

MINIMASI WAKTU Pengerjaan Produk MELANIE SLEIGH CHANGING TABLE MELALUI PENDEKATAN SIMULASI DAN *TABU SEARCH*

Natalia Sofyan, Meifani, dan I Gede Agus W.
Laboratorium Optimasi dan Simulasi, Jurusan Teknik Industri
Universitas Kristen Petra Surabaya

ABSTRAK

Penelitian ini memiliki fokus pada penyelesaian suatu permasalahan yang kompleks melalui pendekatan sistem. Permasalahan yang dibahas adalah bagaimana menurunkan waktu pengerjaan produk di industri perkayuan melalui proses penjadwalan yang tepat. Metode simulasi dan *Tabu Search* digunakan dalam penyelesaian permasalahan. Simulasi digunakan untuk menyelesaikan permasalahan melalui pendekatan sistem sedangkan *Tabu Search* digunakan untuk mencari solusi yang optimal. Hasil penelitian ini menunjukkan bahwa gabungan pendekatan yang diusulkan memiliki solusi yang lebih baik daripada kondisi awal.

Kata kunci: simulasi, *Tabu Search*, penjadwalan.

ABSTRACT

The focus of this research is to solve a complex problem through the system approach. As a study case, we tried to minimize the production time at furniture industry using the appropriate process scheduling. Both simulation and Tabu Search methods were employed to solve the scheduling problem. The simulation helped to solve the problem through system approach, while the Tabu Search was employed to find the optimal solutions. The result shows that the hybrid method gave better solution than the initial condition.

Keywords: simulation, *Tabu Search*, scheduling.

1. PENDAHULUAN

Salah satu permasalahan yang paling kompleks di dunia industri terutama pada bagian operasional berada pada industri perkayuan. Oleh karena itu penelitian ini difokuskan pada industri kayu terutama pada produk *Melanie Sleigh Changing Table*. Produk *Melanie Sleigh Changing Table* adalah produk dengan karakteristik jarang dipesan namun jika tiba-tiba ada order jumlahnya sangat banyak. Produk ini menjadi penghambat pengerjaan produk lain pada saat yang sama ketika sedang diproduksi. Proses pembuatan produk ini sangatlah rumit karena tersusun dari 17 komponen. Sebagai batasan, pada penelitian ini pengamatan dilakukan pada pengerjaan di panel 1 yang terdiri dari 15 komponen.

Penelitian ini menggunakan metode simulasi mengingat begitu rumitnya proses yang terjadi di panel 1 dan banyaknya parameter atau proses yang tidak dapat diasumsikan. Hasil dari simulasi yang dibuat akan digunakan untuk melihat waktu penyelesaian semua komponen produk *Melanie Sleigh Changing Table* yang dikerjakan di panel 1 yang kemudian akan digunakan sebagai acuan dalam pembuatan simulasi usulan. Simulasi usulan dibuat dengan tujuan untuk meminimalkan waktu penyelesaian proses pengerjaan produk *Melanie Sleigh Changing Table*.

Simulasi sebagai salah satu metode yang tidak dapat menghasilkan solusi yang optimal, namun hanya bisa menghasilkan solusi yang lebih baik. Disisi lain, usulan yang akan dibuat memiliki banyak sekali kemungkinan. Oleh sebab itu untuk mendapatkan solusi yang mendekati solusi yang optimal maka digunakan metode *Tabu Search*.

2. SIMULASI DAN *TABU SEARCH*

Simulasi dikenal sebagai suatu metode yang dapat digunakan dalam penyelesaian masalah-masalah operasional suatu perusahaan. Hal ini dikarenakan adanya salah satu keunggulan simulasi yaitu kemampuannya menangani masalah-masalah kompleks yang ada di dunia nyata yang tidak dapat diselesaikan dengan metode-metode analitis (Law & Kelton, 1991). Pendekatan ini memberikan suatu solusi dengan pendekatan sistem sehingga lebih memudahkan pada identifikasi masalah dan pengembangan alternatif solusinya.

Disamping memiliki keunggulan seperti di atas, metode simulasi memiliki kelemahan. Kelemahan utama metode ini adalah tidak dijaminnya mendapatkan solusi terbaik. Solusi yang didapat berupa coba-coba (*trial & error*). Untuk mengurangi kelemahan ini, simulasi dapat digabungkan dengan metode optimasi lainnya. Dalam penelitian ini diupayakan untuk menggabungkan metode simulasi dengan *Tabu Search*.


Algoritma ini diperkenalkan pertama kali oleh Glover (Glover, 1986). Penggunaan algoritma ini antara lain pada *graph coloring problem* (Hertz & de Werra, 1987), *maximum independent set problem* (Friden, Hertz & de Werra, 1990), *the course scheduling problem* (Hertz, 1992). Ide dasar *Tabu Search* seperti halnya metode metaheuristic yang lain adalah tidak hanya mencari solusi yang lebih baik dan meninggalkan solusi yang kurang baik. Hal ini akan menyebabkan solusi terjebak pada lokal optimal. Untuk menjaga agar solusi terbaik tidak hilang, maka pada algoritma *Tabu Search* akan menyimpan solusi terbaik dan terus mencari berdasarkan solusi terakhir. Selain itu metode ini akan menyimpan memori solusi yang pernah ditemui dan melarang untuk menggunakan solusi yang telah dipakai sehingga menghindarkan dari perulangan yang sia-sia.

3. MODEL

3.1 Model Simulasi

- *Entity*: komponen-komponen *melanie sleigh changing table*
- *Attribute*: warna dari produk *melanie sleigh changing table*
- *System state*: waktu penyelesaian seluruh komponen *melanie sleigh changing table*
- *Event*:
 - kedatangan dan selesainya komponen *melanie sleigh changing table* di dalam lantai produksi
- *Exogenous variable*:
 - waktu antar kedatangan komponen *melanie sleigh changing table*
 - jenis mesin yang digunakan untuk membuat komponen *melanie sleigh changing table*
 - waktu proses tiap komponen *melanie sleigh changing table* pada mesin-mesin tertentu
 - urutan *assembly* komponen *melanie sleigh changing table*
- *Endogenous variable* : waktu penyelesaian seluruh komponen *melanie sleigh changing table*

3.2 Layout simulasi

Langkah berikutnya adalah pembuatan layout simulasi yang disesuaikan dengan layout di perusahaan seperti terlihat pada Gambar 1. Gambar meli­hat­kan simbol  yang melambangkan *buffer*. *Buffer* digunakan untuk 2 keperluan, yang pertama *buffer* digunakan untuk menggambarkan mesin-mesin yang ada pada panel 1 namun tidak digunakan dalam proses pembuatan komponen *Melanie Sleigh Changing Table*. Kedua, *buffer* juga berfungsi sebagai tempat pengumpulan *inventory* sesaat sebelum dipindahkan ke mesin berikutnya. Tujuannya adalah agar pengiriman komponen yang sama hanya dilakukan satu kali saja. Pada Gambar 1 tidak dijelaskan mengenai jenis mesin namun terdapat kode-kode seperti M, DET, RS, dll yang sebenarnya menunjukkan nama mesin.



Gambar 1. Layout mesin untuk produk *melanie sleigh changing table*

3.3 Simulasi Awal

Ada beberapa asumsi yang digunakan dalam pembuatan simulasi tentang *melanie sleigh changing table* yaitu tidak memperhitungkan *move logic* (waktu perpindahan) antar mesin karena tidak terlalu signifikan. Pengukuran waktu menggunakan konversi berdasarkan komponen yang berbeda yang dikerjakan pada mesin-mesin tertentu.

Masalah pada *melanie sleigh changing table* adalah jumlah order yang selalu besar dengan warna yang berbeda dan menyebabkan pihak lantai produksi panel 1 susah untuk memproduksi barang lain pada saat yang sama.

Simulasi ini juga memprioritaskan komponen *sleigh changing table* untuk dikerjakan pada mesin tertentu. Jadi, ketika misalnya mesin *double end toner* sudah siap untuk mengerjakan 2 komponen milik *sleigh changing table* dan milik produk lain maka yang dikerjakan terlebih dahulu adalah komponen milik *sleigh changing table*.

Hal yang diperhatikan dalam simulasi ini adalah waktu dalam sistem dari produk ini dan tujuan dibuatnya solusi adalah untuk meminimalkan waktu penyelesaian seluruh pesanan yang berjumlah 430 set (4 warna). Di bawah ini adalah tabel jumlah pesanan tiap warna dari *melanie sleigh changing table*.

Tabel 1. Jumlah pesanan *melanie sleigh changing table* setiap warna

Warna	Jumlah pesanan
<i>White</i>	160 set
<i>Honey Pine</i>	100 set
<i>Cherry</i>	90 set
<i>Maple</i>	80 set

Pada simulasi awal *output* untuk penyelesaian seluruh pesanan adalah 375,7661 jam. Hal ini dengan jadwal kedatangan bahan pada hari yang sama namun pengerjaan diasumsikan mengikuti urutan kode komponen dalam arti komponen 1-1 dikerjakan paling dulu dan komponen 4-6 dikerjakan paling akhir dengan selang waktu kedatangan tiap jenis komponen adalah 5 detik.

4. SOLUSI

Simulasi solusi adalah berupa adanya pengaturan urutan komponen yang harus dikerjakan terlebih dahulu. Metode pengaturan yang digunakan adalah *Tabu Search heuristic algorithm* di mana tiap iterasinya akan menghitung penyelesaian seluruh komponen sampai terkumpul sampai siap dirakit.

Langkah-langkah *Tabu Search Algorithm* pada simulasi solusi kali ini adalah sebagai berikut;

Step 1

- menentukan ukuran *tabu list* (v) yaitu $0.33n$ sampai $0.6n$;
Ukuran *tabu list* (v) $0.33n-0.6n$, yaitu 5-9 *list*. Dalam hal ini jumlah *tabu list* (v) yang dipilih adalah 8. Pemilihan jumlah *tabu list* ini tidak berdasarkan acuan apapun karena aturan dalam *tabu list* mengijinkan pemilihan jumlah *tabu list* berapapun asalkan masih dalam range nilai 5-9.
- menentukan jumlah iterasi maksimum (t) yaitu $7n$ sampai $10n$.
Jumlah maksimum iterasi yang dilakukan adalah sebanyak $7n$ atau $10n$, di mana n adalah jumlah komponen. Dalam hal ini, jumlah komponen *Melanie Sleigh Changing Table* yang dikerjakan adalah 15 komponen jadi jumlah iterasi yang harus dilakukan adalah maksimalnya 105 kali atau 150 kali. Karena keterbatasan waktu dipilih iterasi maksimal(t) 105 kali.

Step 2

- menentukan *sequence* awal yang digunakan sebagai iterasi awal ($k=0$) (dapat menggunakan metode apapun yang disesuaikan dengan tujuan yang ingin dicapai);
Sequence awal yang dibuat adalah berdasarkan urutan kode komponen. Jadi komponen 1-1 akan dikerjakan terlebih dahulu sedangkan komponen 4-6 akan dikerjakan paling akhir.
- menentukan OFV, dimana OFV adalah tujuan yang ingin dicapai (contoh: *minimum makespan* atau *minimum weighted tardiness*).
Objective Function Value(OFV) yang dipilih adalah lama waktu penyelesaian pengerjaan seluruh komponen karena tujuan simulasi solusi ini adalah untuk meminimalkan waktu penyelesaian pengerjaan seluruh komponen.

Step 3

- membuat perhitungan nilai $k=k+1$ untuk setiap satu iterasi yang diselesaikan;
Nilai k yang awalnya adalah nol maka ketika akan dilakukan iterasi 1 nilai k bertambah satu untuk menunjukkan bahwa saat ini dilakukan iterasi 1 atau iterasi 2, dan seterusnya.

- melakukan penukaran tiap kombinasi *job* yang ada di mana pertukaran hanya dilakukan pada 2 komponen saja, misal komponen 1 ditukar dengan komponen 2, komponen 1 ditukar dengan komponen 3, dst;

Pada setiap iterasi dilakukan pertukaran jadwal antara 2 komponen. Misalnya komponen 1-1 ditukar dengan komponen 1-2, komponen 1-2 ditukar dengan komponen 1-3, dan seterusnya sampai komponen 4-5 ditukar dengan komponen 4-6. menghitung OFV masing-masing kombinasi;

- memilih OFV yang paling minimum dalam 1 iterasi;
Setiap kali dilakukan penukaran jadwal pengerjaan komponen, nilai OFV-nya dihitung. Dalam satu iterasi terjadi pertukaran sebanyak 105 kali untuk 14 komponen sehingga ada 105 juga nilai OFV tiap iterasinya. Dari 105 nilai OFV tiap iterasi kemudian dipilih nilai OFV yang terkecil.
- memasukkan pertukaran *job* yang menghasilkan OFV paling minimum ke dalam *tabu List* kemudian urutan *job* yang menghasilkan OFV minimum ditukar.

Setelah dipilih nilai OFV terkecil, kemudian pasangan yang menghasilkan nilai OFV terkecil dimasukkan ke dalam *tabu list*. Dalam iterasi pertama yang menghasilkan nilai OFV minimum adalah pasangan komponen 1-4 dan 4-6 dengan nilai OFV 369.9079. Urutan komponen kemudian berubah untuk kedua komponen ini dan urutan baru ini digunakan sebagai acuan untuk melakukan iterasi kedua. *Tabu list* untuk iterasi pertama adalah:

1-4;4-6							
---------	--	--	--	--	--	--	--

Pada iterasi kedua juga dilakukan pertukaran tiap 2 komponen yang kemudian menghasilkan nilai OFV minimum sebesar 369.8015 untuk pertukaran komponen 1-1 dan 2-1. Kedua pasangan komponen ini juga dimasukkan ke dalam *tabu list* karena OFVnya lebih kecil dari nilai OFV iterasi pertama. Urutan komponen hasil pertukaran komponen 1-1 dan 2-1 kemudian dijadikan acuan untuk melakukan iterasi ketiga. Jadi *tabu list* saat ini adalah,

1-1;2-1	1-4;4-6						
---------	---------	--	--	--	--	--	--

Pada iterasi ketiga yang menghasilkan nilai OFV minimum adalah pertukaran komponen 1-2 dan 4-1 dengan nilai OFV 369.6988. Pasangan kedua komponen ini kemudian ditukar dan dijadikan acuan untuk iterasi keempat. Karena nilai OFV-nya lebih kecil daripada nilai OFV hasil iterasi ketiga maka *tabu list* saat ini adalah,

1-2;4-1	1-1;2-1	1-4;4-6					
---------	---------	---------	--	--	--	--	--

Iterasi keempat menghasilkan nilai OFV minimum sebesar 369.664944 untuk pertukaran komponen 3-3 dan 1-2. Setelah pertukaran dilakukan untuk kedua komponen ini kemudian urutan ini dijadikan acuan untuk iterasi kelima. *Tabu list* saat ini adalah,

3-3;1-2	1-2;4-1	1-1;2-1	1-4;4-6				
---------	---------	---------	---------	--	--	--	--

- jika pada iterasi ke-k OFV minimumnya tidak lebih kecil daripada OFV minimum ke-k-1 maka pertukaran *job* tetap dilakukan namun urutan penukaran tidak dimasukkan ke dalam *Tabu List*.

Pada iterasi kelima nilai OFV paling minimum yang dihasilkan adalah 369.664964, dimana sedikit lebih besar daripada hasil OFV minimum iterasi keempat. Ini berarti pertukaran jadwal pembuatan komponen tetap ditukar namun pasangan ini tidak dimasukkan ke dalam *tabu list*. Pertukaran komponen ini tetap menjadi acuan untuk iterasi 6. Jadi, *tabu list* pada iterasi kelima masih sama dengan *tabu list* pada iterasi keempat yaitu,

3-3;1-2	1-2;4-1	1-1;2-1	1-4;4-6				
---------	---------	---------	---------	--	--	--	--

Iterasi keenam menghasilkan nilai OFV minimum sebesar 369.6581 untuk pertukaran komponen 2-2 dan 3-1. Ini berarti pertukaran diterima dan dijadikan acuan untuk iterasi berikutnya yaitu iterasi ketujuh. *Tabu list* saat ini adalah,

2-2;3-1	3-3;1-2	1-2;4-1	1-1;2-1	1-4;4-6			
---------	---------	---------	---------	---------	--	--	--

Iterasi ketujuh kemudian dilakukan dengan urutan baru hasil iterasi keenam di mana menghasilkan nilai OFV minimum sebesar 369.6648944 untuk komponen 3-1 dan 2-2. Karena nilai OFV ini tidak lebih kecil daripada nilai OFV pada iterasi keenam, pertukaran pasangan ini tidak dimasukkan ke dalam *tabu list*. Namun, tetap saja hasil pertukaran ini dijadikan acuan untuk melakukan iterasi kedelapan.

2-2;3-1	3-3;1-2	1-2;4-1	1-1;2-1	1-4;4-6			
---------	---------	---------	---------	---------	--	--	--

Step 4

- jika $k < t$, ulangi step 3. Namun jika terjadi pengulangan di mana pertukaran komponen yang akan dimasukkan ke dalam *tabu list* sudah ada, iterasi akan dihentikan
Iterasi kedelapan kemudian dilakukan dan ternyata menghasilkan nilai OFV minimum sebesar 369.6581 untuk pertukaran komponen 2-2 dan 3-1. Hal ini sama persis dengan yang terjadi pada iterasi keenam. Oleh karena itu simulasi ini dihentikan.

- hasil urutan yang terakhir yang diperoleh setelah simulasi berhenti merupakan urutan yang mendekati *global optima*.
Iterasi telah dihentikan. Urutan akhir hasil dari iterasi kedelapan adalah urutan yang paling mendekati nilai *global optima* dengan *tabu list* dan urutan komponen usulan sebagai berikut,

2-2;3-1	3-3;1-2	1-2;4-1	1-1;2-1	1-4;4-6			
---------	---------	---------	---------	---------	--	--	--

Dari hasil penukaran komponen pada akhir iterasi kedelapan didapatkan waktu penyelesaian pengerjaan komponen dari hasil simulasi sebesar 369.6581 jam. Hasil ini secara langsung dapat dilihat lebih kecil dari pada hasil pada simulasi awal yaitu sebesar 375,7661 jam. Simulasi solusi hasil pertukaran ini direplikasi sebanyak 20 kali untuk menghilangkan faktor bias atau kebetulan. *Mean* penyelesaian pengerjaan seluruh komponen 369.9282 jam dengan standar deviasi sebesar 0.1880. Jumlah replikasi ini kemudian diuji apakah telah memenuhi jumlah replikasi minimal dengan rumus $N' = (st/kx)^2$ di mana s adalah standar deviasi hasil simulasi waktu penyelesaian pengerjaan komponen yaitu 0.1880, t adalah nilai t_α pada tabel t dengan α sebesar 0.05 dengan nilai *degree of freedom* sebesar 19 yaitu 1.729, k adalah nilai α yaitu 0.05, dan x adalah *mean* dari waktu penyelesaian pengerjaan seluruh komponen yaitu 369.9282. Dari sini diperoleh nilai N' adalah 0.1326. Nilai ini lebih kecil dari jumlah replikasi awal yaitu 20 kali sehingga dapat dikatakan bahwa jumlah replikasi yang dilakukan sudah layak.

5. KESIMPULAN

Produk *Melanie Sleigh Changing Table* setelah menggunakan *Tabu Search Heuristic Algorithm* diperoleh penurunan *mean* waktu penyelesaian sebesar 369.9282 jam dimana sebelumnya adalah 375,7661 jam. Urutan pengerjaan usulan untuk komponen produk ini adalah seperti pada Tabel 2.

Tabel 2. Kode dan Nama Komponen *melanie sleigh changing table* dengan urutan jadwal yang terbaru

KODE	NAMA KOMPONEN
2-1	<i>Left Back Leg</i>
4-1	<i>Top Front Cross</i>
1-2	<i>Right Front Leg</i>
4-6	<i>Top Front Rail</i>
1-5	<i>Lower Side Cross</i>
1-1	<i>Right Back Leg</i>
3-1	<i>Back Top Rail</i>
2-2	<i>Left Front Leg</i>
3-2	<i>Back Upper Rail</i>
1-3	<i>Upper Side Cross</i>
3-3	<i>Back Slat</i>
4-2	<i>Mid Bottom Cross</i>
4-3	<i>Side Mattress Support</i>
4-5	<i>Bottom Mattress Support</i>
1-4	<i>Side Slat</i>

Melihat hasil dan proses yang dilakukan, maka penelitian ini menunjukkan bahwa hibrid pendekatan simulasi dan *Tabu Search* dapat memberikan solusi yang baik pada pemecahan masalah minimasi waktu pengerjaan produk dengan pendekatan sistem. Satu hal yang menjadi kelemahan penelitian adalah belum diintegrasikannya dua pendekatan ini dalam satu *software*, sehingga cukup memakan waktu dalam pencarian solusinya. Selain itu masih ada kemungkinan untuk meningkatkan solusi waktu penyelesaian menjadi lebih baik lagi.

Penelitian lebih lanjut dapat dilakukan dengan menggabungkan antara simulasi dan metode *tabu search* dalam satu *software* penyelesaian sehingga mempercepat pencarian solusi dan dengan menggunakan gabungan metode simulasi dengan metode *metaheuristic* lainnya seperti *genetic algorithm* ataupun *ant colony*.

DAFTAR PUSTAKA

- Friden C., Hertz A., de Werra D., 1990, *TABARIS: an exact algorithm based on Tabu Search for finding a maximum independent set in a graph*, Computers and Operations Research 17, pp. 437-445.
- Glover F., 1986, *Future Paths for Integer Programming and Links to Artificial Intelligence*, Computers and Operations Research 13, pp. 533-549.
- Hertz A., de Werra D., 1987, *Using Tabu Search Techniques for Graph Coloring*, Computing 39, pp. 345-351.
- Hertz A., 1992, *Finding a Feasible Course Schedule Using Tabu Search*, Discrete Applied Mathematics 35, pp. 255-270.
- Law, A.M, and Kelton W.D., 1991, *Simulation Modelling and Analysis*, second edition, Mc Graw Hill, Inc.