

Usulan Penjadwalan Produksi Dengan Algoritma *Ant Colony* (Studi Kasus PT Shima Prima Utama Palembang)

Liliani¹, Achmad Alfian²

¹Jurusan Teknik Industri, Sekolah Tinggi Teknik Musi, Palembang
Jl. Bangau 60 Palembang 30113 Telp 0711 366326

²Jurusan Teknik Industri, Sekolah Tinggi Teknik Musi, Palembang
Jl. Bangau 60 Palembang 30113 Telp 0711 366326
Email: alfian_60@yahoo.com, alfian60@gmail.com

Abstrak

PT Shima Prima Utama adalah perusahaan yang memproduksi alat-alat kesehatan yang bersifat job shop. Penjadwalan produksi selama ini berdasarkan kesamaan proses produksi, tanpa mempertimbangkan waktu siklus penyelesaian seluruh job. Hal ini menyebabkan waktu penyelesaian seluruh produksi (makespan) menjadi lebih panjang. Berdasarkan hal tersebut maka di dalam penelitian ini ini diberikan alternatif metode penjadwalan produksi melalui penerapan algoritma Ant Colony untuk meminimalkan makespan. Algoritma Ant Colony merupakan salah satu pendekatan metaheuristik yang mampu memberikan hasil positif untuk menyelesaikan permasalahan optimasi dengan menemukan solusi yang baik. Perhitungan yang menyebar dapat menghindari terjebak pada keadaan lokal optimum, serta dengan penerapan algoritma heuristik Greedy yang mampu menghasilkan solusi cukup baik pada tahap awal pencarian. Algoritma Ant Colony diimplementasikan dalam bahasa pemrograman Visual Basic dengan kombinasi parameter berdasarkan makespan. Kombinasi parameter berdasarkan makespan adalah 50 siklus, 8 semut, $\alpha = 10$, $\beta = 1$, dan $\rho = 0,5$. Algoritma Ant Colony menghasilkan makespan sebesar 230697,89 detik. Sedangkan metode Shortest Processing Time (SPT) menghasilkan makespan sebesar 223250,63 detik. Hasil usulan rancangan penjadwalan produksi yaitu AA1 - BB1 - GG1 - CC2 - DD2 - EE3 - CC1 - HH2 - MM1 - KK1 - DD1 - EE4 - LL1 - EE2 - EE6 - KK2 - EE1 - FF1 - HH3 - EE5 - III - II2 - HH1 - JJ1 - JJ3 - JJ2. Penjadwalan produksi dengan penerapan algoritma Ant Colony dapat meminimalkan makespan sebesar 7447,26 detik atau 3,23% dibandingkan dengan metode Shortest Processing Time.

Kata kunci: Algoritma Ant Colony; Makespan; Parameter; Penjadwalan produksi

Pendahuluan

PT Shima Prima Utama merupakan sebuah perusahaan yang bergerak di bidang produksi alat-alat kesehatan. Perusahaan ini bersifat *job order* dan juga memproduksi produk sebagai *inventory*. Apabila terdapat *order*, maka produk akan diproduksi berdasarkan permintaan *customer* yang diberikan oleh bagian *marketing*. Permintaan tersebut digunakan oleh bagian *Production Planning and Inventory Control* (PPIC) untuk menyusun rencana produksi selama satu bulan. Setelah itu bagian PPIC merencanakan kebutuhan material untuk masing-masing komponen penyusun produk, sedangkan jadwal produksi disusun oleh kepala divisi di rantai produksi. Daftar rencana produksi yang diberikan oleh bagian PPIC digunakan oleh kepala produksi dalam menyusun jadwal produksi. Selama ini jadwal produksi disusun secara harian dengan berdasarkan material yang tersedia di gudang.

Dalam menentukan urutan produk yang akan diproduksi, maka kepala produksi mengelompokkan tipe produk yang memiliki kesamaan proses untuk diproduksi secara bersama-sama. Sedangkan antara produk yang tidak memiliki kesamaan proses, maka tidak ada prioritas khusus. Jadwal produksi yang telah disusun selanjutnya diberikan kepada operator untuk dikerjakan. Setiap operator ditetapkan target dalam mengerjakan suatu proses tertentu. Keterbatasan tenaga kepala produksi dalam menyusun jadwal produksi dan mengatur sumber daya yang tersedia, menyebabkan waktu yang dibutuhkan untuk menyelesaikan keseluruhan produksi menjadi lebih panjang. Adanya proses produksi yang memerlukan waktu siklus yang cukup panjang menyebabkan nilai *makespan* yang tinggi.

Oleh karena itu, perlu adanya penjadwalan produksi untuk menanggulangi masalah tersebut. Salah satu metode penjadwalan yang dapat meminimasi *makespan* dan menghasilkan solusi yang mendekati optimal adalah algoritma *Ant Colony*. Dengan melakukan proses penjadwalan produksi maka nilai *makespan* dapat diminimasi sesuai dengan harapan perusahaan. Sesuai dengan fenomena di atas, rumusan masalah untuk penelitian ini adalah bagaimana usulan rancangan penjadwalan produksi yang dapat meminimalkan *makespan* di PT Shima Prima Utama. Tujuan dilakukannya penelitian ini adalah menerapkan algoritma *Ant Colony* untuk memperoleh metode

penjadwalan produksi dengan nilai parameter terbaik dan meminimasi nilai *makespan* dengan algoritma *Ant Colony* dibandingkan dengan metode heuristik lainnya.

Untuk mendukung penelitian maka ditetapkan batasan masalah dan asumsi. Batasan masalah pada penelitian ini adalah produk yang dibahas merupakan produk yang rutin dikerjakan dan sesuai dengan pertimbangan dari pihak perusahaan yaitu *Baby Cot With Trolley*, *Instrument Trolley*, *Baby Dressing Bed*, *Instrument Cabinet*, *Economy Double Bowl Stand* dan *Medicine Trolley With 66 Drawers* dan pembahasan dilakukan hanya pada metode penjadwalan produksi tanpa meneliti sumber daya yang lain. Sedangkan asumsi yang digunakan pada penelitian ini adalah tidak ada perubahan metode kerja selama penelitian dilakukan, tidak ada penambahan dan pengurangan sumber daya selama penelitian dilakukan, tidak terjadi kerusakan mesin selama penelitian dilakukan dan jam kerja yang tersedia pada seluruh mesin adalah sama, bahan baku selalu tersedia dan cukup untuk memenuhi target produksi, *ready time* pada seluruh *job* adalah nol dan *no preemption*.

Tinjauan Pustaka

Algoritma *Ant Colony*

Algoritma *Ant Colony* merupakan salah satu pendekatan metaheuristik yang mampu memberikan hasil positif untuk menyelesaikan permasalahan optimasi dengan menemukan solusi yang baik. Perhitungan yang menyebar dapat menghindari terjebak pada keadaan lokal optimum, serta dengan penerapan algoritma heuristik *Greedy* yang mampu menghasilkan solusi cukup baik pada tahap awal pencarian.

Latar Belakang Biologi

Algoritma ini berawal dari pengamatan terhadap kegiatan semut dalam mencari makanan, sehingga semut yang lain dapat mengetahui jalan terbaik untuk mencari sumber makanan. Dalam perjalanan dari sarang semut sampai ke sumber makanan, maka semut mengeluarkan wewangian yang bernama *pheromone*. *Pheromone* dapat dicium oleh semut-semut dari koloni berikutnya, sehingga mereka akan terpengaruh untuk mengikuti jalan yang sama. Berawal dari pengamatan ini, maka Deneubourg dan temannya melakukan percobaan menggunakan dua buah jembatan yang panjangnya divariasikan untuk menghubungkan sarang semut dengan letak makanan.

Prinsip Dasar Algoritma *Ant Colony*

Dalam algoritma *Ant Colony*, terdapat sejumlah semut yang masing-masing akan menggunakan jalan terpendek dalam mencari kandidat solusi dari suatu permasalahan. Proses pembentukan solusi berupa *construction heuristic* dengan menerapkan algoritma heuristik *Greedy*, yaitu melakukan pemilihan secara bertahap berdasarkan nilai fungsi *pheromone trail* dan informasi heuristik yang terbesar. *Pheromone trail* menunjukkan kualitas solusi yang telah dicapai oleh semut dari perjalanan sebelumnya, sedangkan informasi heuristik sesuai dengan *input* data dari suatu permasalahan. Parameter *pheromone trail* diberi bobot sebesar α , sedangkan parameter informasi heuristik diberikan bobot sebesar β . Dalam menghasilkan solusi pada setiap tahapan, maka semut melakukan proses eksploitasi atau eksplorasi berdasarkan probabilitas tertentu.

Solusi yang dipilih pada setiap tahap akan disimpan di dalam *tabu list*, yaitu daftar solusi yang tidak boleh dipilih pada tahapan berikutnya. Setelah semut memilih suatu pilihan tertentu untuk dimasukkan ke dalam *tabu*, maka nilai *pheromone trail* yang bersangkutan akan mengalami evaporasi. Kegiatan ini dilakukan oleh semua semut dalam dalam satu koloni. Setelah satu koloni semut menyusun kombinasi solusi, maka dilakukan pemilihan semut terbaik yang akan dibandingkan dengan semut terbaik secara global. Nilai *pheromone trail* dari *job* terbaik secara global akan mengalami evaporasi, sehingga mempengaruhi semut pada koloni berikutnya dalam menyusun urutan *job*. Nilai *pheromone trail* mengalami evaporasi untuk menghindari semut terlalu cepat mencapai konvergensi pada keadaan lokal optimum.

Parameter Algoritma *Ant Colony*

Berikut ini merupakan parameter-parameter yang terdapat pada algoritma *Ant Colony*:

- NC_{max} , yaitu jumlah siklus yang dibutuhkan dalam mencari suatu solusi. Jumlah siklus menunjukkan jumlah koloni semut dan di dalamnya terdapat sejumlah semut yang akan mencari solusi dari suatu permasalahan. Jika jumlah siklus semakin besar, maka solusi yang dihasilkan akan semakin mendekati solusi optimal.
- Jumlah semut (m), yaitu jumlah kombinasi solusi pada tiap siklus. Semakin banyak jumlah semut, maka semakin banyak kombinasi solusi yang dapat dipilih pada tiap siklus.
- Pheromone trail* (τ_{ij}), yaitu wewangian dari tiap semut selama perjalanan agar semut berikutnya dapat mengikuti jalan yang sama.
- Informasi heuristik (η_{ij}), yaitu visibilitas suatu *job* dipilih pada setiap tahapan berdasarkan nilai fungsi secara matematis.
- Tingkat kepentingan relatif dari *pheromone trail* (α), yaitu besarnya bobot yang diberikan terhadap parameter *pheromone trail*, sehingga solusi yang dihasilkan cenderung mengikuti sejarah masa lalu dari semut dalam perjalanan sebelumnya. Nilai untuk parameter α adalah ≥ 0 .

- f. Tingkat kepentingan relatif dari informasi heuristik (β), yaitu besarnya bobot yang diberikan terhadap parameter informasi heuristik, sehingga solusi yang dihasilkan cenderung berdasarkan nilai fungsi matematis. Nilai untuk parameter β adalah ≥ 0 .
 - g. Koefisien evaporasi (ρ), yaitu besarnya koefisien penguapan dari *pheromone trail*. Adanya penguapan *pheromone trail* menyebabkan tidak semua semut mengikuti alur yang sama dengan semut sebelumnya. Nilai untuk parameter koefisien evaporasi berkisar pada *range* 0 sampai 1.
- Probabilitas semut melakukan proses eksploitasi pada setiap tahapan (q_0), dimana $0 \leq q_0 \leq 1$.

Prosedur Algoritma Ant Colony

Algoritma *Ant Colony* menggunakan koloni semut, dimana setiap semut akan memilih nilai terbaik pada setiap tahapan yang dipengaruhi oleh semut sebelumnya dan kualitas dari setiap alur. Mekanisme pemilihan solusi oleh semut dapat dilakukan melalui dua proses (Bauer, n.d., p.7), yaitu:

- a. Eksploitasi, yaitu semut memilih *job* yang memiliki nilai fungsi *pheromone trail* dan informasi heuristik yang paling tinggi di antara seluruh kandidat *job* yang ada. Hal ini dilakukan untuk menghindari pencarian yang terlalu random dan jauh dari solusi optimal. Berikut ini merupakan persamaan dalam memilih *job* berdasarkan proses eksploitasi.

$$Job\ j = \text{arrange maximum}_{h \in \Omega} \{ [\tau_{ih}]^\alpha [\eta_{ih}]^\beta \}$$

dimana: Ω = himpunan *job* yang belum dimasukkan ke dalam *tabu* (k).

- b. Eksplorasi, yaitu semut melakukan penyelidikan terhadap suatu *job* dengan probabilitas pemilihan berdasarkan nilai fungsi *pheromone trail* dan informasi heuristik. Proses ini dapat menghasilkan alternatif solusi yang lebih banyak sehingga mengurangi kemungkinan terjebak pada keadaan lokal.

$$P_{ij} = \begin{cases} \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{h \in \Omega} [\tau_{ih}]^\alpha [\eta_{ih}]^\beta} & \text{jika } j \in \Omega \\ 0 & \text{selain itu} \end{cases}$$

Setelah setiap semut memilih satu *job* ke dalam *tabu* pada urutan tertentu, maka *pheromone trail* dari *job* pada urutan tersebut akan mengalami evaporasi yang disebut proses *local trail update* sebagai berikut:

$$\tau_{ij}(t+1) = (1-\rho) \tau_{ij}(t) + \rho \tau_0$$

$$\tau_0 = \frac{1}{n \cdot T_{SPT}}$$

dimana: τ_0 = *pheromone trail* awal
 T_{SPT} = nilai fungsi tujuan dari metode SPT

Setelah semua *job* dimasukkan ke dalam *tabu* oleh setiap semut, selanjutnya dilakukan perhitungan nilai fungsi tujuan dari setiap semut. Semut terbaik dari tiap siklus akan dibandingkan dengan solusi terbaik secara global dari siklus sebelumnya. *Pheromone trail* dari solusi terbaik secara global akan mengalami evaporasi yang dikenal dengan proses *global trail update* sebagai berikut:

$$\tau_{ij}(t+1) = (1-\rho) \tau_{ij}(t) + \rho \Delta \tau_{ij}(t)$$

$$\Delta \tau_{ij}(t) = \begin{cases} \frac{1}{T^*} & \text{untuk setiap } i, j \text{ dari solusi terbaik} \\ 0 & \text{selain itu} \end{cases} \quad \text{dimana: } T^* = \text{nilai fungsi tujuan dari penerapan}$$

algoritma *Ant Colony*

Fungsi Tujuan dan Parameter Algoritma Ant Colony

Fungsi Tujuan

Penyusunan jadwal produksi ini bertujuan untuk meminimumkan waktu penyelesaian dari *n job* yang dijadwalkan. Adapun rumusan fungsi tujuan adalah sebagai berikut:

$$Z = \min (C_j)$$

dimana : $j = 1, 2, \dots, n$
 C_j = *makespan* dari *n job*

Penetapan Nilai Parameter Algoritma Ant Colony

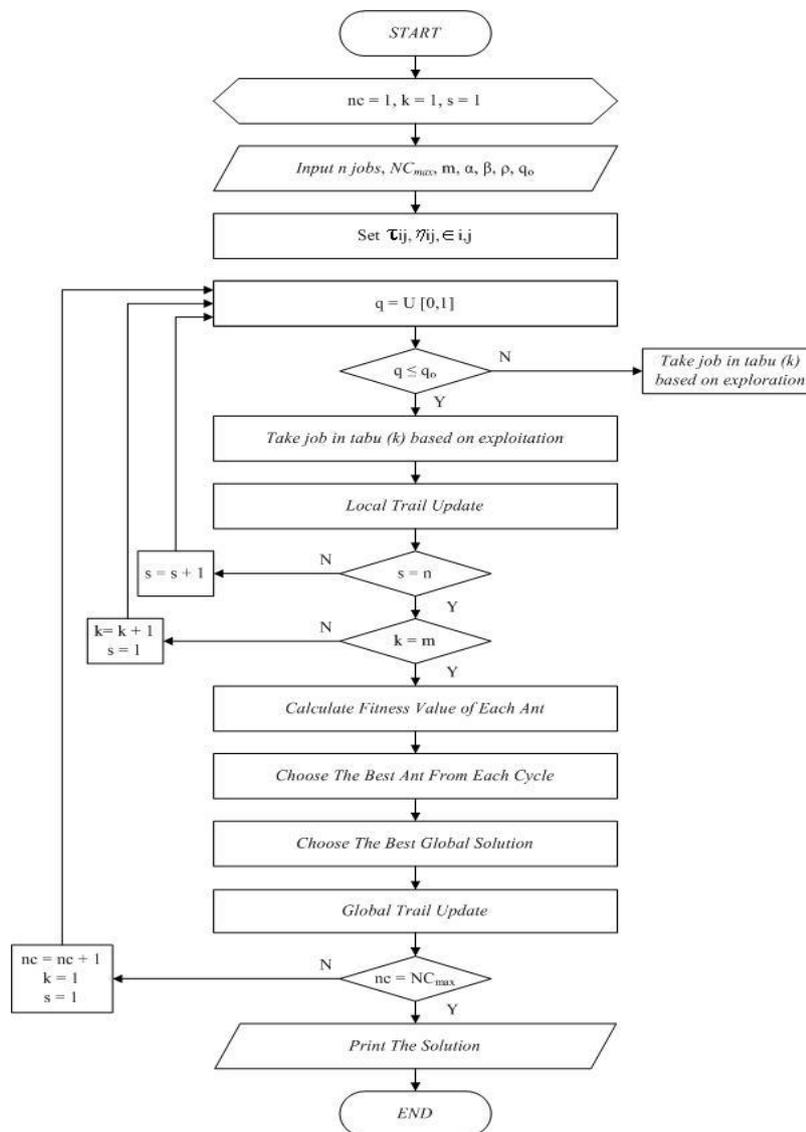
Algoritma *Ant Colony* memiliki beberapa parameter yang nilainya dapat berpengaruh terhadap *makespan*. Untuk mendapatkan nilai parameter yang terbaik, maka dilakukan *running* program dengan kombinasi level dari lima parameter. Setiap parameter terdiri atas dua level, dengan nilai level dari masing-masing parameter terdapat pada Tabel 1.

Tabel 1 Penetapan Level dari Parameter ANT COLONY

Parameter	Level rendah (-1)	Level Tinggi (1)
NCmax	20	50
m	4	8
α	1	10
β	1	10
ρ	0.1	0.5

Sumber : Sjoerd van der Zwaan & Carlos Marques, *Ant Colony Optimisation for Job Shop Scheduling*

Sesuai dengan prinsip optimisasi maka level terbaik untuk parameter jumlah siklus adalah level 1, yaitu 50 siklus. Hal ini menunjukkan bahwa semakin banyak jumlah siklus, maka *makespan* yang dihasilkan semakin rendah karena semakin banyak iterasi yang dijalankan dalam mencari solusi yang lebih baik. Semakin bertambahnya jumlah siklus maka solusi yang dihasilkan semakin mendekati solusi global optimum. Sedangkan level terbaik untuk parameter jumlah semut adalah level 1, yaitu 8 semut. Ketika jumlah semut semakin banyak, maka alternatif solusi yang dapat dipilih pada tiap iterasi semakin banyak. Hal ini sesuai dengan prinsip pencarian solusi secara menyebar pada algoritma *Ant Colony*. Bertambahnya jumlah siklus dan jumlah semut maka alternatif solusi semakin banyak sehingga menghasilkan solusi yang lebih baik.



Gambar 1. Flow Chart Metodologi Penelitian

Hasil dan Pembahasan

Tabel 2. Urutan Pengerjaan Job

Nama Produk	Jumlah Jenis	Nama Produk	Jumlah Jenis
1 Ranjang / Hospital Bed		2 Lemari	
- Motorized Hi-Lo	3	- Instrument Cabinet	1
- Manual Hi-Lo	3	- Medicine Cabinet	1
- Advance	3	- Bedside Cabinet	4
- Moderate	3	3 Tongkat	15
- Standard	3	4 Kursi Roda	9
- Children	2	5 Trolley	12
- Examination	1	6 Others	19
- Gynaecology	2		
- Delivery	1		
Total		82	

Tabel 3. Data Penjadwalan Perusahaan

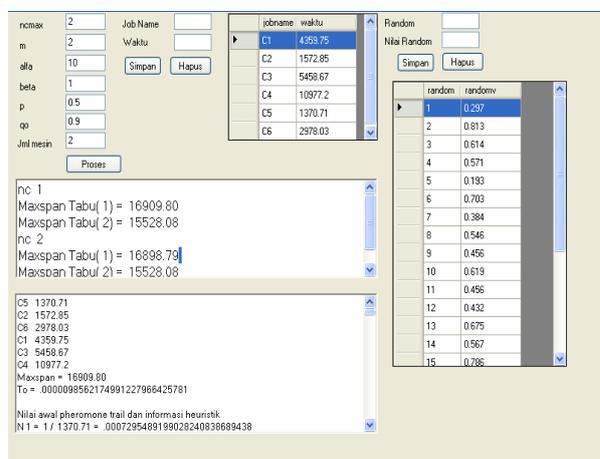
Divisi	Nama Mesin/Operator	Kode Mesin	Jumlah (Unit)
Preparasi	Mesin ukur & potong pipa	A	1
	Mesin ukur & potong plat	B	1
	Mesin bending	C	2
	Mesin bor	K	2
Welding	Gerinda	H	3
	Mesin punch	D	2
	Mesin las	E	6
	Mesin roll	G	1
Coating	Mesin coak	L	1
	Operator inspeksi	F	1
	Alat fitting	M	1
	Operator coating	I	2
Assembly	Operator assembly	J	3

Tabel 4. Data Produksi/Bulan

Nama Produk	Kode Produk	Mat	Produksi/ Bulan
Baby Cot With Trolley	SM-9018B	SS	3
Instrument Trolley	SM-9046A	SS	8
Baby Dressing Bed	SM-9124	PC	1
Instrument Cabinet	SM-9063	PC	6
Economy Double Bowl Stand	SM-9033B	PC	3
Medicine Trolley With 66 Drawers	SM-9048	PC	1

Program *Ant Colony* dijalankan dengan menggunakan *software Visual Basic 6.0* dengan jumlah replikasi sebanyak 7 kali dengan kombinasi nilai parameter yang berbeda-beda. *Makespan* minimum diperoleh dari kombinasi parameter $NC_{max} = 50$ siklus, $m = 8$ semut, $\alpha = 10$, $\beta = 1$ dan $\rho = 0.5$. Nilai *makespan* yang diperoleh adalah sebesar 223250.63 detik. Berikut hasil *running* program yang optimum dilihat pada Gambar 2.

Program dijalankan dengan jumlah replikasi sebanyak 7 kali dengan kombinasi nilai parameter yang berbeda-beda. Dari setiap kombinasi nilai parameter *Ant Colony* dilakukan beberapa percobaan. Hasil percobaan berupa *makespan* yang dinyatakan dalam satuan detik dan dapat dilihat pada Tabel 4.

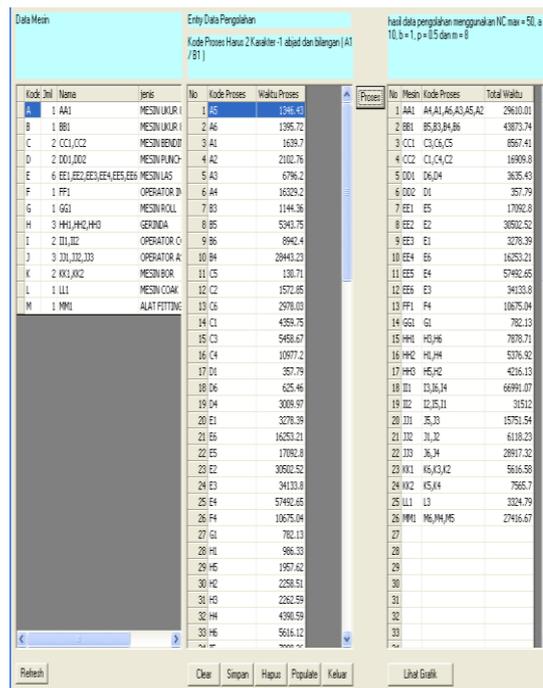


Gambar 2. Output Algoritma ANT COLONY

Tabel 5. *Makespan* Setiap Percobaan

α	β	ρ	NCmax	m	<i>Makespan</i>
1	1	0.1	50	8	238903.39
1	1	0.5	50	8	237460.81
1	10	0.1	50	8	244453.77
1	10	0.5	50	8	248033.23
10	1	0.1	50	8	232068.60
10	1	0.5	50	8	223250.63
10	10	0.1	50	8	239699.74

Makespan minimum diperoleh dari kombinasi parameter NCmax = 50 siklus, m = 8 semut, $\alpha = 10$, $\beta = 1$ dan $\rho = 0.5$. Nilai *makespan* yang diperoleh adalah sebesar 223250.63 detik. Hasil *running* program yang optimum dapat dilihat pada Gambar 3



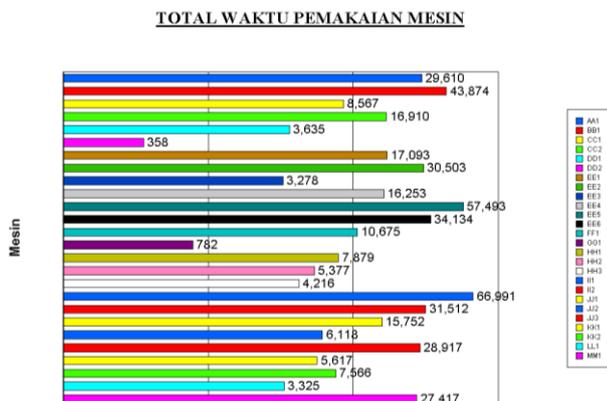
Gambar 3 Hasil *Running* Parameter Optimum

Tabel 6. Urutan Proses Produksi Metode SPT

No	Kode Proses	Waktu siklus <i>job</i> (detik)	No	Kode Proses	Waktu siklus <i>job</i> (detik)
1	A5	1346.43	28	H1	986.33
2	A6	1395.72	29	H5	1957.62
3	A1	1639.70	30	H2	2258.51
4	A2	2102.76	31	H3	2262.59
5	A3	6796.20	32	H4	4390.59
6	A4	16329.20	33	H6	5615.12
7	B3	1144.36	34	I5	7989.26
8	B5	5343.75	35	I3	8575.56
9	B6	8942.40	36	I6	10396.2
10	B4	28443.23	37	I2	11729.81
11	C5	1370.71	38	I1	11792.93
12	C2	1572.85	39	I4	48019.31
13	C6	2978.033	40	J2	2151.28
14	C1	4359.75	41	J1	3966.95
15	C3	5458.67	42	J6	5088.39
16	C4	10977.2	43	J5	5227.47
17	D1	357.79	44	J3	10524.07
18	D6	625.46	45	J4	23823.93

19	D4	3009.97	46	K5	412.67
20	E1	3278.39	47	K2	492.63
21	E6	16253.21	48	K6	1667.66
22	E5	17092.80	49	K3	3456.29
23	E2	30502.52	50	K4	7153.03
24	E3	34133.8	51	L3	3324.79
25	E4	57492.65	52	M5	2354.95
26	F4	10675.04	53	M6	2354.95
27	G1	13782.00	54	M4	21277.43

Penjadwalan produksi dengan metode *Shortest Processing Time* (SPT) menghasilkan *makespan* sebesar 230697.89 detik. Sedangkan waktu pemakaian mesin dapat dilihat pada Gambar 4.



Gambar 4 Penjadwalan Produksi dengan metode *Shortest Processing Time* (SPT)

Tabel 8. Perbandingan *Makespan* Algoritma Ant Colony dengan Metode SPT

<i>Makespan</i>	
Algoritma Ant Colony	Metode <i>Shortest Processing Time</i> (SPT)
223250,63 detik	230697,89 detik

Pembahasan

Berdasarkan tabel diatas maka diketahui bahwa *makespan* yang dihasilkan dengan algoritma *Ant Colony* lebih kecil dibandingkan dengan *makespan* yang dihasilkan dengan metode *Shortest Processing Time* (SPT). Algoritma *Ant Colony* menghasilkan *makespan* sebesar 223250,63 detik. Sedangkan metode *Shortest Processing Time* (SPT) menghasilkan *makespan* sebesar 230697,89 detik. Oleh karena itu, algoritma *Ant Colony* merupakan metode penjadwalan produksi yang menghasilkan solusi optimum untuk meminimasi *makespan*. Perbedaan nilai *makespan* antara algoritma *Ant Colony* dengan metode SPT adalah sebesar 7447,26 detik. Jadi dibandingkan dengan metode SPT, penerapan algoritma *Ant Colony* dalam penjadwalan produksi dapat meminimasi rata-rata *makespan* sebesar 3,228%.

Kesimpulan

Berdasarkan hasil analisa terhadap penerapan algoritma *Ant Colony* dalam menyelesaikan permasalahan penjadwalan produksi, maka diperoleh kesimpulan sebagai berikut:

- Hasil usulan rancangan penjadwalan produksi yaitu AA1 - BB1 - GG1 - CC2 - DD2 - EE3 - CC1 - HH2 - MM1 - KK1 - DD1 - EE4 - LL1 - EE2 - EE6 - KK2 - EE1 - FF1 - HH3 - EE5 - II1 - II2 - HH1 - JJ1 - JJ3 - JJ2.
- Nilai parameter yang menghasilkan *makespan* paling minimum adalah interaksi 50 siklus, 8 semut, $\alpha = 10$, $\beta = 1$, dan $\rho = 0.5$. *Makespan* yang dihasilkan dari interaksi ini adalah sebesar 223250.63 detik.
- Dengan menggunakan asumsi yang sama, penjadwalan produksi dengan penerapan algoritma *Ant Colony* memberikan hasil yang lebih baik dibandingkan dengan metode *Shortest Processing Time*. Adanya penerapan algoritma *Ant Colony* dalam penjadwalan produksi dapat menurunkan rata-rata *makespan* sebesar 7447.26 detik atau 3.228%.

Daftar Pustaka

Bauer, A., Bullnheimer, B., Hartl, R.F., & Strauss, C. *An ant colony optimization approach for the single machine total tardines problem*. Department of Management Science. University of Vienna. dalam Internet: <http://citeseer.ist.psu.edu/cache/papers/cs/10752/http:zSzzSzwwww.bwl.unive.ac.atzSzbwlzSzprodzszpaperszSzpom-wp-1-99.pdf/bauer99ant.pdf> diakses pada 8 Agustus 2011.

Bhattacharya, Gouri K. & Johnson, Richard A. 1977. *Statistical concepts and methods*, John Wiley & Sons, Inc. Canada.

Dorigo, M., Stutzle, T. *The ant colony optimization metaheuristic: Algorithms, applications, and advances*. dalam Internet: www.acm.ndsu.nodak.edu/~nhuff/the-ant-colony-optimization.pdf. diakses pada 8 Agustus 2011

Morton, Thomas E. & Pentico, David W. 1993. *Heuristic Scheduling Systems*, Canada, John Wiley & Sonc, Inc.

Niebel, B. & Freivalds, A. 2003. *Methods Standards and Work Design*, 11nd ed., McGraw Hill Company Inc. New York.

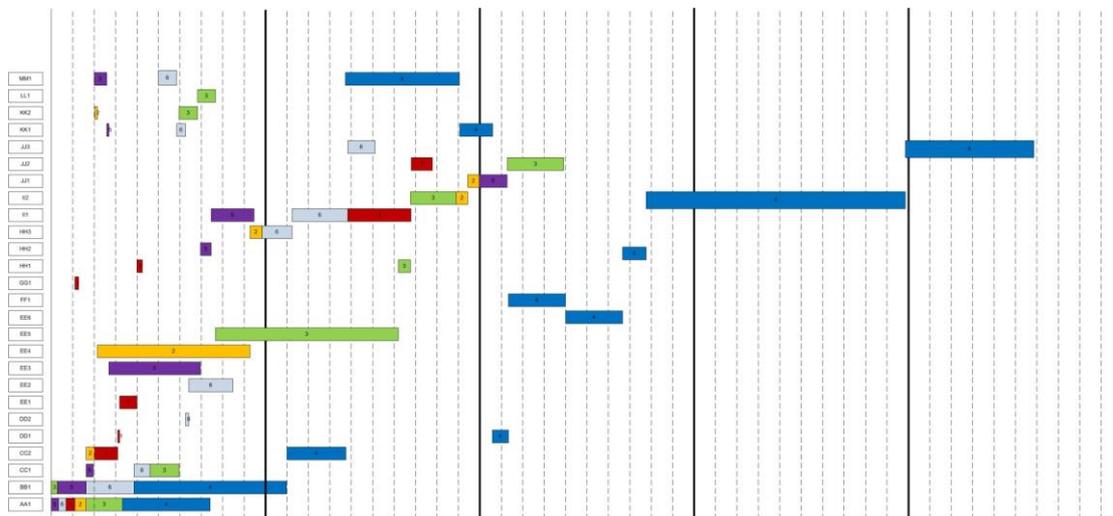
Pranata, Lina O. 2006. Penjadwalan Produksi dengan Algoritma *Ant Colony* di PT FSCM *Manufacturing* Indonesia Plant 4, Tugas Akhir Fakultas Teknologi Industri Universitas Kristen Petra.

Saptadi, Singgih & Gunawan, Aris. 2003. Aplikasi Algoritma *Ant Colony System* pada Penjadwalan *Flow Shop* di CV Citra Indomebel, Tugas Akhir Fakultas Teknik Industri Universitas Diponegoro.

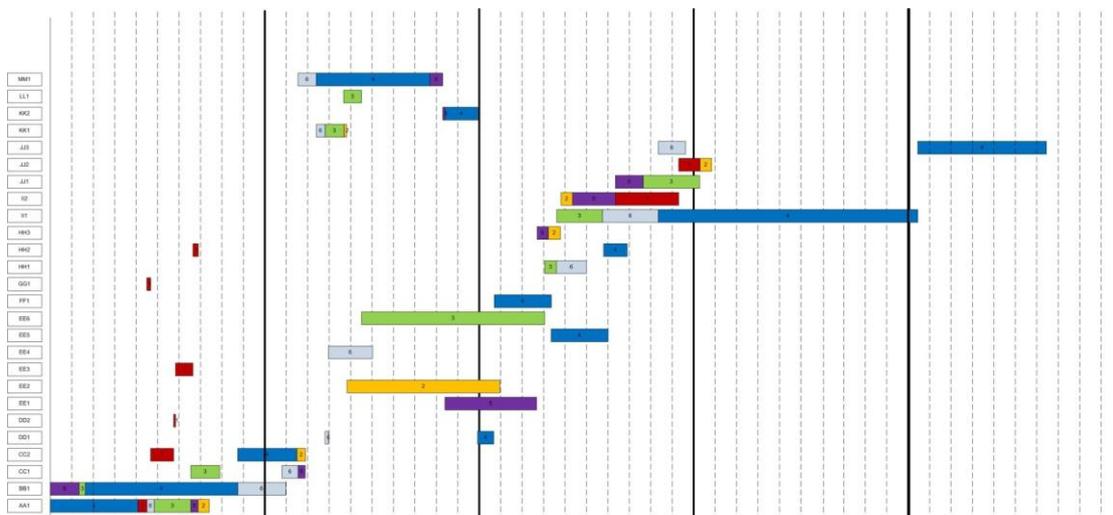
Sutalaksana, I.Z., Anggawisastra, R., & Tjakraatmadja, J.H. 1979. *Teknik Tata Cara Pengukuran Kerja*, 2nd ed., Institut Teknologi Bandung.

Wignjosoebroto, Sritomo. 2000. *Ergonomi Studi Gerak dan Waktu*, Penerbit Guna Widya, Surabaya.

Zwaan, S., Marques, C. *Ant colony optimisation for job shop scheduling*. dalam Internet: <http://citeseer.ist.psu.edu/cache/papers/cs/6401/http:zSzzSzviriato.ir.ist.utl.ptzSz~sjoerdzSzants.pdf/vanderzwaan99ant.pdf>. diakses pada 8 Agustus 2011



GANTT CHART METODE SPT



GANTT CHART ALGORITMA ANT COLONY