

PENJADWALAN PROSES PRODUKSI MENGGUNAKAN ANT COLONY ALGORITHM

Dedy Kurniadi¹⁾, Sam Farisa Chaerul Haviana²⁾

^{1,2} Jurusan Teknik Informatika, Fakultas Teknologi Industri,
Universitas Islam Sultan Agung Semarang
Jl. Raya Kaligawe Km.4 Semarang
Email : ddy.kurniadi@unissula.ac.id

ABSTRACT

Pada penelitian ini telah dilakukan analisa penjadwalan pada sektor industri garment, penelitian ini dilakukan dengan tujuan untuk mengoptimalkan proses produksi pada perusahaan CV. Inda Jaya sebagai alternatif pemecahan masalah yang bisa digunakan untuk membantu mengoptimalkan waktu pemrosesan produksi dengan menggunakan ant colony algorithm, algoritma tersebut mempunyai tingkat keberhasilan yang tinggi, ant colony algorithm juga banyak digunakan untuk menyelesaikan berbagai macam proses penjadwalan baik di bidang industri maupun bidang lainnya, ant colony algorithm sendiri banyak dikembangkan karena memang memberikan hasil yang memuaskan dalam menemukan rute terpendek untuk menyelesaikan permasalahan penjadwalan, proses yang dilakukan pada penelitian ini terbagi dua tahap dimana tahap pertama yaitu mengolah bahan mentah kemudian menjadi bahan setengah jadi yang diproses pada tahap dua yaitu proses finishing, hasil dari penelitian ini adalah generate otomatis makespan dengan menggunakan ant colony algorithm, pada tahap pertama menghasilkan jumlah makespan sebanyak 110 menit hasil tersebut menghemat waktu pemrosesan produksi sebanyak 10 menit dan pada tahap kedua menghasilkan waktu makespan sebanyak 140 menit hasil tersebut menghemat waktu sebanyak 40 menit.

Kata Kunci : Ant Colony Algorithm, Makespan, Penjadwalan

1. PENDAHULUAN

Penjadwalan produksi merupakan hal yang sangat vital pada proses produksi sebuah produk pada perusahaan, produksi sendiri adalah sebuah proses atau cara untuk menambahkan jumlah produk menggunakan cara – cara tertentu dengan memperhatikan tingkat produksi yang optimal dengan meminimalkan total waktu yang digunakan serta menghemat jumlah biaya yang dikeluarkan untuk satu kali proses produksi (Dewi dan Abusini, 2010).

Pasar membutuhkan pasokan yang memadai untuk mencukupi jumlah kebutuhan dari produk yang diinginkan (*demand*), proses penjadwalan bisa dijadikan solusi untuk memenuhi kebutuhan produk dan bisa dijadikan key concept untuk proses produksi, salah satu metode yang bisa dijadikan alternatif awal adalah *first come first served* (FCFS) metode tersebut lebih mengutamakan prioritas dalam proses produksinya.

Perusahaan yang bergerak dibidang produksi adalah CV. Inda Jaya Perusahaan

yang bergerak di bidang Garment, memproduksi tiga jenis produk yaitu Kemeja Kantor, Kaos Polo Bordir dan Celana Jeans Bordir, proses produksi menggunakan cara konvensional dalam satu kali produksi mengabiskan waktu 2 jam atau 120 menit untuk mesin Jahit dan pada mesin bordir membutuhkan waktu 180 menit untuk setiap 12 produk, proses produksi tersebut menggunakan metode first come first served (FCFS) pada perusahaan tersebut proses produksi berdasarkan prioritas mesin yang available pada mesin yang available tersebut proses produksi dijalankan secara bersama - sama artinya proses produksi tidak berdasarkan tahapan kontinuitas tetapi berdasarkan proses yang bisa dikerjakan terlebih dahulu,.

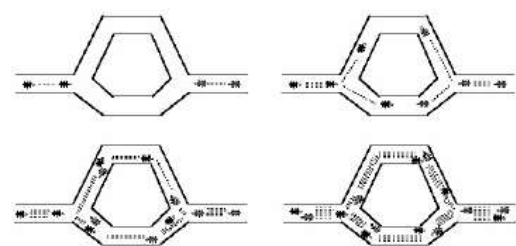
Penelitian ini mencari alternatif yang bisa digunakan untuk proses penjadwalan yang bisa mengoptimalkan waktu pengerjaan satu kali proses produksi, adapun algoritma yang diterapkan pada penelitian ini adalah menggunakan algoritma semut atau sering disebut dengan ant colony algorithm, algoritma ini adalah salah satu algoritma yang menyelesaikan masalah komputasi dengan teknik probabilistik yaitu dengan menemukan jalur terpendek atau terbaik yang akan diambil berdasarkan perilaku semut dalam

menemukan jalur ke koloninya (Hidayat, 2007).

Kriteria yang akan diukur pada penelitian ini adalah waktu penyelesaian seluruh pekerjaan (Job) pada kriteria ini diukur keseluruhan total waktu yang digunakan untuk menyelesaikan sebuah job apabila waktu yang digunakan semakin kecil maka produktivitas akan semakin baik dan semakin optimal.

2. LANDASAN TEORI

Algoritma semut merupakan algoritma yang diadopsi berdasarkan perilaku yang ditunjukkan oleh semut yang berkoloni dikenal juga dengan sebutan *ant systems*, semut secara alamiah bisa menemukan rute-rute yang terbaik dan terpendek dari koloni mereka ke tempat sumber makanan yang dituju (Kambey, 2011).



Gambar 1. Perjalanan semut ke sumber makanan (Alamsyah, 2010)

1.1 Langkah-Langkah Algoritma Semut

- a. Inialisasi paramter yang ada
- b. Menghitung *makespan time* setiap *ant*

$$Z_{best} = \max \{C_i, i=1, 2, \dots, n\}$$

$$(1)$$

$$C_i = q(\sigma_i, m)$$

$$(2)$$

$$q(\sigma_i, j) = \max \{q(\sigma, j); q(\sigma_i, j-1)\} + t_{ij}$$

$$(3)$$

c. Penghitungan probabilitas

$$P_{ij}(t) = \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k=0}^n [\tau_{ik}]^\alpha \cdot [\eta_{ik}]^\beta}$$

$$(4)$$

d. Memperbarui intensitas jejak kaki

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij} \quad (5)$$

$$\Delta\tau_{ij} = \frac{1}{P_{ij} \cdot C} \quad (6)$$

$$\Delta\tau_{ij} = \begin{cases} P_{ij}^{-1} & \text{Jika } ij \in \text{tur_terbaik} \\ 0 & \end{cases} \quad (7)$$

Dimana :

P_{ij} = waktu pekerjaan i pada mesin j

C = jumlah lokasi

ρ = nilai 0 sampai dengan 1

$\Delta\tau_{ij}$ = perubahan *pheromone*

e. Jika pemberhentian terpenuhi atau jumlah iterasi sudah selesai ambil urutan selanjutnya (*job*) dengan *makespan time* paling kecil (Yuwono, dkk, 2009).

3. TUJUAN

1. Merancang sistem penjadwalan dengan *ant colony algorithm*.

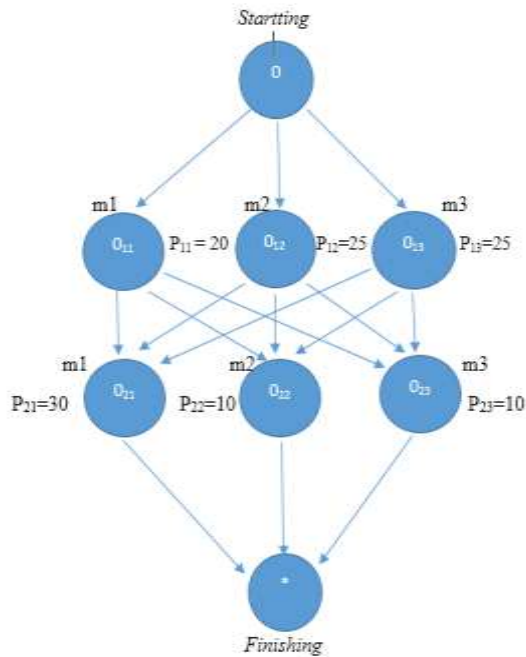
2. Implementasikan *ant colony algorithm* untuk proses penjadwalan produksi

4. HASIL DAN PEMBAHASAN

Berdasarkan penelitian yang sudah dilakukan di perusahaan CV. Inda Jaya, terdapat dua kali proses yaitu tahapan yang pertama adalah mengolah bahan mentah yang diproduksi menjadi bahan setengah jadi, kemudian bahan tersebut bisa diproduksi secara simultan pada tahapan proses kedua yaitu tahapan desain dan *matching size* memperhatikan mesin yang sedang *available* maka proses produksi bisa langsung dilakukan secara bersama-sama.

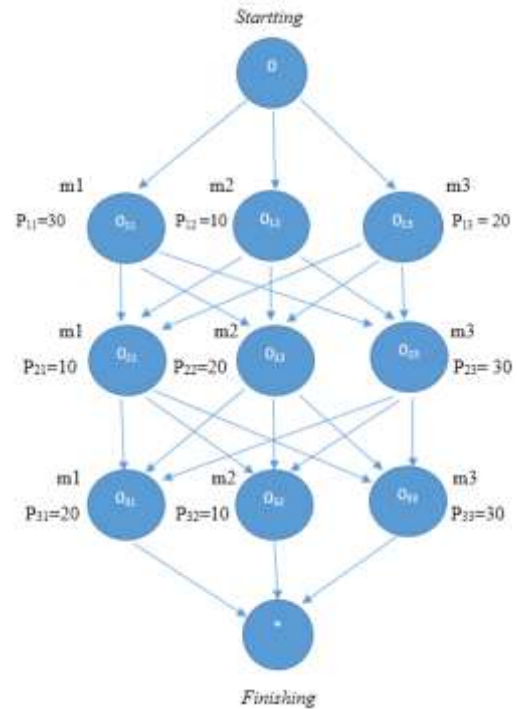
Pada tahap pertama terdapat 2 *job* dengan menggunakan 3 mesin, dalam setiap *job* akan diproses oleh 3 mesin tersebut, hasil akhir dari tahap pertama ini adalah bahan semi jadi yang akan diproses kedalam tahap dua yaitu desain dan *matching size*, pada tahap kedua bahan setengah jadi memasuki proses desain ataupun *matching size* dengan mesin yang berbeda, ada 3 *job* dengan menggunakan 3 mesin. Pada tahapan ini memerlukan data – data mulai dari waktu tiap pemrosesan (*Processing Time*) sebagai variabel untuk mencari *makespan time* paling kecil, pada proses ini data diolah menggunakan graf, proses tersebut dilakukan untuk merepresentasikan data *flow shop* dimana

data tersebut dimasukan kedalam *node* – *node* pada *data flow* yang mana satu dengan lainnya saling berhubungan sehingga dapat digunakan untuk menentukan rute terpendek dari proses tersebut.



Gambar 2. Graph Proses Produksi Pada Tahap 1

Pada gambar 2, *graph* tersebut merepresentasikan proses produksi secara umum pada bahan mentah menjadi bahan setengah jadi, kemudian dilanjutkan dengan *graph* yang merepresentasikan proses produksi secara umum pada tahap kedua ditunjukkan pada gambar 3.



Gambar 3. Graph Proses Produksi Pada Tahap Kedua

Pada penjadwalan proses produksi menggunakan *ant colony algorithm* ada beberapa parameter untuk menghitung nilai *makespan* ada sembilan *job* pada proses tahap pertama yang bisa dihitung nilai *makespan*nya seperti pada tabel 1.

Tabel 1. Perhitungan Waktu Tahap 1

Ket	Mesin 1	Mesin 2	Mesin 3	Total (Menit)
J1	20	25	25	70
J2	30	10	10	50
Total				120

Pada tabel 1 menunjukkan proses produksi pada CV. Inda Jaya menggunakan 3 mesin dengan jumlah produksi atau *job* sebanyak 2 *job* yang berbeda, pada tabel 2,

data yang digunakan untuk acuan proses produksi masing – masing *job* dijabarkan dari mulai awal pengerjaan hingga akhir tiap *job* yang ada pada tahap 1.

Tabel 2. Peghitungan *Makespan* Tahap 1

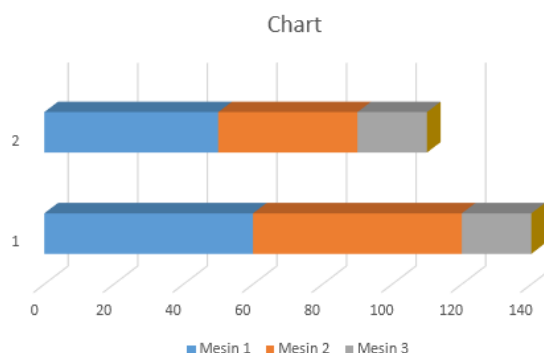
Ket	Mesin 1		Mesin 2		Mesin 3	
	aw	ak	aw	ak	aw	ak
J1	0	20	20	45	45	70
J2	20	50	50	60	60	70

Tabel 2 menunjukkan pemrosesan data dari proses produksi, dari tabel 2 diperoleh hasil seluruh waktu yang digunakan (*completion time*) menggunakan urutan kerja mesin 1-2-3 terdapat beberapa hasil waktu penyelesaian pada mesin 1 total waktu yang digunakan adalah 70 menit, kemudian pada mesin kedua selesai dengan jumlah 105 menit dan pada mesin ketiga menghasilkan waktu sebanyak 140 menit, artinya urutan tersebut jika diimplementasikan di proses produksi maka akan terjadi pemborosan waktu sebanyak 20 menit.

Tabel 3. *Makespan* Urutan *Job* Tahap 1

No	Urutan	Makespan
1	1-2-3	140
2	1-3-2	125
3	2-1-3	140
4	2-3-1	110
5	3-1-2	125
6	3-2-1	110

Pada tabel 3 perhitungan *makespan* dilakukan pada tiap iterasi dan didapatkan hasil batas bawah yang paling minimal adalah pada iterasi ke 4 dan iterasi ke 6, dari semua *job* urutan, *gant chart* pada tahap pertama ditunjukkan pada gambar 4.



Gambar 4. *Gantt Chart* Penjadwalan Produksi Tahap 1

Kemudian perhitungan *makespan* pada tahap 2 dimana pada perhitungan tersebut terdapat tiga *job* yang akan dikerjakan, tabel 4 merupakan perhitungan tiap iterasi yang dikerjakan pada tahap kedua.

Tabel 4. Perhitungan *makespan* Tahap 2

Ket	Mesin 1	Mesin 2	Mesin 3	Total (Menit)
J1	30	50	20	100
J2	20	45	25	90
J3	15	15	20	50
Total				240

Tabel 4 menunjukkan proses *finishing* yang pada tahap kedua pada CV. Inda Jaya dengan menggunakan 3 mesin dan memproduksi 12 produk dengan tiga varian

produk, artinya terdapat 4 proses yang dikerjakan sekaligus dan terdapat 3 tahapan (Mesin 1, Mesin 2 dan Mesin 3) yang dipakai, tahapan tersebut adalah tahap implementasi desain, tahap *joining* dan tahap *packaging*, data waktu yang digunakan masing-masing *job* ditunjukkan pada Tabel 5.

Tabel 5. Proses Tahap 2

Ket	Waktu Mesin			Total Waktu
	Mesin 1	Mesin 2	Mesin 3	
J1	30	10	20	60
J2	10	20	30	60
J3	20	10	30	60
Total				180

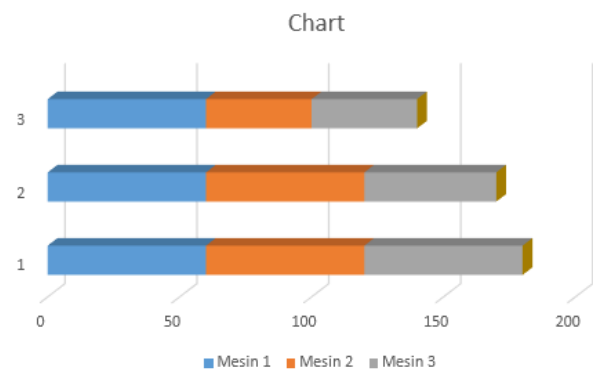
Tabel 5 menunjukkan proses produksi pada tahap kedua yang membutuhkan waktu pada mesin 1 sebanyak 60 menit, mesin kedua sebanyak 40 menit dan pada mesin ketiga sebanyak 80 menit, kemudian pada tahap kedua dilakukan perhitungan *makespan* yang ditunjukkan pada tabel 6.

Tabel 6. *Makespan* Tahap 2

No	Urutan	Makespan
1	1-2-3	180
2	1-3-2	170
3	2-1-3	140
4	2-3-1	160
5	3-1-2	170

6	3-2-1	160
7	1-2-3	180
8	1-3-2	170
9	2-1-3	140

Proses pada tabel 6 menunjukkan *makespan* yang paling kecil adalah 140 yaitu pada iterasi 3 dan iterasi 9 dari semua *job* urutan pada tiap iterasi yang dilakukan *makespan* yang paling minimal untuk bisa meminimalisir waktu produksi ada pada iterasi ke 3 karena menggunakan langkah yang lebih sedikit sehingga lebih optimal, *gantt chart* pada tahap kedua ditunjukkan pada gambar 5.



Gambar 5. *Gantt Chart* Tahap 2

Gambar 5 menunjukkan hasil pada tahap kedua dalam bentuk *gantt chart* yang menunjukna proses paling minimal sebanyak 140 menit.

4. KESIMPULAN

1. Sistem Informasi ini bisa menghitung *makespan* paling kecil, pada tahap pertama adalah sebesar 110 menit dan tahap kedua adalah sebesar 140 menit,

Dedy Kurniadi, Sam Farisa Chaerul Haviana

hasil tersebut bisa mengoptimalkan proses produksi.

2. Implementasi proses penjadwalan dengan menggunakan ant colony algorithm bisa dijadikan sebagai proses penjadwalan alternatif untuk produksi berdasarkan urutan dengan makespan paling kecil.

5. DAFTAR PUSTAKA

- Alamsyah, 2010. Pemanfaatan metode heuristik dalam pencarian minimum spanning tree dengan algoritma semut. *Majalah Ilmiah Mektek* 3.
- Dewi, I.M. & Abusini, S., 2010. Penjadualan Produksi Flow Shop Menggunakan Algoritma Modified Ant Colony (Studi Kasus Penjadualan Pengalengan Ikan PT . Indo Bali di Negara Bali), e-journal *Teknik elektro*, 1–4
- Hidayat, T., 2007. Pencarian Jalur Terpendek Menggunakan Algoritma Semut, *Seminar Nasional Aplikasi Teknologi Informasi*, 81-84
- Kambey, F.D., 2011. Optimasi pada Rute Truk Peti Kemas. , e-journal *teknik elektro dan komputer*, 7–11.
- Yuwono, B, Aribowo, A.S dan Wardoyo, S.B., 2009. Implementasi Algoritma Koloni Semut Pada Proses, *Seminar Nasional Informatika*, 111–120.