

# Perbandingan Metode *Ranked Positional Weight* dan *Kilbridge Wester* Pada Permasalahan Keseimbangan Lini Lintasan Produksi Berbasis *Single Model*

Dyah Saptanti Perwitasari

Teknik Informatika ITB, Bandung 40135, email: if14017@students.if.itb.ac.id

**Abstract** – Permasalahan keseimbangan lini muncul sebagai akibat dari adanya kebutuhan untuk mendapatkan lini perakitan yang terbaik. Berbagai pendekatan heuristik untuk mengatasi permasalahan keseimbangan lini telah dikembangkan. Beberapa diantaranya adalah *Ranked Positional Weight* dan *Kilbridge Wester*. Berdasarkan banyak penelitian disebutkan bahwa kedua metode ini cukup bagus dibandingkan metode lainnya. Meski demikian, tidak ada penelitian yang membuktikan metode mana yang lebih baik.

Pada Tugas Akhir ini dibangun perangkat lunak “anTarmuka” untuk membandingkan metode *Ranked Positional Weight* dan *Kilbridge Wester*. Perangkat lunak “anTarmuka” dikembangkan dengan menggunakan bahasa pemrograman Java pada framework JDK 1.5.0. Untuk perbandingan digunakan beberapa komponen pengukuran, antara lain performansi, Efisiensi Lini, Balance Delay, dan Smoothness Index.

Berdasarkan hasil pengujian, perangkat lunak dapat menghasilkan lini perakitan yang valid. Pada akhir pelaksanaan Tugas Akhir ini, diambil beberapa kesimpulan terkait hasil perbandingan kedua metode tersebut. Dari segi performansi, metode *Ranked Positional Weight* lebih unggul daripada *Kilbridge Wester*. Sedangkan dari segi algoritma, sulit untuk menyimpulkan metode mana yang lebih baik karena hasil pembangunan lini oleh kedua metode ini hampir sama.

Kata kunci : keseimbangan lini, lini perakitan, *Ranked Positional Weight*, *Kilbridge Wester*.

## 1. PENDAHULUAN

Suatu badan usaha yang menghasilkan produk tertentu memiliki sistem produksi. Sistem produksi tersebut merupakan suatu sistem yang bertujuan untuk mengelola berbagai proses untuk menghasilkan produk yang bersangkutan. Terdapat beberapa hal penting yang harus dipahami oleh suatu badan usaha sebelum merancang sistem produksi. Salah satunya adalah dalam hal perancangan tata ruang (*layout planning*). Penentuan lintasan perakitan adalah salah satu dari aspek-aspek yang dirancang dalam perancangan tata ruang. Suatu lintasan perakitan terdiri dari beberapa stasiun kerja, dan setiap stasiun

kerja terdiri dari minimal satu *task*. Keseimbangan lini merupakan suatu permasalahan yang harus dihadapi dalam pembangunan suatu lintasan perakitan. Tujuan keseimbangan lini (*line balancing*) pada kasus ini adalah untuk menentukan jumlah stasiun kerja yang seminimal mungkin dengan memperhatikan urutan antar-*task* dan waktu siklus sehingga batasan keterhubungan terpenuhi dan waktu stasiun tidak melebihi waktu siklus [BOY06]. Semakin sedikit jumlah stasiun kerja kebutuhan ruang akan semakin sedikit.

Berdasarkan jumlah produk yang dihasilkan, pemodelan keseimbangan lini dapat dibagi menjadi dua macam, yaitu *Single* dan *Multi Model*. *Single Model* digunakan untuk kegiatan produksi yang menghasilkan sebuah produk [CHS04]. Pada *Multi Model*, dua atau lebih produk yang berbeda dihasilkan pada mesin yang terpisah secara bersamaan, tanpa adanya perubahan jumlah mesin selama pergantian proses pengolahan produk-produk tersebut [CHS04].

Beberapa pendekatan terkait keseimbangan lini telah dilakukan, yaitu menggunakan teknik heuristik. Beberapa teknik heuristik pada produksi produk yang berjenis *Single Model* adalah *Ranked Positional Weight* (RPW) dan *Kilbridge Wester*. Metode RPW yang berbasis akumulasi waktu penyelesaian *task* ini merupakan metode yang dapat menemukan solusi dengan cepat [MAL00].

Pada Tugas Akhir ini akan dilakukan perbandingan metode RPW dan *Kilbridge Wester* untuk diterapkan pada persoalan keseimbangan lini untuk model tunggal (*Single Model*) dengan menerima masukan berupa waktu siklus, keterhubungan antar-*task*, dan waktu proses setiap *task*, serta menghasilkan keluaran dalam bentuk simulasi metode RPW dan *Kilbridge Wester*.

## 2. LANDASAN TEORI

### 2.1. Perencanaan Tata Ruang

Perencanaan tata ruang (*layout planning*) adalah salah satu aspek penting yang diperlukan dalam perancangan sistem produksi. Perencanaan tata ruang dilakukan pada awal ketika suatu sistem produksi akan dibangun maupun pada saat pengembangan produksi. Perencanaan tata ruang yang dilakukan pada awal produksi bertujuan untuk menghasilkan tata ruang

yang baik karena tata ruang mempengaruhi aspek lain. Sedangkan perencanaan yang dilakukan ketika sistem produksi yang telah ada sebelumnya bertujuan untuk meningkatkan tata fasilitas yang lebih baik untuk meningkatkan produktivitas kerja. Perencanaan tata ruang yang buruk dapat mempengaruhi produktivitas, misalnya tingkat produksi menjadi rendah karena penataan tata ruang yang tidak tepat.

*Task* adalah unit pekerjaan terkecil [SLG07], yaitu tahap dalam pemrosesan sebuah produk. *Task* diberikan kepada stasiun kerja yang diberikan tanggung jawab untuk melaksanakan tahapan pekerjaan tertentu. Setiap *task* mempunyai waktu pemrosesan masing-masing. Contoh pendefinisian *task* pada industri perakitan mobil [MAL00], mulai dari pemasangan ban hingga pemberian lapisan cat pada mobil, ditunjukkan pada Tabel 1.

**Tabel 1 Contoh Task**

Task	Aktivitas	Waktu Proses (menit)	Task Pendahulu
A	Pemasangan poros roda depan	20	-
B	Pemasangan batangan kipas	6	A
C	Pemasangan penutup kipas	5	B
D	Pemasangan poros roda depan	21	-
E	Pemasangan roda	8	-
F	Pemasangan jendela	35	-
G	Pemasangan roda gigi	15	C, D
H	Pemasangan pengatur jarak pada roda gigi	10	G
I	Pengencangan roda depan	15	E, H
J	Pemasangan mesin	5	C
K	Pemasangan kap mobil	46	F, I, J
L	Pelapisan body mobil	16	K

Stasiun kerja adalah tempat eksekusi *task-task* yang diberikan pada stasiun kerja yang bersangkutan [SLG07]. Stasiun mempunyai waktu stasiun, yaitu total waktu proses seluruh *task* yang terdapat pada stasiun kerja tersebut. Eksekusi yang dilakukan di setiap stasiun kerja dilakukan secara terurut, misalnya jika  $S_1$  adalah stasiun kerja pertama,  $S_2$  adalah stasiun kerja kedua, dan  $S_n$  adalah stasiun kerja ke- $n$ , maka pemrosesan sebuah produk dilakukan sesuai urutan stasiun, yaitu  $S_1$  hingga  $S_n$ . Jumlah minimal stasiun kerja yang mungkin dapat dibentuk berdasarkan waktu siklus adalah sebagai berikut [SLG07]:

$$M \geq \left( \frac{1}{C} \sum_{i=1}^N (t_i) \right) \times 100\% \quad (1)$$

dengan :

M = jumlah stasiun

C = waktu siklus

$t_i$  = waktu proses *task* ke- $i$

N = jumlah *task*

$i = 1, 2, 3, \dots, N$

## 2.2. Keseimbangan Lini

Keseimbangan lini adalah permasalahan pemberian *task* kepada stasiun kerja sehingga pembagian *task* merata (seimbang) dengan mempertimbangkan beberapa batasan [SLG07]. Keseimbangan lini sangat penting karena akan menentukan aspek-aspek lain dalam sistem produksi dalam jangka waktu yang cukup lama. Beberapa aspek yang terpengaruh antara lain biaya, keuntungan, tenaga kerja, peralatan, dan sebagainya. Keseimbangan lini ini digunakan untuk mendapatkan lintasan perakitan yang memenuhi tingkat produksi tertentu [SLG07]. Dengan demikian penyeimbangan lini harus dilakukan dengan metode yang tepat sehingga menghasilkan keluaran berupa keseimbangan lini yang terbaik. Batasan yang harus diperhatikan dalam pemberian *task* yaitu *precedence constraint* dan waktu siklus produksi.

### 1. Precedence Constraint

*Precedence constraint* adalah batasan kebergantungan suatu *task* terhadap *task* lain, dalam hal ini mengenai keterhubungan antar-*task*. Pemberian *task* kepada stasiun kerja harus memperhatikan apakah *task* tersebut ditempatkan pada stasiun kerja yang tepat sehingga tidak melanggar *precedence constraint*.

### 2. Waktu Siklus

Waktu siklus yaitu waktu yang dialokasikan kepada setiap stasiun kerja untuk mengeksekusi seluruh *task* yang diberikan kepada stasiun kerja tersebut. *Production rate* atau tingkat produksi adalah jumlah produk yang ingin dihasilkan (Q) dalam jangka waktu tertentu (T). Jika waktu proses sebuah *task* melebihi waktu siklus, artinya dengan lintasan perakitan yang sekarang ada tidak dapat mencapai tingkat produksi seperti yang diharapkan.

## 2.3. Ranked Positional Weight

*Ranked Positional Weight* adalah metode yang diusulkan oleh Helgeson dan Birnie sebagai pendekatan untuk memecahkan permasalahan pada keseimbangan lini dan menemukan solusi dengan cepat [MAL00]. Konsep dari metode ini adalah menentukan jumlah stasiun kerja minimal dan melakukan pembagian *task* ke dalam stasiun kerja dengan cara memberikan bobot posisi kepada setiap *task* sehingga semua *task* telah ditempatkan kepada sebuah stasiun kerja. Bobot setiap *task*, misal *task* ke- $i$  dihitung sebagai waktu yang dibutuhkan untuk melakukan *task* ke- $i$  ditambah dengan waktu untuk

mengeksekusi semua *task* yang akan dijalankan setelah *task* ke-*i* tersebut.

Urutan langkah-langkah pada metode *Ranked Positional Weight* adalah sebagai berikut [HAH03]:

1. Lakukan penghitungan bobot posisi untuk setiap *task*. Bobot posisi setiap *task* dihitung dari bobot suatu *task* ditambah dengan bobot *task-task* setelahnya.
2. Lakukan pengurutan *task-task* berdasarkan bobot posisi, yaitu dari bobot posisi besar ke bobot posisi kecil.
3. Tempatkan *task* dengan bobot terbesar ke sebuah stasiun kerja sepanjang tidak melanggar *precedence constraint* dan waktu stasiun kerja tidak melebihi waktu siklus.
4. Lakukan langkah 3 hingga semua *task* telah ditempatkan kepada suatu stasiun kerja.

#### 2.4. Kilbridge Wester

*Kilbridge Wester* adalah metode yang dirancang oleh M.Kilbridge dan L.Wester sebagai pendekatan lain untuk mengatasi permasalahan keseimbangan lini. Pada metode ini, dilakukan pengelompokan *task-task* ke dalam sejumlah kelompok yang mempunyai tingkat keterhubungan yang sama. Langkah-langkah yang digunakan metode *Kilbridge Wester* adalah sebagai berikut [HAH03]:

1. Lakukan pengelompokan beberapa *task* ke dalam kelompok yang sama. Misalnya Kelompok ke-*i* berisi *task-task* yang tidak mempunyai *task* pendahulu, Kelompok ke-*i*+1 berisi *task-task* yang mempunyai *task* pendahulu di Kelompok ke-*i*, Kelompok ke-*i*+2 berisi *task-task* yang mempunyai *task* pendahulu di Kelompok ke-*i*+1, dan sebagainya hingga semua *task* telah dimasukkan ke suatu kelompok.
2. Lakukan penempatan *task-task* di suatu kelompok, dalam hal ini mula-mula Kelompok 1, ke dalam sebuah stasiun kerja yang sama, ambil hasil penggabungan terbaik, yaitu waktu total semua *task* mendekati atau sama dengan waktu siklus. Jika penempatan sebuah *task* ke dalam stasiun kerja menyebabkan waktu total semua *task* yang berada di stasiun kerja bersangkutan melebihi waktu siklus, maka *task* tersebut ditempatkan di stasiun kerja yang berikutnya. Hapus *task-task* yang telah ditempatkan dari kelompok yang bersangkutan.
3. Jika terdapat beberapa *task-task* yang belum ditempatkan di suatu stasiun kerja dan waktu totalnya berjumlah kurang dari waktu siklus, lanjutkan penggabungan dengan *task* di setelahnya, dalam hal ini Kelompok 2.
4. Lakukan kembali langkah 2 dan 3 hingga semua *task* telah tergabung dalam suatu stasiun kerja.

#### 2.5. Utilisasi

Untuk mengukur hasil suatu lini perakitan, digunakan pengukuran utilisasi. Pengukuran utilisasi dilakukan

dengan cara melakukan pengukuran Efisiensi Lini dan *Balance Delay*, dan *Smoothness Index*.

##### 1. Efisiensi Lini

Yaitu tingkat efisiensi stasiun kerja rata-rata pada suatu lini perakitan. Semakin mendekati waktu siklus, efisiensi suatu lini semakin bagus. Efisiensi Lini dihitung dari presentase total waktu proses setiap *task* dibandingkan dengan total waktu siklus di semua stasiun kerja. Efisiensi Lini dinotasikan dengan [HAH03]:

$$EL = \left( \frac{1}{MC} \sum_{i=1}^N t_i \right) \times 100\% \quad (2)$$

dengan :

M = jumlah stasiun  
C = waktu siklus  
 $t_i$  = waktu proses *task* ke-*i*  
N = jumlah *task*  
 $i = 1, 2, 3, \dots, N$   
EL = Efisiensi Lini

##### 2. Balance Delay (BD)

*Balance Delay* merupakan jumlah waktu menganggur suatu lini perakitan karena pembagian kerja antarstasiun yang tidak merata [KIW61]. Penghitungan *Balance Delay* dilakukan dengan menggunakan persamaan seperti berikut ini [KIW61]:

$$BD = \frac{C - \bar{S}}{C} \times 100\% \quad (3)$$

dengan :

C = waktu siklus  
 $\bar{S}$  = waktu stasiun rata-rata

Untuk stasiun kerja sejumlah M, persamaan BD dapat juga dituliskan seperti berikut ini :

$$BD = \frac{MC - M\bar{S}}{MC} \times 100\% \quad (4)$$

atau :

$$BD = \frac{MC - \sum_{i=1}^N t_i}{MC} \times 100\% \quad (5)$$

dengan selisih total waktu siklus di semua stasiun dan total waktu stasiun pasti positif :

$$MC - \sum_{i=1}^N t_i \geq 0 \quad (6)$$

dan :

M = jumlah stasiun  
C = waktu siklus  
 $\bar{S}$  = waktu stasiun rata-rata  
 $t_i$  = waktu proses *task* ke-*i*  
N = jumlah *task*  
 $i = 1, 2, 3, \dots, N$

Dengan kata lain, *Balance Delay* dapat

didefinisikan sebagai :

$$BD = 100\% - EL \quad (7)$$

dengan :

EL = Efisiensi Lini  
BD = Balance Delay

### 3. Smoothness Index

Yaitu cara untuk mengukur tingkat waktu tunggu relatif dari suatu lini perakitan [KSS07]. Semakin mendekati nol nilai *Smoothness Index* suatu lini, maka semakin seimbang suatu lini, artinya *pembagian task-task* cukup merata. Lini dikatakan mempunyai keseimbangan sempurna jika nilai *Smoothness Index* nol [HAH03]. *Smoothness Index* dinotasikan sebagai berikut [KSS07] :

$$SI = \sqrt{\sum_{i=1}^M (C - S_i)^2} \quad (8)$$

dengan :

SI = Smoothness Index  
M = jumlah stasiun  
C = waktu siklus  
S<sub>i</sub> = waktu stasiun kerja ke-i  
i = 1, 2, 3, ..., N

## 3. ANALISIS MASALAH

### 3.1. Analisis Masukan

Untuk menentukan waktu siklus, diperlukan masukan berupa tingkat produksi. Waktu siklus berbanding terbalik dengan tingkat produksi yang diharapkan. Semakin tinggi tingkat produksi, semakin rendah waktu siklus yang dialokasikan di setiap stasiun kerja.

Waktu siklus yang terlalu rendah dapat menyebabkan keseimbangan lini tidak akan pernah tercapai. Misalnya ketika waktu proses *task* terkecil lebih besar dibandingkan dengan waktu siklus yang dialokasikan kepada stasiun kerja yang bersangkutan. Hal ini menyebabkan tak ada satu *task* pun yang dapat diberikan ke suatu stasiun kerja karena tidak memenuhi batasan keseimbangan lini.

Pada keseimbangan lini, sebuah *task* dapat diberikan kepada sebuah stasiun kerja jika dengan adanya penambahan *task* tersebut tidak membuat total waktu proses *task-task* di stasiun kerja tersebut melebihi waktu siklus tertentu. Dengan demikian waktu siklus masukan harus lebih besar atau sama dengan waktu proses *task* yang terbesar. Hal ini dapat dinotasikan sebagai berikut :

$$C \geq t_{i\text{maks}} \quad (8)$$

dengan :

C = waktu siklus  
t<sub>i maks</sub> = waktu proses *task* terbesar

Selain waktu siklus, masukan dapat berupa tingkat produksi (*production rate*). Tingkat produksi

berbanding terbalik dengan waktu siklus. Dengan kata lain, tingkat produksi harus lebih besar atau sama dengan satu per waktu proses *task* yang terbesar.

$$TP \leq \frac{1}{t_{i\text{maks}}} \quad (9)$$

dengan :

TP = tingkat produksi  
t<sub>i maks</sub> = waktu proses *task* terbesar

Keterhubungan antar-*task* atau *precedence constraint* merupakan salah satu masukan yang diperlukan untuk membangun lini. Keterhubungan antar-*task* didefinisikan dengan cara memeriksa dua buah *task* yang mempunyai kebergantungan secara langsung terlebih dahulu. Jika suatu *task* dilakukan langsung setelah *task* lainnya selesai dieksekusi, maka hubungan antara kedua *task* tersebut harus didefinisikan. Keseimbangan lini tidak dapat tercapai jika ada keterhubungan antar-*task* yang bersifat *cyclic*. Keterhubungan beberapa *task* dikatakan *cyclic* jika memenuhi hal-hal sebagai berikut :

#### 1. Cyclic Langsung

Yaitu hubungan antar-*task* yang saling mempunyai kebergantungan *precedence* secara langsung. Contoh : *task* terakhir harus dilakukan sebelum *task* pertama dan sebaliknya, *task* pertama harus dilakukan sebelum *task* terakhir.

#### 2. Cyclic Tak Langsung

Yaitu hubungan antar-*task* jika ada dua buah *task* yang saling mempunyai kebergantungan *precedence* secara tidak langsung melalui *task* lainnya. Contoh : *task* pertama harus dilakukan sebelum *task* kedua, *task* kedua harus dilakukan sebelum *task* ketiga, *task* ketiga harus dilakukan sebelum *task* pertama.

Keterhubungan *cyclic* ini akan menyebabkan tak ada sebuah *task* pun yang dapat diberikan ke suatu stasiun kerja. Karena itu masukan keterhubungan antar-*task*, yang apabila digambarkan dengan graf, harus berupa graf non *cyclic*.

Pada keseimbangan lini terdapat batasan, yaitu bahwa sebuah *task* dapat diberikan kepada sebuah stasiun kerja jika dengan adanya penambahan *task* tersebut tidak membuat total waktu proses *task-task* di stasiun kerja tersebut melebihi waktu siklus tertentu. Sebuah stasiun kerja terdiri dari minimal satu buah *task*. Dengan demikian waktu proses setiap *task* tidak boleh melebihi waktu siklus agar semua *task* dapat dialokasikan ke suatu stasiun kerja. Batasan ini dapat dinotasikan dengan pertidaksamaan berikut ini :

$$t_i \leq C \quad (10)$$

dengan :

t<sub>i</sub> = waktu proses *task* ke-i  
C = waktu siklus  
i = 1, 2, 3, ..., N

### 3.2. Analisis Metode

Pada metode *Ranked Positional Weight*, dilakukan pembobotan berdasarkan waktu proses *task* tersebut dan *task-task* setelahnya. Untuk mengetahui *task-task* mana saja yang akan dieksekusi setelah *task* tersebut, perlu diketahui hubungan kebergantungan antar-*task*. Pendefinisian keterhubungan antar-*task* dilakukan sejak awal, yaitu setelah *task* dan waktu proses *task-task* didefinisikan.

*Task* ke-N yang merupakan *task* terakhir yang dieksekusi, mungkin tidak disebutkan secara eksplisit pada masukan bahwa *task* ke-1 adalah *task* pendahulu *task* ke-N karena keterhubungan antar-*task* yang didefinisikan sejak awal berupa kebergantungan langsung. Meski demikian, dengan menggunakan *Precedence Graph* terlihat dengan jelas hubungan kedua *task* tersebut, yaitu bahwa *task* ke-*i* merupakan salah satu *task* pendahulu dari *task*-N. Dengan demikian pada proses pembobotan metode ini perlu dilakukan penelusuran terhadap *task-task* yang mempunyai kebergantungan tak langsung. Keterhubungan antar-*task* berpengaruh terhadap pembobotan dan proses selanjutnya pada metode *Ranked Positional Weight*, karena itu penelusuran keterhubungan seluruh *task* harus memberikan hasil yang tepat.

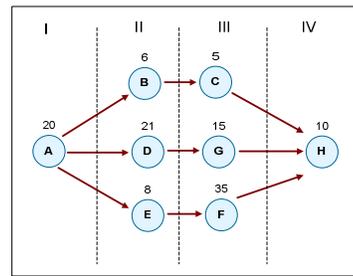
Berdasarkan urutan langkah pada metode *Ranked Positional Weight*, dapat disimpulkan bahwa penempatan *task* dilakukan berdasarkan urutan pembobotan dan dengan mempertimbangkan waktu yang tersisa di stasiun tersebut. Jika waktu di stasiun kerja yang bersangkutan masih ada, artinya masih ada *task* pada urutan selanjutnya yang mungkin dapat ditempatkan di stasiun kerja tersebut. Dengan mengetahui urutan *task-task* yang akan ditempatkan, dapat disimpulkan bahwa penempatan *task-task* ke suatu stasiun kerja relatif cepat karena penempatan *task* hanya memperhatikan waktu sisa dan *precedence constraint*.

Pada metode *Kilbridge Wester*, setiap *task* dikelompokkan ke dalam kelompok yang sama sesuai dengan tingkat keterhubungan setiap *task*. Kelompok ke-*i* adalah kelompok yang berisikan *task-task* yang tidak mempunyai *task* pendahulu, kelompok ke-*i*+1 adalah kelompok yang terdiri dari *task-task* yang mempunyai *task* pendahulu di kelompok ke-*i*, dan seterusnya. Kemudian ketika dilakukan penempatan *task*, kelompok yang diambil pertama kali adalah kelompok ke-*i*.

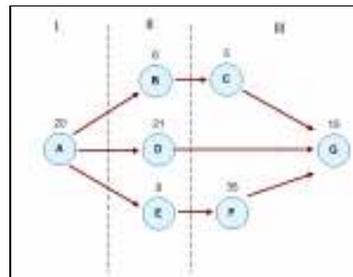
Ada satu hal penting yang harus diperhatikan terkait pengelompokan *task*. Misalnya terdapat pemisahan daerah kelompok seperti pada Gambar 1. Pada *precedence graph* tersebut terlihat bahwa pengelompokan *task-task* cukup mudah karena tingkat keterhubungan *task* yang sama, yaitu tidak ada *task-task* yang ditempatkan dalam satu kelompok dimana

terdapat keterhubungan tak langsung antara dua *task* atau lebih pada *task-task* yang berada pada kelompok tersebut.

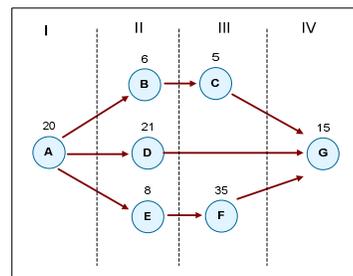
Pada kasus lainnya, misalnya seperti pada Gambar 2, terdapat kesalahan pada pengelompokan *task-task*. *Task* G tidak dapat ditempatkan ke dalam kelompok yang sama dengan *task* C dan F karena kedua *task* tersebut harus dilakukan sebelum *task* G dieksekusi. Hal ini berarti tingkat keterhubungan *task* C dengan G dan F dengan G berbeda. Dengan demikian pada metode ini perlu dilakukan pemeriksaan terhadap keterhubungan tidak langsung agar pengelompokan *task-task* tetap memenuhi *precedence constraint*. *Task* G harus ditempatkan pada kelompok setelah *task* C dan *task* F. Pengelompokan *task* yang ditunjukkan pada Gambar 3 adalah contoh pengelompokan *task-task* yang benar.



Gambar 1 Pengelompokan Task-task Kasus 1



Gambar 2 Pengelompokan yang Salah Pada Kasus 2



Gambar 3 Pengelompokan yang Benar Pada Kasus 2

Sebelum dilakukan penempatan *task-task* ke stasiun kerja, dilakukan pemilihan *task-task* yang memberikan waktu stasiun terbaik, yaitu waktu stasiun yang paling

mendekati waktu siklus dan tidak melebihi waktu siklus. Karena itu untuk mendapatkan waktu stasiun terbaik dilakukan kombinasi terhadap beberapa *task* yang akan ditempatkan di stasiun kerja yang sama.

### 3.3. Analisis Keluaran

Pembangunan lini dengan kedua metode ini menghasilkan *task-task* yang telah ditempatkan ke suatu stasiun kerja. Berdasarkan analisis masukan, semua *task* pasti dapat ditempatkan ke stasiun kerja manapun jika waktu siklus melebihi atau sama dengan waktu proses setiap *task*. Pada pembangunan lini dengan menggunakan metode *Ranked Positional Weight*, penempatan *task* ke suatu stasiun dilakukan berdasarkan hasil pembobotan. Sedangkan dengan metode *Kilbridge Wester*, penempatan *task* ke stasiun dilakukan berdasarkan tingkat keterhubungan antar-*task*. Metode *Ranked Positional Weight* memandang keterhubungan *task-task* secara global, yaitu antara suatu *task* dengan keseluruhan *task*, sedangkan pada metode *Kilbridge Wester* memandang keterhubungan *task-task* secara lokal, yaitu keterhubungan antara dua buah *task* antar kelompok.

Lini perakitan yang dibangun terdiri dari stasiun kerja dengan jumlah minimal satu. Setiap stasiun kerja pasti memiliki *task* tertentu yang harus dikerjakan. Jumlah *task* yang terdapat di sebuah stasiun kerja minimal berjumlah satu, artinya tidak mungkin sebuah stasiun kerja tidak memiliki *task* apapun yang harus dikerjakan. Pada metode *Ranked Positional Weight*, *task-task* yang banyak berpengaruh terhadap *task-task* lain, yaitu *task* yang harus dilakukan sebelum *task* lain dapat dieksekusi, cenderung ditempatkan di stasiun kerja awal. Sedangkan pada metode *Kilbridge Wester*, *task-task* yang tidak mempunyai pendahulu yang ditempatkan di stasiun kerja awal. Penempatan *task-task* lain ke dalam stasiun kerja selanjutnya tergantung kepada keterhubungan *task-task* tersebut dengan *task-task* yang telah ditempatkan pada stasiun kerja sebelumnya.

## 4. IMPLEMENTASI

Perangkat lunak pada Tugas Akhir ini dibangun dengan menggunakan bahasa pemrograman Java, *Java Development Kit (JDK) 1.5.0*. Spesifikasi perangkat lunak yang digunakan pada pembangunan perangkat lunak antara lain :

1. Sistem operasi Windows XP Service Pack 2
2. Kakas pengembang Netbeans IDE 5.5

Perangkat keras yang digunakan untuk mendukung pembangunan perangkat lunak adalah berupa satu unit komputer. Spesifikasi perangkat keras yang digunakan adalah sebagai berikut :

1. Prosesor : Intel Pentium IV 1,5 GHz
2. RAM : 640 MB
3. Hard Disk : 20 GB

## 5. PENGUJIAN

Pengujian dilakukan dengan membandingkan metode *Ranked Positional Weight* dan *Kilbridge Wester* berdasarkan kriteria pengujian tertentu. Tujuan perbandingan pada Tugas Akhir ini adalah untuk mengetahui metode mana yang dapat membangun lini perakitan yang terbaik, yaitu lini perakitan yang mempunyai jumlah stasiun yang seminim mungkin dan memiliki waktu stasiun yang semerata mungkin. Lini perakitan yang mempunyai waktu stasiun yang semerata mungkin mengindikasikan meratanya pembagian *task-task* di setiap stasiun kerja.

Kriteria yang diujikan antara lain :

1. Kevalidan lini perakitan yang terbentuk  
Kriteria ini digunakan untuk menguji apakah hasil pembangunan lini perakitan telah valid, yaitu memenuhi batasan keterhubungan yang didefinisikan oleh setiap *task* dan waktu stasiun di setiap stasiun kerja tidak melebihi waktu siklus.
2. Pengukuran lini perakitan  
Pengukuran lini perakitan bertujuan untuk membandingkan kedua metode, yaitu *Ranked Positional Weight* dan *Kilbridge Wester*, berdasarkan komponen-komponen tertentu. Berikut adalah beberapa komponen yang digunakan pada perbandingan.
  - a. Performansi  
Performansi diukur berdasarkan lama waktu eksekusi suatu metode. Kecepatan waktu eksekusi berbanding lurus dengan performansi metode. Dengan demikian metode dengan performansi terbaik adalah metode yang memiliki waktu eksekusi terendah.
  - b. Jumlah Stasiun  
Lini perakitan dengan stasiun kerja yang berjumlah seminim mungkin merupakan salah satu tujuan permasalahan keseimbangan lini yang ingin dicapai pada Tugas Akhir ini.
  - c. Efisiensi Lini  
Lini perakitan yang memiliki Efisiensi Lini tertinggi mengindikasikan suatu lini perakitan yang memiliki tingkat produktifitas tertinggi.
  - d. Balance Delay  
Lini perakitan yang memiliki Balance Delay terendah mengindikasikan suatu lini perakitan yang memiliki tingkat produktifitas tertinggi
  - e. Smoothness Index  
Lini perakitan yang memiliki Smoothness Index terendah mengindikasikan suatu lini perakitan yang memiliki tingkat pemerataan terbaik, yaitu dalam hal penempatan *task-task* ke dalam stasiun kerja.

### 5.1. Kasus Uji

Untuk menguji seluruh kriteria pengujian yang telah dibahas pada sub bab sebelumnya, digunakan tujuh jenis kasus uji yang berbeda-beda. Pada kasus uji terakhir, yaitu nomor UJ-04, digunakan data perakitan mobil, yaitu data yang sama yang digunakan sebagai contoh pada sub bab sebelumnya. Sedangkan data pada kasus uji sebelumnya merupakan data tidak riil yang digunakan untuk menguji kedua metode dalam berbagai jenis kasus yang berbeda-beda.

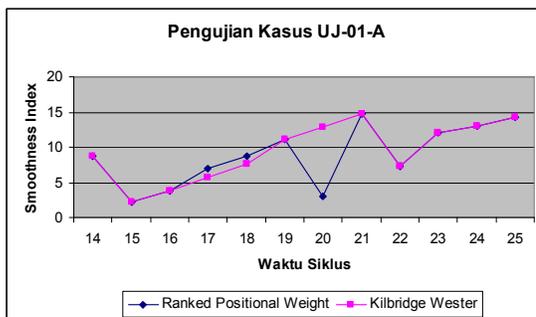
**Tabel 2 Daftar Kasus Pengujian**

Nomor Kasus Uji	Jumlah task	Keterhubungan	Nama File Uji
UJ-01-A	sedikit	sederhana	UJ-01-A.txt
UJ-01-B	sedikit	rumit	UJ-01-B.txt
UJ-02-A	sedang	sederhana	UJ-02-A.txt
UJ-02-B	sedang	rumit	UJ-02-B.txt
UJ-03-A	banyak	sederhana	UJ-03-A.txt
UJ-03-B	banyak	rumit	UJ-03-B.txt
UJ-04	sedang	rumit	UJ-04.xls

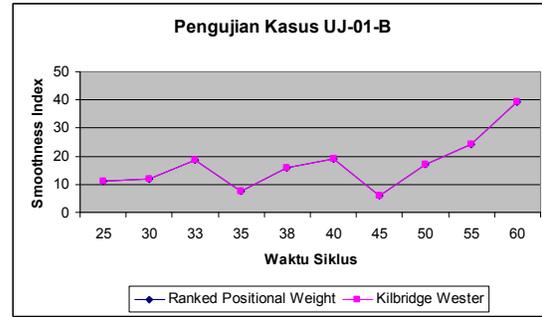
Untuk setiap jenis kasus uji dilakukan pengujian ulang untuk waktu siklus yang berbeda. Daftar file pengujian dan jenis kasus uji dapat dilihat pada Tabel 2.. Deskripsi kasus uji terakhir, yaitu UJ-04, dapat dilihat pada Tabel 1.

### 5.2. Hasil Pengujian

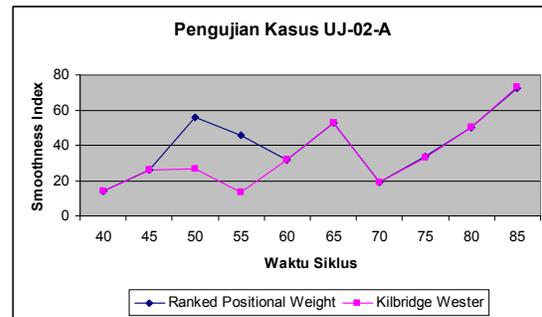
Pada grafik hasil pengujian, lini perakitan yang dihasilkan oleh kedua metode tersebut tidak stabil. Berdasarkan hasil pengukuran, grafik garis untuk metode *Ranked Positional Weight* tidak selalu berada di atas atau di bawah metode *Kilbridge Wester*. Hal ini mengindikasikan bahwa metode *Ranked Positional Weight* selalu lebih baik daripada metode *Kilbridge Wester*, dan sebaliknya.



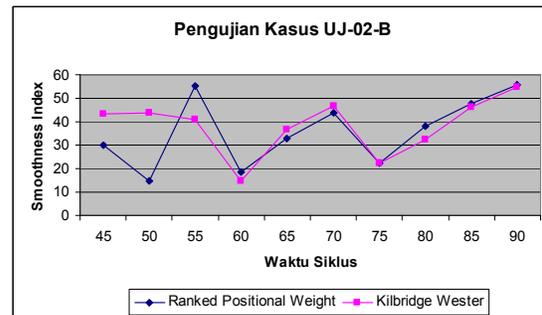
**Gambar 4 Grafik Pengujian Kasus UJ-01-A**



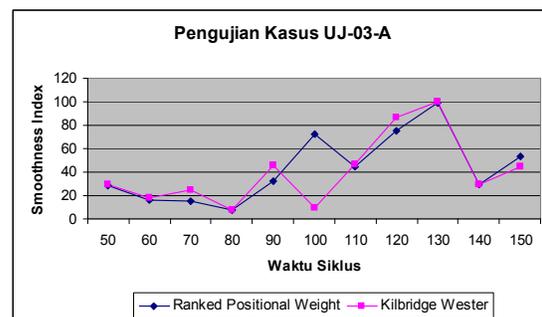
**Gambar 5 Grafik Pengujian Kasus UJ-01-B**



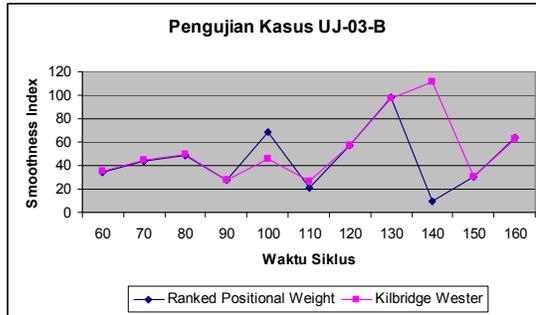
**Gambar 6 Grafik Pengujian Kasus UJ-02-A**



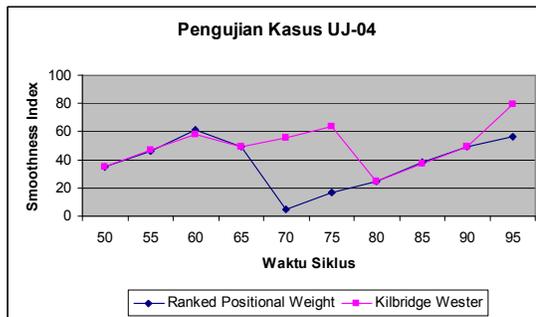
**Gambar 7 Grafik Pengujian Kasus UJ-02-B**



**Gambar 8 Grafik Pengujian Kasus UJ-03-A**



Gambar 9 Grafik Pengujian Kasus UJ-03-B



Gambar 10 Grafik Pengujian Kasus UJ-04

Berdasarkan pengujian yang telah dilakukan pada sub bab sebelumnya, dihasilkan dua buah lini perakitan yang dibangun dengan menggunakan metode yang berbeda. Untuk menjamin bahwa hasil lini perakitan yang dibangun telah valid, dilakukan pengujian terhadap beberapa jenis data uji yang berbeda. Berdasarkan hasil pengujian terhadap keseluruhan data uji dan setelah dilakukan pengecekan terhadap keterhubungan antar-task yang telah didefinisikan sebelumnya dan waktu siklus, seluruh lini perakitan yang dihasilkan telah valid karena tidak melanggar kedua batasan tersebut.

Dari segi performansi, pembangunan lini perakitan dengan menggunakan metode *Ranked Positional Weight* mempunyai performansi yang lebih baik dibandingkan dengan pembangunan lini perakitan dengan menggunakan metode *Kilbridge Wester*. Performansi dilihat dari perbandingan waktu eksekusi antara kedua metode tersebut. Waktu eksekusi proses pembangunan lini perakitan untuk metode *Kilbridge Wester* lebih lama sekitar 3 hingga 6 kali lipat dari waktu eksekusi metode *Ranked Positional Weight*. Hal ini disebabkan karena cara pembangunan lini pada metode *Ranked Positional Weight* sangat sederhana, yaitu melakukan pembobotan, mengurutkan, dan menempatkan task ke dalam stasiun. Sedangkan cara pembangunan lini perakitan dengan menggunakan metode *Kilbridge Wester* lebih rumit meskipun hanya terdiri dari dua langkah, yaitu mengelompokkan dan menempatkan task ke dalam stasiun.

Berdasarkan hasil pengukuran jumlah stasiun, efisiensi lini, *Balance Delay*, dan *Smoothness Index*,

pembangunan lini perakitan oleh kedua metode memberikan hasil yang tidak sama untuk waktu siklus yang berbeda. Untuk waktu siklus tertentu, metode *Ranked Positional Weight* lebih baik daripada metode *Kilbridge Wester* karena lini perakitan lebih seimbang. Untuk waktu siklus yang lain, dapat berlaku sebaliknya. Hal ini disebabkan karena penempatan task ke dalam stasiun kerja bergantung kepada waktu siklus yang dialokasikan di stasiun yang bersangkutan. Pada pembangunan lini perakitan dengan menggunakan metode *Ranked Positional Weight*, hasil pembobotan task akan selalu sama untuk waktu siklus yang berbeda-beda. Begitu pula dengan metode *Kilbridge Wester*, hasil pengelompokan task akan selalu sama untuk waktu siklus yang berlainan. Yang menjadikan lini perakitan berbeda adalah seberapa besar waktu siklus yang dialokasikan di setiap stasiun, karena waktu siklus mempengaruhi cara penempatan task pada kedua metode, yang pada akhirnya akan tidak selalu menghasilkan lini perakitan yang terbaik. Karena itu untuk kasus uji yang sama dan waktu siklus yang berbeda, metode *Ranked Positional Weight* tidak selalu lebih baik dari metode *Kilbridge Wester*, dan sebaliknya.

## 6. KESIMPULAN DAN SARAN

Beberapa kesimpulan yang dapat ditarik terkait kegiatan Tugas Akhir ini adalah sebagai berikut :

- Perangkat lunak yang mengimplementasikan teknik *Ranked Positional Weight* dan *Kilbridge Wester* berhasil dibangun. Semua lini perakitan yang dibangun dengan menggunakan perangkat lunak ini merupakan lini perakitan yang valid.
- Berdasarkan hasil perbandingan antara metode *Ranked Positional Weight* dan *Kilbridge Wester*, sulit untuk menentukan metode mana yang lebih baik. Pada kondisi tertentu metode *Ranked Positional Weight* lebih baik, sedangkan pada kondisi yang lain metode *Kilbridge Wester* yang lebih unggul. Tidak ada pola yang mengindikasikan bahwa suatu metode selalu lebih baik daripada metode lainnya pada contoh kasus tertentu. Hal ini tergantung kepada besar waktu siklus yang dialokasikan di setiap stasiun.
- Metode *Ranked Positional Weight* mempunyai kelebihan dalam hal kecepatan eksekusi. Berdasarkan hasil pengujian, terbukti bahwa dalam hal performansi metode *Ranked Positional Weight* memang lebih unggul dibandingkan dengan metode *Kilbridge Wester*.
- Pada metode *Ranked Positional Weight*, task dengan urutan pembobotan yang lebih tinggi, yaitu task yang mempunyai bobot yang lebih besar, tidak selalu ditempatkan di stasiun kerja yang lebih awal daripada task lain yang memiliki urutan pembobotan yang lebih rendah.
- Pada metode *Kilbridge Wester*, task ditempatkan ke dalam stasiun berdasarkan urutan kelompok. Kelompok yang lebih awal, yaitu kelompok

dimana *task-task* yang terdapat pada kelompok-kelompok setelahnya bergantung kepada *task-task* yang berada pada kelompok tersebut, pasti ditempatkan pada stasiun yang lebih awal dari semua kelompok setelahnya.

Saran yang dapat diberikan pada akhir pelaksanaan Tugas Akhir ini antara lain :

- a. Sebaiknya dilakukan optimasi terhadap metode *Kilbridge Wester*, yaitu dalam hal pengelompokan *task*. Pada teknik ini, pengelompokan dilakukan secara rata kiri. Dengan memperhitungkan pengelompokan *task* dengan cara lain, misalnya rata kanan atau rata tengah, dapat dihasilkan beberapa himpunan solusi lini perakitan. Solusi yang dipilih adalah solusi yang dapat menghasilkan lini perakitan dengan keseimbangan lini yang terbaik.
- b. Metode *Kilbridge Wester* mempunyai kekurangan dalam hal waktu eksekusi. Meski jenis optimasi seperti apapun tidak akan membuat waktu eksekusi metode ini menjadi lebih cepat daripada metode *Ranked Positional Weight*, sebaiknya dilakukan optimasi untuk mempercepat waktu eksekusi pembangunan lini perakitan dengan menggunakan teknik ini.

#### DAFTAR REFERENSI

- [BOY06] Boysen, et all. *Assembly Line Balancing : Which Model to Use When?*. Friedrich-Schiller-Universitat Jena. 2006  
[http://www.wiwi.uni-jena.de/Entscheidung/alb/Boyse\\_n%20et%20al.%20\(2006\)%20-%20Assembly%20line%20balancing%20-%20Which%20model%20to%20use%20when.pdf](http://www.wiwi.uni-jena.de/Entscheidung/alb/Boyse_n%20et%20al.%20(2006)%20-%20Assembly%20line%20balancing%20-%20Which%20model%20to%20use%20when.pdf)  
 (diakses tanggal 4 Juni 2008)
- [CHS04] Chutima, et all. 2004. *Practical Assembly-Line Balancing in a Monitor Manufacturing Company*. Chulalongkorn University  
[www.tijsat.tu.ac.th/issues/2004/no2/2004\\_V9\\_No2\\_8.PDF](http://www.tijsat.tu.ac.th/issues/2004/no2/2004_V9_No2_8.PDF)  
 (diakses tanggal 28 Februari 2008)
- [HAH03] Halim, A.H. 2003. *TI-3122 Perencanaan dan Pengendalian Produksi: Keseimbangan Lintasan*. Institut Teknologi Bandung  
<http://lspitb.org>  
 (diakses tanggal 4 Juni 2008)
- [KIW61] Kilbridge, M, Wester, L. 1961. *Management Science*, Vol. 8, No. 1, pp. 69-72. Informs  
<http://www.jstor.org/action/showPublication?journalCode=manascie>  
 (diakses tanggal 16 Juni 2008)
- [KSS07] Saha, et all. 2007. *Assembly Line Balancing*. Indian Institute of Technology Kanpur  
[http://home.iitk.ac.in/~sourabks/Papers/Term\\_Paper\\_ME765.pdf](http://home.iitk.ac.in/~sourabks/Papers/Term_Paper_ME765.pdf)  
 (diakses tanggal 10 Juni 2008)
- [MAL00] Malave, Cesar. 2000. *Approach to Line Balancing Comsoal & RPW*. Texas A&M University  
[www.foundationcoalition.org/resources/ie/LineBalancing/lb.ppt](http://www.foundationcoalition.org/resources/ie/LineBalancing/lb.ppt) (diakses tanggal 29 Februari 2008)
- [SLG07] Sly, et all. 2007. *A Practical Approach to Solving Multi-objective Line Balancing Problem*. Proplanner  
<http://www.proplanner.com/Documents/Support/LB/LBTechnical.pdf>  
 (diakses tanggal 29 Februari 2008)