penjadwalan *batch* untuk kasus *multi item* dengan sistem *batch* berupa *batch processor* yang meminimasi total *weighted flow time*. Pada penelitian ini seluruh *job*

diasumsikan siap pada saat  $t = 0$ . *Job* berasal dari famili yang berbeda- beda sehingga waktu proses antar famili berbeda. Kriteria pengelompokan *job* ke dalam satu *batch* adalah kesamaan waktu proses sehingga *job* yang hanya berasal dari famili sejenis yang diproses dalam satu *batch*. Sukoyo dkk [7] mengusulkan model untuk memecahkan penjadwalan *batch* yang diproses menggunakan *job processor* dengan waktu proses yang nilainya mengalami kenaikan. Kenaikan ini disebabkan oleh urutan pengerjaan *batch*. Hidayat dkk [4] mengusulkan model untuk memecahkan penjadwalan *batch* yang diproses menggunakan *batch processor* untuk *single item*. Waktu proses untuk memproses seluruh *part* dalam *batch* adalah konstan yang besarnya sama dengan waktu yang diperlukan untuk memproses satu *part*. Astuti dkk [8] mengusulkan model untuk penjadwalan *batch* yang diproses dengan menggunakan *batch processor* tunggal untuk masalah *single item* dengan *due date* yang sama (*common due date*) dengan waktu proses yang tidak konstan agar total waktu tinggal aktual minimal.

Pada penelitian ini diusulkan model penjadwalan *batch* yang merupakan pengembangan dari model pada penelitian – penelitian sebelumnya. Model yang diusulkan untuk menyelesaikan permasalahan penjadwalan *batch* yang diproses menggunakan *batch processor* untuk *single item* dengan *due date* yang berbeda – beda dengan waktu proses yang tidak konstan agar total waktu tinggal aktual minimal. Pengembangan yang dilakukan dari penelitian sebelumnya adalah pada waktu proses *batch* yang tidak konstan karena pemasangan *tools* pada setiap *part* dalam *batch* yang akan diproses dan pada *due date* yang berbeda- beda.

## II. STUDI LITERATUR

Waktu tinggal aktual merupakan lamanya suatu *job* berada di lantai produksi dari saat *job* mulai diproses sampai saat *job* tersebut dikirim pada *due date* nya [9]. Jika terdapat sejumlah *job* ( $n$ ) yang diminta pada *due date* yang sama ( $d$ ) dengan waktu proses *job* adalah  $p_i$ , saat mulai pemrosesan *job*  $i$  adalah  $B_i$ , dan waktu *set up* mesin adalah  $s$  dan diasumsikan besarnya waktu *set up* tidak dipengaruhi oleh urutan pemrosesan *job*, sehingga waktu tinggal aktual diberikan oleh persamaan:

$$F_i^a = d - B_i, \quad \text{untuk } i = 1, 2, \dots, n \quad (1)$$

Dengan menggunakan pendekatan penjadwalan mundur maka persamaan (1) dapat ditulis menjadi:

$$F_i^a = \sum_{j=1}^i (p_j + s_j) - s_i \quad \text{untuk } i = 1, 2, \dots, n \quad (2)$$

Persamaan (2) merupakan formulasi untuk menentukan waktu tinggal aktual *job*. Waktu tinggal aktual *batch* dihitung melalui cara seperti pada persamaan (2), namun untuk waktu proses dikalikan dengan jumlah *part* yang terdapat pada satu *batch*, sehingga didapatkan waktu tinggal aktual untuk *batch* adalah:

$$F_i^a = \sum_{j=1}^i (t_j Q_{[j]} + S_j) - S_i \quad (3)$$

Dengan  $t_j$  adalah waktu proses untuk setiap *part* dalam *batch*,  $Q_{[j]}$  adalah jumlah *part* yang berada pada satu *batch* dan  $s_i$  adalah waktu yang diperlukan untuk kegiatan *set up* sebelum *batch* diproses. Persamaan (3) digunakan untuk menentukan waktu tinggal aktual untuk *batch* yang diproses pada *job processor* dimana *job* yang terdapat pada satu *batch* diproses secara bergantian. Pada *batch processor*, menggunakan sistem yang berbeda dengan *job processor*. Pada *batch processor*, seluruh *part* yang terdapat pada satu *batch* diproses secara bersamaan sehingga waktu yang dibutuhkan untuk memproses *batch* besarnya sama dengan waktu untuk memproses satu *part* [4]. Dengan demikian waktu tinggal aktual seluruh *batch* pada *batch processor* adalah:

$$FS^a = \sum_{i=1}^N \left\{ \sum_{j=1}^i (t_j + s_j) \right\} - s_i \quad (4)$$

Persamaan (4) digunakan untuk menentukan total waktu tinggal aktual untuk total *part* pada total *batch* dengan waktu proses  $t$  dan waktu *set up*  $s$ .

Permasalahan penjadwalan *batch* yang diproses pada *batch processor* dengan besarnya waktu proses tidak konstan karena adanya aktivitas memasang *tools* pada *part* dalam *batch* sebelum diproses pada mesin. Dengan demikian, waktu yang dibutuhkan untuk memproses *batch* merupakan penjumlahan dari waktu untuk memproses satu *part* dan waktu untuk memasang *tools* untuk setiap *part* dalam satu *batch*. Total waktu tinggal aktual untuk permasalahan ini dirumuskan sebagai berikut:

$$F^a = \sum_{i=1}^N \left\{ \sum_{j=1}^i \left( (t_j + aQ_{[j]}) + s_j \right) - s_i \right\} \quad (5)$$

$t_j$  menyatakan waktu proses *batch processor*,  $a$  menyatakan waktu untuk memasang *tools* pada setiap *part* dalam *batch*,  $Q_{[j]}$  menyatakan jumlah *part* pada satu *batch*, dan  $s_i$  menyatakan waktu *set up* *batch*.

### III. METODOLOGI PENELITIAN

Model penjadwalan *batch* untuk *single item multi due date* merupakan pengembangan dari model *single item common due date*. Pada permasalahan *common due date* semua *part* yang selesai dikerjakan dikirim pada waktu (*due date*) yang sama. Pada permasalahan *multi due date* pengiriman *part* yang selesai dikerjakan dilakukan pada saat yang berbeda – beda sesuai dengan *due date* dari masing – masing *part*. Pada penjadwalan *multi due date*, penjadwalan memperhatikan *due date* dari *part* sehingga pada permasalahan ini tidak diperbolehkan mengirimkan *part* melebihi *due date* yang diminta.

Misalkan terdapat  $n_1, n_2, \dots, n_r$  *part* yang merupakan *item* sejenis (*single item*) yang akan diproses pada *batch processor* tunggal dengan *due date* masing – masing *part* adalah  $d_1, d_2, \dots, d_r$ . Waktu proses untuk setiap *part* adalah sama ( $t$ ) dan pengelompokan *part* ke dalam *batch* berdasarkan *due date* dari masing- masing *part*. *Batch processor* yang digunakan untuk memproses *part* membutuhkan waktu *set up* ( $s$ ) pada

setiap pemrosesan *batch*. Lamanya waktu *set up* untuk setiap *batch* tidak tergantung kepada banyaknya *part* yang akan diproses dalam setiap *batch*. Proses *set up* ini dilakukan sebelum kedatangan *part* di lantai produksi. Ketika *part* akan diproses pada *batch processor*, pada setiap *part* dipasang *tools*. Waktu yang dibutuhkan untuk memasang *tools* diasumsikan sama untuk setiap *part* ( $a$ ). Pemasangan *tools* dilakukan saat *part* sudah siap di lantai produksi sehingga akan mempengaruhi waktu untuk memproses *batch*. Waktu untuk memproses *batch* menjadi waktu proses pada *batch processor* dan waktu untuk pemasangan *tools* pada setiap *part* untuk satu *batch*. Permasalahan disini adalah bagaimana mengelompokkan sejumlah *part* ( $n_h$ ) untuk  $h = 1, 2, \dots, r$  ke dalam *batch* dan bagaimana menjadwalkan *batch* yang dihasilkan agar dapat memenuhi *due date* dari masing-masing *part* dengan waktu tinggal aktual seluruh *part* pada lantai produksi ( $F^a$ ) adalah minimal.

Pada penelitian ini didapatkan model matematika untuk penjadwalan *batch* untuk *single item multi due date* untuk meminimasi total waktu tinggal aktual. Dengan menggunakan notasi:

- a Waktu pemasangan *tools* pada *part* yang nilainya diasumsikan konstan (unit waktu/ unit)
- b urutan interval waktu
- $B_{h[i]}$  Waktu mulai *batch* pada interval  $H_h$  pada posisi ke-  $i$
- $d_b$  *Due date* ke- $h$  ( $h=1, 2, 3, \dots, r$ ) yang dihitung dari posisi terakhir pada skala waktu (*multi due date*)
- $D_h$  Jumlah *part* yang diproses pada periode  $H_h$
- $F^a$  Total waktu tinggal aktual *part* di lantai produksi
- $h$  Indeks untuk mengidentifikasi interval waktu, diantara dua *due date* yang berdekatan, dalam periode penjadwalan,  $h=1, \dots, r$ .  $h$  dihitung dari posisi terakhir pada rentang waktu
- $H_h$  Periode waktu yang didefinisikan sebagai:
 
$$H_h = \begin{cases} d_h - d_{h-1} - s_k, & h = 1, \dots, (r-1) \\ d_h, & h = r \end{cases}$$
- $i$  Indeks yang menunjukkan posisi *batch*, dihitung dari posisi terakhir pada skala waktu dari *batch* pada jadwal produksi.  $i \neq j$
- $j$  Indeks yang menunjukkan posisi *batch*, dihitung dari posisi terakhir pada skala waktu dari *batch* pada jadwal produksi.  $i \neq j$
- $n_h$  Jumlah *part* yang diminta pada tiap interval  $h$
- $N_h$  Jumlah *batch* untuk setiap interval waktu  $H_h$  pada kasus *multi due date*
- $Q_{h[i]}$  Ukuran *batch* (unit) pada posisi ke- $i$  pada interval waktu  $H_h$
- $Q_{h[j]}$  Ukuran *batch* (unit) pada posisi ke- $j$  pada interval waktu  $H_h$
- $r$  Jumlah *due date*
- $s$  Waktu *set up* *batch processor*
- $t$  waktu proses untuk setiap *part*
- $s_i$  Waktu *set up* *batch processor* pada posisi ke-  $i$
- $s_j$  Waktu *set up* *batch processor* pada posisi ke-  $j$
- $W_h$  Jumlah *part* yang dapat dikerjakan pada interval waktu  $d_h$  yang bukan  $H_h$  (*multi due date*)

Asumsi yang digunakan:

- Permasalahan yang diselesaikan adalah penjadwalan untuk *single item multi due date* yang diproses pada *batch processor* tunggal.
- Waktu untuk memasang *tools* diasumsikan sama untuk setiap *part*
- Waktu *set up* untuk setiap *batch* diasumsikan konstan (tidak terpengaruh oleh urutan pemrosesan *batch* dan jumlah *part* yang terdapat pada satu *batch*)
- Waktu untuk memproses setiap *part* diasumsikan konstan
- Ukuran dari setiap *batch* yang dihasilkan dinyatakan dalam bilangan *real* positif
- Jumlah *batch* yang dihasilkan untuk setiap interval waktu (periode) adalah integer
- Penjadwalan *batch* dilakukan dengan pendekatan *backward scheduling*.

Berdasarkan asumsi dan notasi yang sudah ditetapkan, maka masalah pada penelitian ini dirumuskan sebagai berikut:

Minimasi:

$$F^a = \left[ \sum_{h=1}^r \left[ \sum_{i=1}^{N_h} \left\{ \sum_{j=1}^i \left( (t + a Q_{h[j]}) + s_j \right) - s_i \right\} Q_{h[i]} \right] \right] + \left[ \sum_{b=1}^{r-1} \left\{ \sum_{h=b+1}^r (D_h - n_h) \right\} (d_b - d_{b+1}) \right] \quad (6)$$

Pembatas :

$$(N_h - 1) s + t N_h + a \sum_{i=1}^{N_h} Q_{h[i]} \leq H_h, \quad h=1, \dots, r \quad (7)$$

$$\sum_{h=b+1}^r (D_h - n_h) \geq 0, \quad b=1, \dots, (r-1) \quad (8)$$

$$\sum_{h=1}^r D_h - n_h = 0, \quad (9)$$

$$\sum_{i=1}^{N_h} Q_{h[i]} = D_h, \quad h=1, \dots, r \quad (10)$$

$$B_{h[1]} + a Q_{h[1]} + t = d_h, \quad h=1, \dots, r \quad (11)$$

$$Q_{h[i]}, D_h \geq 0, N_h \geq 1, i = 1, \dots, N_h; h = 1, \dots, r \quad (12)$$

Persamaan (7) menunjukkan bahwa pada setiap periode penjadwalan  $H_h$ , seluruh *part*  $D_h$  harus diproses pada interval waktu produksi  $H_h$ , yang merupakan selisih dari  $(d_b - d_{b+1})$ . Persamaan (8) menunjukkan bahwa jumlah seluruh *part* yang diproduksi dalam interval  $H_h$  sampai  $H_r$  minimal sama atau harus lebih besar dibandingkan dengan total jumlah *part* yang diminta pada interval waktu tersebut. Persamaan (9) menunjukkan bahwa total jumlah *part* yang diproses dalam periode penjadwalan harus sama dengan total jumlah *part* yang diminta pada periode penjadwalan. Persamaan (10) menunjukkan adanya keseimbangan material dalam tiap periode  $H_h$ . Persamaan (11) menunjukkan bahwa saat selesai pemrosesan *batch* ke satu pada setiap periode penjadwalan  $H_h$  harus sama dengan *due date* dari periode tersebut sehingga hal ini menjamin tidak akan terjadi keterlambatan pengiriman. Persamaan (12) menunjukkan bahwa ukuran *batch* pada setiap interval waktu, jumlah permintaan pada setiap interval waktu harus lebih besar atau sama dengan nol, dan jumlah *batch* untuk setiap interval waktu harus lebih besar atau sama dengan satu.

### Sub Algoritma 1.1

#### Langkah 1

Gunakan algoritma yang diusulkan pada [8]. *Batch*  $n_h^*$  dengan *dummy due date*  $H_h$  yang nilainya adalah  $(d_h - d_{h+1})$ .  $n_h^*$  merupakan jumlah *part* yang akan dijadwalkan. Untuk  $h = 1$ , maka  $n_h^* = n_1$ . Jika  $(N_h - 1)s + N_h t + a n_h^* \leq H_h$ , set  $D_h = n_h^*$ , dan lanjutkan ke langkah 3. Jika lainnya,, lanjutkan ke langkah 2.

#### Langkah 2

Dengan menggunakan *due date*  $d_h$  yang bukan  $H_h$ , gunakan algoritma yang diusulkan pada [8], *batch*  $n_h^*$ , dan jadwalkan *batch* yang dihasilkan pada periode waktu untuk mendapatkan  $W_h$  yang merupakan jumlah *part* yang akan diproses pada rentang  $H_h$ . Gunakan algoritma pada [8], *batch* kembali  $W_h$  dengan interval waktu  $H_h$ , dan jadwalkan *batch* yang dihasilkan dengan waktu yang tersedia. Jika pada jadwal yang dihasilkan tidak terdapat *idle time*  $\omega$ , set  $D_h = W_h$  dan lanjutkan ke langkah 3. Jika terdapat *idle time*, dan  $\text{idle time} \geq (t + a)$  maka tetapkan  $D_h = W_h + \frac{\omega - t}{a}$  Lanjutkan ke langkah 3.

#### Langkah 3.

Definisikan  $D_h$  sebagai jumlah *part* yang akan diproses pada mesin dengan periode  $H_h$

### Algoritma 1.1

Langkah 0.

Inisialisasi parameter masalah. Lanjutkan ke

langkah 1.

Set  $h = 1$ . Lanjutkan ke langkah 2.

Langkah 1.

Gunakan Sub algoritma 1.1, definisikan  $D_h$ . Lanjutkan ke langkah 3.

langkah 3.

Set  $\gamma_h = n_h^* - D_h$ .  $\gamma_h$  menyatakan jumlah *part* yang belum dijadwalkan pada periode  $H_h$  ( untuk  $\gamma_0 = 0$ ). Definisikan jumlah *part* yang dijadwalkan di periode  $H_{h+1}$ , gunakan  $n_{h+1}^* = n_{h+1} + \gamma_h$ . Set  $h = h + 1$ . Jika  $h \leq r$  kemudian lanjutkan ke langkah 2. Jika yang lainnya lanjutkan ke langkah 4.

Langkah 4.

Gunakan algoritma pada [8], *batch* kan  $D_h$  dengan  $H_h$  dan kemudian jadwalkan *batch* yang dihasilkan pada jadwal yang terakhir. Berdasarkan hal ini, hitung total waktu tinggal aktual dari seluruh *part* pada rantai produksi menggunakan persamaan (6).

## IV. HASIL DAN ANALISIS

Model yang dihasilkan pada penelitian ini diuji dengan menggunakan data hipotetik. Data hipotetik yang digunakan disusun dalam tiga skenario berikut:

Skenario 1 disusun dengan menggunakan data permintaan *part single item* sejumlah  $n_1, n_2, n_3, n_4$  dengan *due date*  $d_1, d_2, d_3, d_4$ . Ketentuan pada skenario 1 adalah seluruh permintaan

part pada setiap periode permintaan dapat dikerjakan pada periode sesuai dengan periode saat part tersebut diminta. Waktu set up mesin adalah s, waktu untuk memproses satu part adalah t, dan waktu memasang tools untuk setiap part adalah a.

Tabel 1  
Set data dan hasil perhitungan total waktu tinggal aktual untuk skenario 1

$n_h$	$d_h$	t	s	a	$N_h$	$Q_{h[i]}$	$B_{h[i]}$	$F^a$
$n_1 = 200,$ $n_2 = 180,$ $n_3 = 175,$ $n_4 = 200$	$d_1 = 1200,$ $d_2 = 900,$ $d_3 = 650,$ $d_4 = 400$	30	5	1	$N_1 = 2$ $N_2 = 2$ $N_3 = 2$ $N_4 = 3$	$Q_{1[1]} = 117,5,$ $Q_{1[2]} = 82,5,$ $Q_{2[1]} = 107,5,$ $Q_{2[2]} = 72,5,$ $Q_{3[1]} = 105, Q_{3[2]} = 70,$ $Q_{4[1]} = 101,67,$ $Q_{4[2]} = 66,67,$ $Q_{4[3]} = 31,66$	$B_{1[1]} = 1052,5,$ $B_{1[2]} = 935,$ $B_{2[1]} = 762,5,$ $B_{2[2]} = 655,$ $B_{3[1]} = 515, B_{3[2]} = 410,$ $B_{4[1]} = 268,33,$ $B_{4[2]} = 166,66,$ $B_{4[3]} = 100$	141.154,2
Total=755					Total=755			

Pada tabel 1 dapat dilihat bahwa total part yang diminta setiap periode sama dengan total part yang diproses di setiap periode dan total part yang diminta untuk seluruh periode sama dengan total part yang diproses untuk seluruh periode. Hal ini menunjukkan bahwa seluruh permintaan part pada setiap periode dapat diproses pada periode sesuai dengan periode part tersebut diminta. Tidak terdapat part yang dikerjakan pada periode setelah periode permintaan part. Pada skenario 1 ini diperoleh jadwal yang layak dengan waktu mulai pemrosesan batch pada saat  $t = 31,66$  dengan total waktu tinggal aktual adalah 141.154,2 menit.

Skenario 2 disusun dengan menggunakan data permintaan part single item sejumlah  $n_1, n_2, n_3, n_4$  dengan due date  $d_1, d_2, d_3, d_4$ . Ketentuan pada skenario 2 adalah tidak seluruh permintaan part pada setiap periode dapat dikerjakan pada interval waktu sesuai dengan waktu saat part tersebut diminta. Part yang tidak dapat dikerjakan pada periode part tersebut diminta akan dikerjakan pada periode setelahnya. Waktu set up mesin adalah s, waktu untuk memproses satu part adalah t, dan waktu memasang tools untuk setiap part adalah a.

Tabel 2  
Set data dan hasil perhitungan total waktu tinggal aktual untuk skenario 2

$n_h$	$d_h$	t	s	a	$N_h$	$Q_{h[i]}$	$B_{h[i]}$	$F^a$
$n_1 = 250,$ $n_2 = 220,$ $n_3 = 240,$ $n_4 = 205$	$d_1 = 1200,$ $d_2 = 900,$ $d_3 = 650,$ $d_4 = 400$	30	5	1	$N_1 = 1$ $N_2 = 2$ $N_3 = 3$ $N_4 = 1$	$Q_{1[1]} = 250,$ $Q_{2[1]} = 107,5,$ $Q_{2[2]} = 72,5,$ $Q_{3[1]} = 75,83,$ $Q_{3[2]} = 40,83,$ $Q_{3[3]} = 5,83,$ $Q_{4[1]} = 362,5,$ $n_{23} = 40, n_{34} = 157,51$	$B_{1[1]} = 920,$ $B_{2[1]} = 762,5,$ $B_{2[2]} = 655,$ $B_{3[1]} = 544,17,$ $B_{3[2]} = 468,34,$ $B_{3[3]} = 427,51,$ $B_{4[1]} = 2,5$	310.941,9
Total=915					Total= 915			

Pada tabel 2 dapat dilihat bahwa total part yang diminta setiap periode tidak sama dengan total part yang diproses setiap periode, seperti pada interval waktu ke empat. Jumlah

permintaan pada periode 4 adalah 205, namun jumlah part yang dikerjakan pada interval waktu tersebut sejumlah 485 part. Hal ini menunjukkan bahwa terdapat sejumlah part yang diminta pada periode sebelumnya namun waktu yang tersedia pada periode sebelumnya tidak cukup untuk melakukan pemrosesan part, sehingga part diproses pada periode setelahnya. Namun, jika dilihat pada total part yang diminta untuk seluruh periode sama dengan total part yang diproses untuk seluruh periode. Hal ini menunjukkan bahwa meskipun terdapat part yang dikerjakan tidak sesuai dengan periode permintaan part, seluruh permintaan part tetap dapat diproses dan diperoleh jadwal yang layak dengan waktu mulai pemrosesan batch pada saat  $t = 2,5$  dengan total waktu tinggal aktual adalah 310.941,9 menit.

Skenario 3 disusun dengan menggunakan data permintaan part single item sejumlah  $n_1, n_2, n_3, n_4$  dengan due date  $d_1, d_2, d_3, d_4$ . Ketentuan pada skenario 3 adalah tidak seluruh permintaan part dapat dikerjakan pada periode sesuai dengan waktu saat part tersebut diminta. Part yang tidak dapat dikerjakan pada interval waktu dimana part tersebut diminta dikerjakan pada periode setelahnya, namun waktu yang tersedia tidak mencukupi untuk mengerjakan permintaan part. Waktu set up mesin adalah s, waktu untuk memproses satu part adalah t, dan waktu memasang tools untuk setiap part adalah a.

Tabel 3  
Set data dan hasil perhitungan total waktu tinggal aktual untuk skenario 3

$n_h$	$d_h$	t	s	a	$N_h$	$Q_{h[i]}$	$B_{h[i]}$	$F^a$
$n_1 = 275,$ $n_2 = 250,$ $n_3 = 260,$ $n_4 = 210$	$d_1 = 1200,$ $d_2 = 900,$ $d_3 = 650,$ $d_4 = 400$	30	5	1	$N_1 = 2$ $N_2 = 3$ $N_3 = 2$ $N_4 = 2$	$Q_{1[1]} = 121,25,$ $Q_{1[2]} = 86,25,$ $Q_{2[1]} = 78,95,$ $Q_{2[2]} = 43,95,$ $Q_{2[3]} = 8,95,$ $Q_{3[1]} = 97,06,$ $Q_{3[2]} = 62,06,$ $Q_{4[1]} = 170,$ $Q_{4[2]} = 135,$ $n_{12} = 67,5, n_{23} = 185,65, n_{24} = 26,53, n_{34} = 260$	$B_{1[1]} = -$ $1048,75, B_{1[2]} = 927,5,$ $B_{2[1]} = 791,05, B_{2[2]} = 712,55,$ $B_{2[3]} = 667,6,$ $B_{3[1]} = 522,94, B_{3[2]} = 425,88,$ $B_{4[1]} = 200,$ $B_{4[2]} = 30$	
Total=995					Total= 802,07			

Pada tabel 3 dapat dilihat bahwa total part yang diminta setiap periode tidak sama dengan total part yang diproses di setiap periode. Hal ini menunjukkan bahwa terdapat sejumlah part yang diminta pada periode sebelumnya namun waktu pada periode sebelumnya tidak cukup untuk melakukan pemrosesan part, sehingga part diproses pada periode setelahnya. Namun, jika dilihat pada jumlah seluruh part yang diminta untuk seluruh periode tidak sama dengan jumlah seluruh part yang diproses pada seluruh periode, sehingga jadwal yang dihasilkan adalah jadwal yang tidak layak karena untuk mengerjakan seluruh permintaan part pada seluruh periode akan melanggar saat  $t = 0$ .

## V. KESIMPULAN

Pada masalah *single item multi due date*, jumlah *batch*, ukuran *batch*, dan urutan pemrosesan *batch* untuk setiap interval waktu  $H_h$  diselesaikan dengan memandang satu interval waktu  $H_h$  sebagai permasalahan *common due date* dan dijadwalkan secara *backward*. Berdasarkan perhitungan dan analisis yang dilakukan dengan data hipotetik dapat disimpulkan:

- *Part* yang diminta pada satu periode akan dikerjakan pada periode yang sama dengan periode permintaan *part* jika untuk setiap periode terpenuhi kondisi  $(N_h - 1) s + N_h t + a_{n_h} \leq H_h$  dengan  $H_h$  adalah *dummy due date* yang didalamnya sudah memperhitungkan waktu *set up batch*.
- Pada  $h < r$  (interval waktu sebelum interval waktu terakhir ( $r$ )) jika kondisi  $(N_h - 1) s + N_h t + a_{n_h} \leq H_h$  tidak selalu terpenuhi untuk setiap periode maka *part* dapat dikerjakan pada periode setelahnya ( $H_{h+1}$ ) dan akan dikirim sesuai dengan *due date* dari *part* tersebut.
- Jadwal akan layak jika pada saat  $h = r$  (interval waktu terakhir) terpenuhi kondisi  $(N_h - 1) s + N_h t + a_{n_h} \leq H_h$  dengan  $n_h$  adalah jumlah *part* yang diminta pada periode terakhir ditambah dengan jumlah *part* yang tidak dapat diproses pada periode sebelumnya (jika ada).
- Total waktu tinggal aktual akan semakin besar jika terdapat *part* yang diproses pada periode yang tidak sesuai dengan periode permintaan *part* tersebut.

## DAFTAR PUSTAKA

- [1] Sukoyo., Samadhi, TMA.A., Iskandar, BP., Halim, A.H., "Model Penjadwalan *Batch Multi Item* dengan Dependent Processing Time", Jurnal Teknik Industri, Vol 12 no.2, 69-80. Desember, 2010.
- [2] Potts, C N., Kovalyov, M.Y., "Scheduling with *Batching*: a Review", European Journal of Operational Research, 120, 228-249. 2001.
- [3] Mahtirajan, M., Sivakumar, A.I., "a literatur review, Classification and Simple Meta Heuristic on Scheduling of *Batch processors* in Semiconductor", International Journal Advance Manufacturing Technology, 29, 990 – 1001. January, 2006
- [4] Hidayat, Nita P.A., Cakravastia, A., Samadhi, TMA.A., "A *Single item Batch Scheduling Model* on a *Batch processor* to Minimize Total actual Flow time of *Parts* Through The Shop", Proceedings of Asia Pasific Industrial Engineering & Management System Conference (APIEMS).
- [5] Halim, A.H., "Batch Scheduling for Production Under Just In Time Environment", Dissertation, University of Osaka Prefectur, 1993.
- [6] Dobson, Greogory., Nambimadom, Ramakrishnan S., "The *Batch Loading and Scheduling Problem*. Operation

Research", Operations Research. Vol 49 no.1, pp 52 – 56. (Januari – Februari ), 2001.

[7] Sukoyo., Samadhi, TMA Ari., Iskandar, Bermawi P., Halim, Abdul Hakim, "Batch scheduling for a single machine processing *parts* of a *single item* with increasing processing time to minimize total actual flow time", *Proceedings of the 9th Asia Pasific Industrial Engineering & Management Systems Conference*. December, 2008.

[8] Astuti, M.D., Halim, A.H., "Model Penjadwalan *Batch* pada *Batch Processor* dengan Waktu Proses yang Tidak Konstan", Prosiding Seminar nasional BKSTI pp II-175. September, 2014.

[9] Halim, A.H., Miyazaki, S., and Ohta, H., "Batch-scheduling Problems to Minimize Actual Flow times of *Parts* Through the Shop under JIT Environment", European Journal of Operational Research, 72, 529-544. 1994.