

# **PENJADWALAN PRODUKSI *FLEXIBLE FLOWSHOPS* DENGAN *SEQUENCE-DEPENDENT SETUP TIMES* MENGUNAKAN METODE RELAKSASI *LAGRANGIAN* (Studi Kasus pada PT. Cahaya Angkasa Abadi)**

**Herry Christian Palit, Tessa Vanina Soetanto**

Dosen Fakultas Teknologi Industri, Jurusan Teknik Industri, Universitas Kristen Petra

**Hermin Novianti**

Alumnus Fakultas Teknologi Industri, Jurusan Teknik Industri, Universitas Kristen Petra

## **ABSTRAK**

Artikel ini membahas penjadwalan yang mempunyai karakteristik *flexible flowshop* dengan *sequence-dependent setup*. Metode penyelesaian yang digunakan adalah relaksasi *Lagrangian* dengan menguraikan problem awal yang kompleks menjadi sebuah subproblem yang lebih sederhana.

**Kata-kunci:** *Flexible flowshops, sequence-dependent setup, relaksasi Lagrangian.*

## **ABSTRACT**

*The article discusses the problem of flexible flowshop scheduling with sequence-dependent setup. The Lagrangian relaxation method is used to solve the problem by decomposes an initial complex problem into simpler subproblem.*

**Keywords:** *Flexible flowshops, sequence-dependent setup, Lagrangian relaxation.*

## **1. PENDAHULUAN**

Penjadwalan *flowshop* dicirikan dengan adanya aliran proses yang searah untuk setiap jenis *job* yang dikerjakan. Metode yang digunakan biasanya mengasumsikan bahwa *setup* operasi independen terhadap urutan pengerjaan dan termasuk dalam waktu proses. Sedangkan dalam kondisi yang dijumpai di beberapa industri adalah adanya waktu *setup* mesin untuk setiap pergantian *job* yang berbeda, sehingga hal ini akan mempengaruhi urutan penjadwalan. Permasalahan akan makin kompleks bila dalam suatu operasi terdapat beberapa mesin yang digunakan (*flexible flowshop*).

Liu dan Chang (2000) menguraikan penggunaan metode relaksasi *Lagrangian* untuk menyelesaikan penjadwalan produksi *flexible flowshop* dengan *sequence dependent setup times* yang bertujuan untuk memenuhi *due dates customer*, mengurangi *work in process* (WIP), dan mengurangi biaya setup mesin.

Artikel ini memuat penjadwalan menggunakan metode yang dikembangkan Liu dan Chang (2000) di PT. Cahaya Angkasa Abadi yang bergerak di bidang produksi filamen dan *light in wire*.

## 2. PENJADWALAN *FLEXIBLE FLOWSHOP* DENGAN *SEQUENCE DEPENDENT SETUPS*

Penjadwalan *flowshop* adalah penjadwalan dimana setiap *job* melalui urutan proses atau operasi yang sama, sedangkan penjadwalan *flexible flowshop* memiliki konsep yang sama dengan *flowshop*, hanya perbedaannya terdapat pada setiap proses atau operasi yang memiliki sejumlah mesin identik yang disusun paralel. Operasi produksi dapat diproses oleh semua mesin yang identik dalam satu grup. *Flexible flowshop* dapat dilihat sebagai lingkungan manufaktur dengan multiproses dan multimesin.

Masalah penjadwalan berupa *sequence dependent setup* merupakan masalah yang terjadi pada beberapa sistem produksi. Ketika suatu mesin digunakan untuk memproses beragam tipe produk, maka bisa muncul waktu dan biaya setup mesin akibat adanya pergantian tipe produk. Bila setup mesin sering dilakukan dan waktunya cukup besar, maka tentunya akan berpengaruh terhadap penjadwalan, sehingga konsekuensinya kita harus mempertimbangkan pengaruh setup mesin dalam penjadwalan.

## 3. RELAKSASI *LAGRANGIAN*

Metode Relaksasi *Lagrangian* awalnya dirumuskan oleh Fisher (1981) untuk menyelesaikan masalah *integer programming*. Selanjutnya Liu dan Chang (2000) mengembangkan metode ini untuk menyelesaikan penjadwalan produksi *flexible flowshops* dimana efek *setup* berupa waktu dan biaya yang bergantung pada urutan proses itu signifikan. Pendekatan ini mengembangkan ide penjadwalan produksi *part-part* secara individual, tanpa mempertimbangkan efek *setup* mesin dan penjadwalan penggunaan mesin untuk memproses *part-part* yang ada. Kedua hal ini lebih sederhana daripada penjadwalan produksi *flexible flowshops* dengan efek *setup*.

Oleh karena *setup* setiap mesin harus cocok dengan pemrosesan *part*, dalam solusi yang layak untuk penjadwalan produksi *flexible flowshop*, maka penjadwalan produksi *part* dan penggunaan mesin harus disinkronisasi. Ide tersebut memotivasi pengembangan metode penyelesaian relaksasi *Lagrangian*.

Penjadwalan produksi *flexible flowshops* pertama diformulasikan sebagai permasalahan *integer programming* yang terpisah dengan sinkronisasi kendala di antara variabel penjadwalan produksi *part* dan penggunaan mesin.

Relaksasi *Lagrangian* kemudian diaplikasikan untuk sinkronisasi kendala-kendala, dan masalah penjadwalan dikelompokkan ke dalam dua kelas subproblem yaitu penjadwalan *part* dan mesin. Kendala yang kompleks direlaksasi dengan menggunakan *Lagrangian multipliers*, mengijinkan problem asli dikelompokkan menjadi subproblem yang dapat diselesaikan dengan lebih mudah. Problem ganda terbentuk dimana *multipliers* dioptimalkan melalui iterasi. *Lagrangian multipliers* mempunyai kelebihan yaitu menggabungkan fungsi tujuan utama dengan kendala-kendalanya menjadi satu fungsi saja, sehingga mempermudah dalam mencari solusinya.

Penjadwalan produksi *part* berfokus pada produksi *part* secara individual. *Precedence relationships*, jumlah permintaan, dan kebutuhan *due date* setiap pesanan dapat dengan mudah dipertimbangkan ketika menyelesaikan masalah penjadwalan dari aspek tersebut tanpa mempertimbangkan kendala kapasitas mesin dan *setup*.

Penjadwalan mesin berfokus pada mesin secara individual dan menentukan urutan pemrosesan tugas pada setiap mesin. Kendala kapasitas dan *sequence-dependent setup times* dari satu mesin adalah hal utama pada aspek ini.

Dalam masing-masing subproblem terdapat struktur aliran jaringan yang menggambarkan perubahan status mesin dan keseimbangan aliran produksi. Solusi ini mengaplikasikan biaya minimum dalam algoritma aliran jaringan untuk menyelesaikan subproblem secara individual dan mengadopsi metode *surrogate subgradient* yang efisien dari Luh untuk mengoptimalkan *Lagrangian multipliers*. Solusi problem ganda ini mungkin tidak layak dimana terdapat kendala yang tidak sinkron, maka heuristik pencarian mesin yang tersedia dikembangkan untuk mendapatkan solusi yang memenuhi semua kendala sinkronisasi.

Heuristik juga akan mencakup sinkronisasi dari problem ganda akan diambil sebagai solusi penjadwalan yang layak. Sedangkan untuk *part* yang harus dijadwalkan ulang karena tidak sinkron, maka waktu mesin yang tersedia untuk *part* ini dapat diputuskan berdasarkan *part* yang telah terjadwal dan hubungannya dengan waktu *setup*. *Part* yang harus dijadwalkan ulang ini mengubah waktu mesin yang tersedia. Untuk *part* yang belum terjadwal ini harus dijadwalkan ulang satu per satu. Berdasarkan observasi ini, algoritma menetapkan penjadwalan untuk kelayakan satu *part* pada satu waktu dengan eksploitasi struktur jaringan dari aliran produksi dan kendala penggunaan mesin dan interpretasi biaya marginal dari *Lagrangian multipliers*.

Secara umum, heuristik ini memanfaatkan interpretasi ekonomi dari *Lagrangian multipliers*, menggunakan struktur jaringan, dan kebijakan *slack time*. Jadi untuk *part* dengan aliran produksi yang tidak sinkron dengan *setup* mesin dapat menggunakan heuristik berikut ini:

1. Menentukan prioritas *part* dengan membandingkan *slack time*.
2. Mencari waktu mesin yang tersedia dan mengurutkan kembali aliran produksi dari *part* yang memiliki prioritas tertinggi dengan menyelesaikan model minimasi biaya aliran jaringan linear (MCLNF) untuk *part* dan memanfaatkan solusi relaksasi.

#### 4. FORMULASI PENJADWALAN

Novianti (2004), mengaplikasikan metode relaksasi Lagrangian untuk penjadwalan di PT. Cahaya Angkasa Abadi, yang memproduksi berbagai jenis filamen yang mempunyai aliran proses produksi yang sama. Jenis filamen yang terdapat di perusahaan adalah TL, SC, DC, AL, dan ANCHOR. Jenis *part* yang diteliti, volume produksi dan due dates dapat dilihat pada Tabel 1. Proses produksi yang berlaku di perusahaan memiliki buffer untuk setiap tahap proses dengan kapasitas yang tidak terbatas. Bahan baku diasumsikan tersedia pada awal proses.

Data *part* yang tidak terjadwal dan mengalami keterlambatan pada kondisi awal perusahaan dapat dilihat pada Tabel 2.

**Tabel 1. Jenis Filamen, Volume, dan Due Dates**

Indeks tipe (i)	Jenis filamen	Volume (unit)	Volume (Batch)	Due dates (menit)
1	TL 10W STD	670000	23	9000
2	TL 20W	910000	31	10800
3	TL 40W	610000	21	9000
4	TC 9412 W	90000	3	7200
5	TC 36W	635000	22	10800
6	FDL 27W	20000	1	5400
7	DC 6W	20000	1	5400

**Tabel 2. Data Part yang Tidak Terjadwal dan Terlambat pada Kondisi Awal**

Indeks tipe (i)	$\Sigma$ terjadwal (unit)	$\Sigma$ tak terjadwal (unit)	$\Sigma$ keterlambatan (unit)
1	435000	235000	315000
2	600000	310000	400000
3	250000	360000	90000
4	90000	0	45000
5	600000	35000	180000
6	20000	0	0
7	20000	0	0
Total	2015000	940000	1030000

Keterangan : 1 *batch* = 30000 unit

Setiap tahapan proses produksi memiliki sekumpulan mesin identik yang paralel dan kelompok mesin yang berbeda digunakan pada proses yang berbeda pula. Pesanan *customer* dapat dipisahkan untuk diproses pada mesin yang berbeda karena mesinnya identik untuk setiap proses. Satu mesin hanya dapat memproses satu tipe produk pada waktu yang sama dan mesin diasumsikan tidak mengalami kerusakan.

Waktu *setup* mesin yang berlaku pada proses produksi adalah signifikan terhadap waktu prosesnya. Hal ini bergantung pada setiap pergantian *part* yang diproses di satu mesin pada satu proses, sehingga kondisi awal mesin berupa kesiapan untuk memproses tipe tertentu harus diketahui.

Proses produksi yang berlaku memiliki *buffer* pada setiap proses dengan kapasitas tidak terbatas, sehingga *buffer* menjadi kendala yang tidak terlalu signifikan. Bahan baku diasumsikan selalu tersedia pada awal proses. Setiap pesanan hanya satu tipe *part* dengan *due date* yang telah ditetapkan.

Berdasarkan kondisi di atas serta berpegang pada metode relaksasi *Lagrangian* untuk kasus *flexible flowshops*, dibuat formulasi masalah penjadwalan *flexible flowshops* dengan *sequence-dependent setup* dengan sejumlah notasi sebagai berikut:

1. Variabel input terdiri dari:

$T$  adalah horizon waktu penjadwalan,  $t$  menyatakan index waktu,  $t=1, \dots, T$ .

$I$  menyatakan jumlah total tipe *parts*,  $i$  adalah index tipe *part*,  $i=1, \dots, I$ .

$J$  adalah jumlah total proses,  $j$  adalah index proses,  $j=1, \dots, J$ .

$K$  sebagai jumlah total pesanan,  $k$  adalah index pesanan,  $k=1, \dots, K$ .

$E_k$  adalah tipe *part* yang dibutuhkan oleh pesanan- $k$ ,  $D_k$  jumlah yang dibutuhkan oleh tipe *part*  $E_k$  oleh pesanan- $k$ .

$r_{kt}$ : kuantitas yang sudah ada dari pesanan- $k$  pada waktu- $t$  dengan  $\sum_{t=1}^T r_{kt} = D_k$

$a_k$  adalah koefisien penalti *earliness* dari pesanan- $k$ ,  $b_k$  merupakan koefisien penalti dari pesanan- $k$ .

$h_{kj}$  adalah *holding cost* per unit dari pesanan- $k$  dalam *buffer* pada proses- $j$ ,  $d_k$  adalah *due date* pesanan- $k$ .

$M$  adalah jumlah total mesin,  $M_j$  kumpulan mesin untuk proses- $j$ ,  $j = 1, \dots, J$ .

$m$  index mesin,  $m = 1, \dots, M$ , dan  $C_{lij}$  adalah biaya *setup* mesin dalam grup  $M_j$  dari pemrosesan *parts* tipe- $l$  ke pemrosesan *parts* tipe- $i$ .

$P_{ij}$  merupakan waktu pemrosesan *part* tipe- $i$  pada proses- $j$ , dan  $A_{lij}$  adalah waktu *setup* mesin pada grup  $M_j$  dari pemrosesan *parts* tipe- $l$  ke pemrosesan *parts* tipe- $i$ .

2. Variabel *intermediate* terdiri dari:

$$\zeta_{it}^m = \begin{cases} 1, & \text{jika mesin } m \text{ kosong dan siap utk memproses part tipe- } i \text{ pada waktu } t \\ 0, & \text{jika sebaliknya} \end{cases}$$

$B_{kjt}$  adalah tingkat *buffer* proses- $j$  dari *part* yang dibutuhkan order- $k$  pada awal periode  $t$ , dan  $O_{kt}$  adalah jumlah *part* dari order- $k$  yang sudah selesai pada awal periode  $t$ .

3. Variabel keputusan terdiri dari:

$$u_{kjt}^m = \begin{cases} 1, & \text{jika suatu part dari order } k \text{ siap diproses pada proses } j \text{ oleh mesin } m \text{ pada awal periode } t \\ 0, & \text{jika sebaliknya} \end{cases}$$

$$\alpha_{it}^m = \begin{cases} 1, & \text{jika mesin } m \text{ mulai memproses part tipe } i \text{ pada waktu } t \\ 0, & \text{jika sebaliknya} \end{cases}$$

$$\xi_{lit}^m = \begin{cases} 1, & \text{jika mesin } m \text{ mulai setup dari proses part tipe } l \text{ ke tipe } i \text{ pada waktu } t \\ 0, & \text{jika sebaliknya} \end{cases}$$

Formulasi masalah akan memiliki sejumlah kendala berikut ini:

1. Pada periode waktu  $t$ , mesin  $m$  yang merupakan bagian dari  $M_j$  siap untuk memproses *part* tipe- $i$ , maka mesin harus berada pada salah satu dari ketiga kondisi di bawah ini sebelum waktu  $t$ , yaitu (1) mesin siap memproses *part* tipe- $i$  pada waktu  $t - P_{ij}$ , (2) mesin kosong tetapi siap untuk memproses *part* tipe- $i$  pada waktu  $t - 1$ , atau (3) mesin memulai *setup* pada waktu  $t - A_{lij}$  untuk memproses *part* tipe- $i$  dari sebelumnya memproses *part* tipe- $l$ .

Hubungan di atas dapat diekspresikan sebagai rumus keseimbangan penggunaan mesin (*machine usage balance equations*):

$$\alpha_{i(t-P_{ij})}^m + \zeta_{i(t-1)}^m + \sum_{l=1, l \neq i}^I \xi_{li(t-A_{lij})}^m = \alpha_{it}^m + \zeta_{it}^m + \sum_{l=1, l \neq i}^I \xi_{lit}^m \quad \forall i, m, 1 \leq t \leq T \quad (1)$$

dengan  $\{ \alpha_{it}^m = 0, \forall i, j, m \in M_j \text{ dan } 1 - P_{ij} \leq t \leq 0 \}$ ,

$\{ \xi_{lit}^m = 0, \forall i, l, j, m \in M_j \text{ dan } 1 - A_{lij} \leq t \leq 0 \}$  dan  $\{ \zeta_{i0}^m, \forall i, m \}$ .

2. Pada persamaan (1), mesin diasumsikan berawal dari kosong tetapi siap untuk memproses satu jenis tipe *part* sehingga:
- $$\sum_{i=1}^J \zeta_{i0}^m = 1 \quad \forall m \quad (2)$$

karena satu mesin hanya siap untuk satu tipe produk pada setiap waktunya. Kendala (1) dan (2) di atas mengacu pada kendala kapasitas.

3. Dalam aliran produksi, sebuah *part* dari order-*k* (tipe- $E_k$ ) dimasukkan dari *buffer* *j*-1 ke dalam mesin  $m \in M_{j-1}$  untuk proses *j*-1, pada waktu  $t - P_{E_k(j-1)} + 1$  masuk dalam *buffer*-*j* setelah periode proses  $P_{E_k(j-1)}$ . Aliran produksi untuk *part* suatu order dapat dideskripsikan sebagai rumus keseimbangan aliran *part* (*part flow balance equations*).

$$B_{k1(t+1)} = B_{k1t} - \sum_{m \in M_1} u_{k1t}^m + r_{k1t} \quad \forall k, t \quad (3)$$

$$B_{kj(t+1)} = B_{kj t} - \sum_{m \in M_j} u_{kj t}^m + \sum_{m \in M_j} u_{kj(t-P_{E_k j}+1)}^m \quad (4)$$

$$B_{kj t} \geq 0 \quad \forall k, j, t \quad (5)$$

4. Jumlah *part* dari order-*k* yang terselesaikan pada periode *t* adalah:

$$O_{kt} = \sum_{m \in M_j} u_{kj(t-P_{E_k j})}^m \quad (6)$$

5. Keputusan untuk memproses *part* harus sinkron dengan keputusan penggunaan mesin yang ditunjukkan dengan kendala sinkronisasi di bawah ini:

$$\sum_{k=1, E_k=i}^K u_{kjt}^m = \alpha_{it}^m \quad \forall j, m, i, t \quad (7)$$

Kendala sinkronisasi ini juga mengacu kepada kendala kapasitas seperti pada kendala (1) dan (2).

6. Salah satu tujuan penjadwalan adalah memenuhi target waktu *customer* (*due date*) secara tepat waktu. Penalti keterlambatan dan kecepatan untuk tiap *part* order-*k* yang selesai pada waktu *t* didefinisikan sebagai:

$$\begin{cases} a_k(d_k - w - t), & \text{if } t < d_k - w \\ 0, & \text{if } d_k - w \leq t \leq d_k \\ b_k(t - d_k) & \text{if } t > d_k \end{cases} \quad (8)$$

Dimana *w* adalah konstanta yang mewakili rentang waktu dimana *part* dapat dijadwalkan bebas penalti. Koefisien  $a_k$  dan  $b_k$  dapat berbeda antar order untuk menggambarkan kepentingannya masing-masing. Biaya penalti diformulasikan sebagai berikut:

$$v_k = a_k \sum_{t=1}^{d_k-w} (d_k - w - t) O_{kt} + b_k \sum_{t=d_k+1}^T (t - d_k) O_{kt} \quad \forall k \quad (9)$$

Tujuan yang ingin dicapai dari penjadwalan ini adalah pengurangan WIP, waktu *setup* mesin, dan memenuhi *due date customer*, dimana dinyatakan dalam biaya total produksi yang minimum, sehingga masalahnya adalah menentukan jadwal yang meminimalkan biaya total produksi dan memenuhi semua kendala sistem. Masalah ini diformulasikan sebagai berikut:

$$\min_{u, \alpha, \xi} \sum_{k=1}^K \left( v_k + \sum_{t=1}^T \sum_{j=1}^J h_{kj} B_{kjt} \right) + \sum_{j=1}^J \sum_{m \in M_j} \sum_{i=1}^I \sum_{l=1, l \neq i}^I \sum_{t=1}^T C_{lij} \xi_{lit}^m \quad (10)$$

dengan kendala (1)–(7), dimana syarat pertama adalah penalti, kedua adalah inventori yang tertahan dalam proses, dan ketiga adalah *setup* mesin.

Dari pemodelan masalah di atas dilakukan adopsi pendekatan yang berdasarkan relaksasi *Lagrangian* dan optimasi aliran jaringan. Relaksasi *Lagrangian* menyederhanakan kendala sinkronisasi (7) pada problem (10) dengan menggunakan *Lagrangian multipliers* yang dirumuskan berikut ini:

$$\sum_{k=1}^K \left( v_k + \sum_{t=1}^T \sum_{j=1}^J h_{kj} B_{kjt} \right) + \sum_{j=1}^J \sum_{m \in M_j} \sum_{i=1}^I \sum_{l=1, l \neq i}^I \sum_{t=1}^T C_{lij} \xi_{lit}^m + \sum_{j=1}^J \sum_{m \in M_j} \sum_{i=1}^I \sum_{t=1}^T \pi_{imt} \left( \sum_{k=1, E_k=1}^K u_{kjt}^m - \alpha_{it}^m \right) \quad (11)$$

dimana  $\{\pi_{imt}\}$  adalah *Lagrangian multipliers*.

Problem (10) kemudian didekomposisi menjadi subproblem penjadwalan produk dan mesin yang independen, dimana *Lagrangian multipliers* mendefinisikan untuk order- $k$  sebagai:

$$PS_k(u_k, \pi) \equiv \left( v_k + \sum_{t=1}^T \sum_{j=1}^J h_{kj} B_{kjt} \right) + \sum_{t=1}^T \sum_{j=1}^J \sum_{m \in M_j} \pi_{E_k mt} u_{kjt}^m \quad (12)$$

selain itu juga mendefinisikan untuk mesin  $m$  sebagai:

$$MS_m(\xi_m, \alpha_m, \pi) \equiv \sum_{t=1}^T \left( \sum_{i=1}^I \sum_{l=1, l \neq i}^I C_{lij} \xi_{lit}^m - \sum_{i=1}^I \pi_{imt} \alpha_{it}^m \right) \quad (13)$$

Problem dual untuk persamaan (11) setelah relaksasi *Lagrangian* dapat didefinisikan sebagai berikut:

$$\max_{\pi} \left\{ \Phi(\pi) \equiv \sum_{k=1}^K \min_{u_k} PS_k(u_k, \pi) + \sum_{j=1}^J \sum_{m \in M_j} \min_{\xi_m, \alpha_m} MS_m(\xi_m, \alpha_m, \pi) \right\} \quad (14)$$

dengan subjek kendala (1)-(5).

Rumus keseimbangan aliran *part* dan rumus keseimbangan penggunaan mesin memberikan suatu struktur aliran jaringan untuk masing-masing subproblem. Melalui *Lagrangian multipliers*, masing-masing subproblem diselesaikan sebagai problem *Minimum Cost Linear Network Flow* (MCLNF). Problem MCLNF ini diselesaikan dengan menempatkan *part* yang diinginkan pada mesin yang dapat menyelesaikan *part* tersebut sesuai dengan *due date*, *part* ditempatkan pada mesin yang memberikan waktu *setup* terkecil, dan mesin yang memberikan *buffer* terkecil.

*Lagrangian multipliers* kemudian diperbaharui oleh Zhao dkk. (1997) menggunakan iterasi dengan metode *surrogate subgradient*, metode ini efisien jika terdapat banyak subproblem. Misal  $u \equiv \{u_k, k=1, \dots, K\}$ ,  $\alpha \equiv \{\alpha_m, m=1, \dots, M\}$ , dan  $\xi \equiv \{\xi_m, m=1, \dots, M\}$  adalah solusi untuk subproblem (PS- $k$ ) dan (MS- $m$ ) setelah menyelesaikan satu subproblem dengan *Lagrangian multipliers* ( $\pi^s$ ) pada iterasi ke- $s$ . *Surrogate subgradient* untuk fungsi ganda *surrogate*  $\Phi(\pi)$  dengan *Lagrangian multipliers*  $\pi_{imt}$  adalah:

$$\tilde{g}_{imt}(u, \alpha, \xi) \equiv \frac{\partial}{\partial \pi_{imt}} \Phi(\pi) = \sum_{k=1, E_k=i}^K u_{kjt}^m - \alpha_{it}^m \quad \forall i, m, t \quad (15)$$

dan SSG menyesuaikan *Lagrangian multipliers*  $\pi$  sebagai

$$\pi_{imt}^{s+1} = \pi_{imt}^s + \beta^s \tilde{g}_{imt}(u, \alpha, \xi) \quad \forall i, m, t \quad (16)$$

dengan nilai  $\pi$  awal sama dengan 0 dan  $\beta$  adalah besar langkah pada iterasi ke- $s$  yang ditentukan oleh:

$$\beta = \frac{\gamma \left[ \Phi^* - \tilde{\Phi}^s \right]}{\left\| \tilde{g}^s \right\|^2} \quad (17)$$

dengan  $0 < \gamma < 1$ ,  $\Phi^*$  mengukur biaya ganda optimal yang hasilnya dihitung melalui perkalian nilai  $\tilde{\Phi}^s$  dengan 1,1. Biaya ganda *surrogate* pada iterasi ke- $s$  dinotasikan dengan  $\Phi^s$ .

### 5. ANALISA JADWAL

Sebelum mendapatkan hasil penjadwalan terdapat sejumlah filamen TL yang belum terjadwal karena tidak memenuhi kendala sinkronisasi. Sejumlah filamen tersebut harus melalui heuristik untuk mendapatkan hasil penjadwalan yang layak, dalam arti penjadwalan part dan mesin harus sinkron. Semua part yang terdapat dalam perencanaan produksi dapat terjadwalkan atau dapat diproses seluruhnya, kecuali tipe TL 40W, TC 9412W, FDL 27W, dan DC 6W yang secara keseluruhan terdapat 10 batch yang tidak terjadwal dari 102 batch yang hendak dijadwalkan sesuai dengan perencanaan produksi.

Semua part dapat diselesaikan sesuai dengan *due date* atau tidak mengalami keterlambatan pengiriman ke *customer*. Adapun perbaikan yang dicapai dengan menggunakan metode ini dibandingkan dengan kondisi awal dapat dilihat pada tabel 3.

**Tabel 3. Perbandingan Performansi**

Karakteristik	Kondisi awal	Metode <i>Lagrangian</i>	% perbaikan
Jumlah tak terjadwal (batch)	32	10	68,1
Jumlah keterlambatan	35	0	100
Jumlah setup	173	146	15,6

### 6. KESIMPULAN

Metode relaksasi *Lagrangian* memberikan kemudahan di dalam menyelesaikan masalah penjadwalan produksi *flexible flowshops* melalui penyederhanaan kendala sinkronisasi yang kompleks, sehingga problem dapat diselesaikan dengan lebih mudah daripada problem awalnya. Berdasarkan hasil perhitungan yang telah dilakukan, metode relaksasi *Lagrangian* memberikan perbaikan performa dalam hal pengurangan jumlah

*part* yang tak terjadwal, pemenuhan *due dates* secara tepat waktu, serta pengurangan waktu *setup* dan *buffer cost* secara signifikan.

Melalui perhitungan *buffer cost* dalam pemodelan masalah berdasarkan metode relaksasi *Lagrangian*, dapat diketahui *bottleneck* terjadi pada proses mana saja. Hal ini dapat digunakan untuk mempertimbangkan jumlah mesin yang ada itu sudah cukup, kekurangan, atau kelebihan.

#### DAFTAR PUSTAKA

- Liu, C. Y., and S. C. Chang, 2000. "Scheduling Flexible Flow Shop with Sequence-Dependent Setup Effects", *IEEE Transactions on Robotics and Automation*, Vol. 16, 408 – 419.
- Fisher, M.L. 1981. "Lagrangian Relaxation Method for Solving Integer Programming Problem", *Management Science*, Vol. 27, 1 - 18
- Novianti, H., 2004, "Penjadwalan Produksi Flexible Flowshops dengan Sequence-Dependent Setup Times di PT Cahaya Angkasa Abadi", *Tugas Akhir Jurusan Teknik Industri*, No. 01/0779/IND/2004, Universitas Kristen Petra, Surabaya.
- Zhao, X., P.B. Luh, and J. Wang, 1997. "Surrogate Gradient Algorithm for Lagrangian Relaxation", *IEEE Conference of Decision and Control*, Vol. 36, 305–310.