

Usulan Penjadwalan Mesin Paralel Menggunakan Metode *Ant Colony Optimization Algorithm* dan *Longest Processing Time*

Proposed Scheduling of Parallel Machines Using Ant Colony Optimization Algorithm and Longest Processing Time Methods

Evi Febianti^{1*}, Yusraini Muharni², Deni Falti³, Lely Herlina⁴, Kulsum⁵

^{1,2,3,4,5}Jurusan Teknik Industri, Universitas Sultan Ageng Tirtayasa, Banten, Indonesia

*Penulis Korespondensi: Evi Febianti, evi@untirta.ac.id

Abstrak

Penjadwalan produksi sangat penting diterapkan pada perusahaan yang menggunakan sistem *make to order*, yaitu produk akan diproduksi sesuai pesanan konsumen. PT ABC adalah perusahaan yang bergerak dibidang konstruksi, fabrikasi dan alat berat. Permasalahan yang terjadi adalah perusahaan sering mengalami kesulitan dalam memenuhi permintaan konsumen secara tepat waktu. Pola produksi pada perusahaan ini adalah *flow shop* dengan setiap pekerjaan memiliki urutan produksi yang sama. Penelitian ini bertujuan untuk mengoptimalkan penjadwalan produksi agar mendapatkan waktu paling minimum untuk menyelesaikan setiap part yang akan di produksi. Metode yang digunakan adalah metode LPT (*Longest Processing Time*) dan metode ACO (*Ant Colony Optimization*) dengan bantuan software MATLAB. Hasil yang didapatkan menunjukkan bahwa metode eksisting memperoleh nilai *makespan* sebesar 9.118 menit dan metode usulan dengan menggunakan LPT dan ACO memperoleh nilai *makespan* sebesar 8.394 menit. Sehingga metode usulan memiliki nilai lebih baik dan dapat diterapkan dalam penjadwalan produksi PT ABC.

Kata kunci: *ant colony optimization algorithm, longest processing time, mesin paralel, penjadwalan flow shop*


Abstract

Production scheduling is critical to a company that has *make-to-order* systems because they produce products based on customer demand. PT ABC is a company engaged in construction, fabrication and heavy equipment. The problem is that companies often need help meeting consumer demand promptly. The production pattern in this company is a *flow shop*, with each job having the same production order. This study aims to optimize production scheduling for the minimum time to complete each production part. The methods used are the LPT (*Longest Processing Time*) method and the ACO (*Ant Colony Optimization*) method with the help of MATLAB software. The results showed that the existing process obtained a *makespan* value of 9,118 minutes, and the proposed method using LPT and ACO obtained a *makespan* value of 8,394 minutes so that the proposed method has better value and can be applied in PT ABC's production scheduling

Keywords: *ant colony optimization algorithm, flow shop scheduling, longest processing time, parallel machine.*

How to Cite:

Febianti, E. *et al.* (2023) 'Usulan penjadwalan mesin paralel menggunakan metode Ant Colony Optimization Algorithm dan Longest Processing Time', *Journal of Integrated System*, 6(1), pp. 42–52. Available at: <https://doi.org/10.28932/jis.v6i1.5610>.

© 2023 Journal of Integrated System. This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License. 

1. Pendahuluan

Perkembangan teknologi pada saat ini sangat ketat dan bukan lagi merupakan suatu halangan bagi suatu perusahaan untuk menerapkan teknologi maju. Perusahaan dituntut untuk dapat memproduksi suatu produk yang baik dan tepat waktu. Perusahaan juga perlu memperhatikan permintaan konsumen yang tinggi akan suatu produk, sehingga perlu diperhatikan proses produksi baik dari sisi waktu produksi dan kualitas barang jadi (Febianti *et al.*, 2020).

Penerapan optimalisasi dan efisiensi produksi pada suatu perusahaan sangat diperlukan sehingga perusahaan dapat melakukan perbaikan yang berkelanjutan dan bertahap pada masing-masing bagian agar dapat memenuhi permintaan konsumen menjadi lebih baik (Panjaitan dan Ginting, 2019). Bagi perusahaan yang menjadi perhatian salah satunya adalah mengenai penjadwalan. Penjadwalan ialah suatu proses pengaturan sumber daya untuk menyelesaikan tugas-tugas dengan melibatkan pekerjaan, sumber daya, serta waktu tertentu (Baker dan Trietsch, 2018; Pinedo, 2016). PT ABC merupakan sebuah perusahaan termuka di Indonesia yang bergerak pada bidang alat berat, konstruksi, dan fabrikasi, dengan sistem produksinya adalah *make to order*. Dalam menjalankan aktivitas produksi, PT ABC mempunyai jenis mesin yang tersusun paralel dan menggunakan susunan jumlah yang berbeda.

Permasalahan yang terjadi adalah pada saat pembuatan jembatan *Truss Bridge B-60*, terdapat keterlambatan produksi, meskipun sudah ditentukan setiap *part* yang akan dibuat, serta waktu proses yang sudah direncanakan, namun adanya masalah seperti material dari vendor yang datang terlambat mengakibatkan kerugian dan perlunya pengaturan ulang sumber daya yang ada untuk kegiatan produksi tersebut. Pada saat pengerjaan dan pengiriman produk diperlukan suatu penjadwalan untuk mengatur proses produksi agar sesuai waktu yang telah di sepakati dengan konsumen (Demir, 2020).

Pendekatan metode LPT (*Longest Processing Time*) dipilih karena mesin-mesin yang ada di PT. ABC tersusun paralel dan penelitian ini dibatasi hanya menjadwalkan 5 job dari 26 job sehingga tidak ada data *due date*. Pada penjadwalan menggunakan algoritma ACO (*Ant Colony Optimization*) karena menurut (Liliani, 2014) ACO 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.

Pada penelitian sebelumnya menggunakan metode LPT, CEGA dan PSO menghasilkan nilai *makespan* terkecil dan yang direkomendasikan adalah menggunakan LPT (Muharni *et al.*, 2020). Penelitian lain (Panjaitan dan Ginting, 2019) menggunakan metode *Shortest Processing Time* (SPT) dan ACO menghasilkan pada metode SPT menghasilkan *makespan* sebesar 236079,89 detik, sedangkan pada algoritma ACO menghasilkan nilai *makespan* 215243,22 detik dan yang terpilih adalah menggunakan metode ACO karena memiliki *makespan* terkecil. Pada penelitian ini menggunakan metode LPT (*Longest Processing Time*) dan metode ACO (*Ant Colony Optimization*) (Della Croce and Scatamacchia, 2020; Demir, 2020). Penelitian ini menggunakan dua metode yaitu LPT dan ACO yang akan dibandingkan dengan eksisting yang ada pada perusahaan yaitu menjadwalkan sesuai urutan pesanan. *Ant Colony Optimization* merupakan teknik *probabilistic* untuk menjawab masalah komputasi yang bisa dikurangi dengan menemukan jalur terbaik (Blum, 2005; Dorigo dan Blum, 2005). Aplikasi dari *ant colony optimization* sering digunakan untuk mencari solusi masalah optimasi. Adapun tujuan dari penelitian ini adalah memberikan usulan penjadwalan pada proses produksi *Steel Bridge (SB) B-60*, mencari nilai *makespan* minimum dengan metode usulan LPT dan ACO serta mengetahui perbandingan nilai *makespan* yang didapat dari metode perusahaan dan metode usulan.

2. Tinjauan Pustaka

2.1 Penjadwalan Produksi

Penjadwalan pada dasarnya adalah rencana eksekusi jangka pendek dari model perencanaan produksi. Penjadwalan produksi terdiri dari kegiatan yang dilakukan pada perusahaan manufaktur untuk mengelola dan mengontrol pelaksanaan proses produksi (Lopez dan Roubellat, 2010; Fera *et al.*, 2013). Penjadwalan produksi berfungsi untuk membuat agar arus produksi dapat berjalan lancar sesuai dengan waktu yang telah direncanakan., sehingga dapat dikatakan bahwa penjadwalan produksi dilakukan agar mesin-mesin dapat bekerja sesuai dengan kapasitas yang ada dan biaya yang seminimal mungkin, serta kuantitas produk yang diinginkan sesuai waktu yang telah ditentukan (Muharni *et al.*, 2019)

2.2 Metode *Longest Processing Time* (LPT)

Metode LPT merupakan salah satu metode penjadwalan yang termasuk ke dalam penjadwalan *heuristic*. Metode LPT sendiri memiliki definisi yaitu metode yang mengurutkan *job* dengan waktu pengerjaan terpanjang terlebih dahulu kemudian untuk diselesaikan dan dijadikan prioritas utamanya. Metode LPT juga dapat digunakan pada kasus penjadwalan mesin yang disusun secara parallel (Muharni *et al.*, 2020).

Penggunaan LPT diawali dengan mengidentifikasi *job order* dan notasi variabel yang dipergunakan untuk menerjemahkan variabel apa saja yang mempengaruhi proses penjadwalan produksi (Della Croce dan Scatamacchia, 2020; de Abreu dan Fuchigami, 2022). Adapun notasi variabel yang digunakan adalah :

i : Nomor *job order* yang akan diproduksi

m : Mesin yang tersedia

k : Mesin identik ke *k* dari jumlah mesin *m* yang tersedia

p : Total waktu proses untuk *job order* *i*

s : *List* seluruh *job* yang akan dijadwalkan

Kemudian mengurutkan pekerjaan berdasarkan waktu proses terpanjang sesuai dengan aturan prioritas menggunakan LPT. *Job order* yang memiliki total waktu proses terpanjang ditempatkan sebagai urutan pertama. Tahap selanjutnya menjadwalkan ke setiap mesin (*k*) terpilih berdasarkan tipe aliran mesin yaitu *flowshop* yang artinya *m1* merupakan mesin yang dijadwalkan pertama untuk *job*, jika *m1* telah selesai dilanjutkan ke *m2*, begitu seterusnya dan urutan *m* tidak boleh acak (*m1*, *m2*, *m3*..., dst).

Selanjutnya *job order* yang telah dijadwalkan ke setiap mesin dikeluarkan dari *list s* sehingga akan terlihat *job* mana saja yang masih tersisa di *list s*. Tahap pemeriksaan pada *list s* untuk mengetahui apakah *list s* sudah kosong atau belum dari urutan *job order*. Jika nilai *list s* sudah 0 (nol) maka selesai, tetapi jika belum mendapatkan nilai 0, maka dilanjutkan ke tahap pengurutan proses waktu terpanjang pada *list s*.

2.3 Metode *Ant Colony Optimization* (ACO)

Ant Colony Optimization (ACO) merupakan salah satu bagian dari metode *metaheuristic*. *Metaheuristic* adalah pendekatan komputasi untuk mencari solusi optimal atau mendekati optimal dari suatu masalah optimisasi dengan cara mencoba secara iteratif untuk memperbaiki kandidat solusi dengan memperhatikan batasan kualitas solusi yang diinginkan. Suatu teknik *metaheuristic* memberi kerangka umum untuk memecahkan masalah. Teknik *metaheuristic* dapat digunakan untuk menyelesaikan berbagai persoalan yang berbeda dengan langkah-langkah yang hampir sama. *Ant Colony Optimization* (ACO) diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut. Semut mampu mengindra lingkungannya yang kompleks untuk mencari makanan dan kemudian kembali ke sarangnya dengan meninggalkan zat *Pheromone* pada rute-rute yang mereka lalui (Dorigo dan Blum, 2005).

Pengolahan data dengan menggunakan ACO, pada tahap pertama menentukan parameter α , β , ρ mengikuti nilai yang telah ditentukan (Dorigo dan Blum, 2005; Panggabean, 2018; Rui *et al.*, 2014)

1. Intensitas jejak semut antar titik dan perubahannya.
2. Banyak titik (n) termasuk koordinat (x,y) atau jarak antar titik (d_{ij}).
3. Titik awal dan titik tujuan.
4. Tetapan siklus semut (Q).
5. Tetapan pengendali intensitas jejak semut (α), nilai $\alpha \geq 0$.
6. Tetapan pengendali visibilitas (β), nilai $\beta \geq 0$.
7. Visibilitas antar titik = $1/d_{ij}$ (η_{ij}).
8. Banyak semut (m).
9. Tetapan penguapan jejak semut (ρ), nilai ρ harus > 0 dan < 1 untuk mencegah jejak pheromone yang tak berhingga.
10. Jumlah siklus maksimum (N_{cmax}) bersifat tetap selama algoritma dijalankan, sedangkan τ_{ij} akan selalu diperbaharui harganya pada setiap siklus algoritma mulai dari siklus pertama ($NC = 1$) sampai tercapai jumlah siklus maksimum ($NC = NC_{max}$) atau sampai terjadi konvergensi.

Dilanjutkan dengan langkah inisiasi. Kemudian tahap berikutnya adalah dimana semut yang telah diletakan pada masing-masing node melakukan *tour* dan menghasilkan sebuah *path* atau rute. Selanjutnya dilakukan pencarian perolehan probabilitas terbesar dan *pheromone* terkecil pada *path* untuk dijadikan rute terbaik. Pada tahap ini dilakukan perhitungan *makespan* untuk masing-masing *job* dan dipilih penjadwalan yang memiliki *makespan* terkecil. *Stopping criteria* dilakukan jika diketahui bahwa faktor terminasi $<$ faktor terminasi maksimum, dan perlu *update pheromone*. Jika tidak maka lakukan kembali ke tahap menentukan jumlah semut atau inisialisasi. Pada tahap *update pheromone* jika diketahui bahwa faktor terminasi, adakah faktor terminasi maksimum.

3. Hasil dan Pembahasan

Pada penelitian ini menggunakan data primer yang diperoleh dari pengamatan langsung dan data sekunder berupa data perusahaan yang mendukung penelitian. Data tersebut berupa data dimensi produk (*job*), data jumlah mesin, data waktu *setup*, data waktu proses dan data total mesin tersedia. Untuk pengolahan data teknik analisis yang digunakan menggunakan metode *Longest Processing Time* (LPT) dan *Ant Colony Optimization* (ACO) dengan bantuan *software MATLAB*.

Pada penelitian ini data produk *SB B-60* yang difokuskan untuk diteliti terdapat lima jenis *part* yaitu *Cross Grider* (CG) sebagai *job 1*, *Top Chord* (TC) sebagai *job 2*, *Bottom Chord* (BC) sebagai *job 3*, *Diagonal Chord* (DG) sebagai *job 4*, dan *Stringer* (ST) sebagai *job 5*. Berikut ini merupakan waktu total untuk setiap *job* yang akan dijadwalkan, tersaji pada Tabel 1.

Tabel 1. Rekapitulasi waktu total untuk setiap *job*
 Mesin (menit)

<i>Job</i>	<i>Cutting Machine</i>	<i>Fit-Up Machine</i>	<i>Welding LMH</i>	<i>Straightening</i>	<i>Fit-Up Fab</i>	<i>Welding</i>	<i>Finishing</i>
TB60-CG	89,06	131,84	419,34	155,12	284,79	288,6	76,38
TB60-TC	150,7	220,96	638,86	262,28	478,7	487,04	130,84
TB60-BC	160,56	244,12	707,76	279,16	523,8	526,6	135,08
TB60-DG	323,4	484,52	1545	562,24	1056,68	1058,92	274,64
TB60-ST	565,04	751,9	2431,32	975,14	1679,98	1678,08	313,4

Sumber: PT ABC

Adapun mesin yang terpakai dari jumlah mesin tersedia, dan juga mesin-mesin yang ada disusun secara paralel untuk proses produksinya. Berikut ini merupakan tabel data jumlah mesin yang terpakai untuk memproduksi *SB B-60* di PT ABC, tersaji pada Tabel 2.

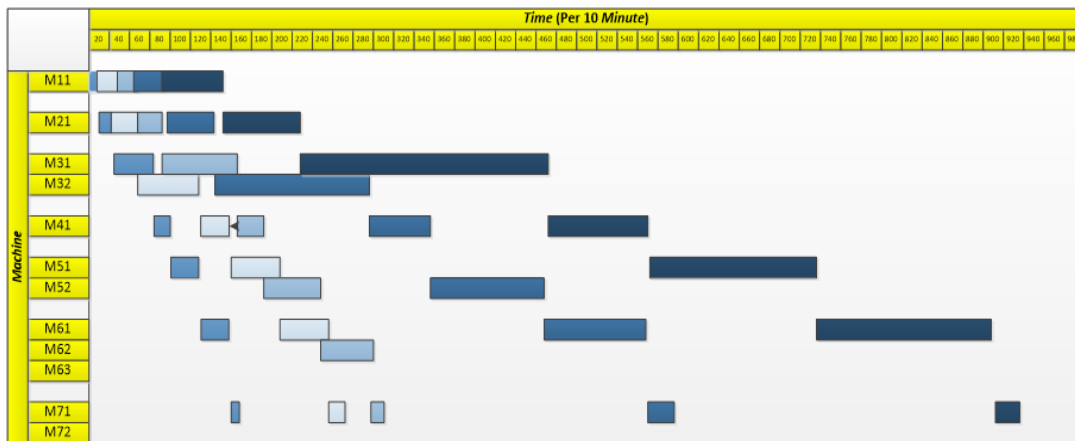
Tabel 2. Data Jumlah Mesin Yang Terpakai

No.	Nama Mesin	Jumlah Mesin
1	<i>Cutting</i>	1
2	<i>Fit-Up Machine</i>	1
3	<i>Welding LMH</i>	2
4	<i>Straightening</i>	1
5	<i>Fit-Up Fab</i>	2
6	<i>Welding</i>	3
7	<i>Finishing</i>	2

Sumber: PT ABC

3.1 Hasil Penjadwalan Eksisting

Berikut ini merupakan urutan *job* pada kondisi eksisting di PT ABC yang memiliki urutan *job* 12-3-4-5, tersaji pada Gambar 1 dan dapat diketahui bahwa *makespan* penjadwalan eksisting sebesar 9.118 menit



Gambar 1. Hasil *gant* chart penjadwalan eksisting

Keterangan:

TB60-CG	
TB60-TC	
TB60-BC	
TB60-DG	
TB60-ST	

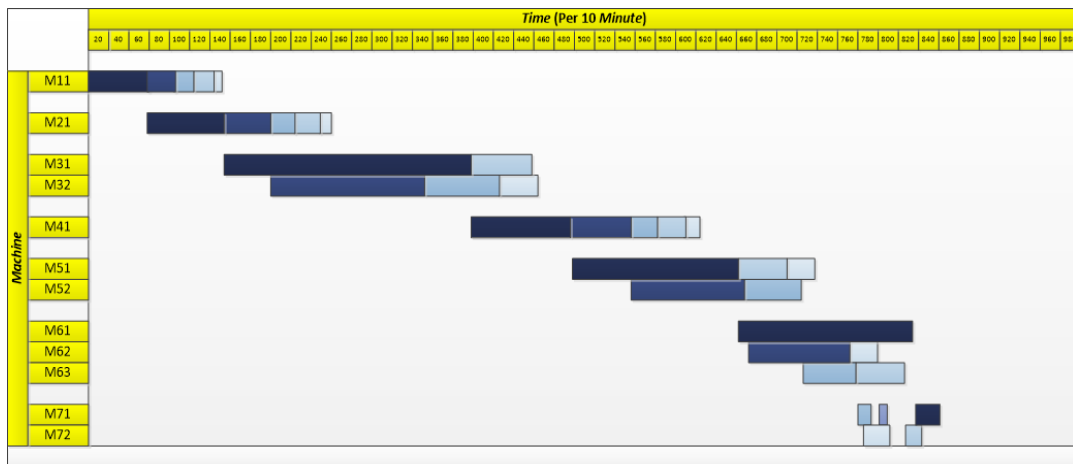
3.2 Penjadwalan Menggunakan Metode *Longest Processing Time* (LPT)

Dari hasil total data waktu proses dan tahap-tahap dalam pengerjaan penjadwalan berdasarkan metode LPT yang sudah dijelaskan sebelumnya, maka diperoleh urutan *job* metode LPT yaitu 5-4-3-2-1. Berikut hasil *list s* metode *Longest Processing Time* (LPT) tersaji pada Tabel 3. Berdasarkan Tabel 3, didapatkan hasil waktu total terpanjang dimiliki oleh TB60-ST, lalu waktu total terpanjang kedua TB60-DG, waktu total terpanjang ketiga TB60-BC, waktu total terpanjang keempat TB60-TC dan waktu total terpanjang terakhir TB60 CG. Maka berdasarkan waktu total di atas diketahui bahwa $p_5 > p_4 > p_3 > p_2 > p_1$ adalah urutan dalam *list job* yang akan dijadwalkan.

Tabel 3. List S metode *Longest Processing Time* (LPT)

Job	Mesin (menit)							Total Time (pembulatan)
	Cutting Machine	Fit-Up Machine	Welding LMH	Straightening	Fit-Up Fab	Welding	Finishing	
TB60-ST	565.04	751.90	2431.32	975.14	1679.08	1678.08	313.40	8394
TB60-DG	323.40	484.52	1545.00	562.24	1056.68	1058.92	274.64	5305
TB60-BC	160,56	244.12	707,76	279,16	523,8	526,6	135,08	2586
TB60-TC	150.70	220.96	638.86	262.28	478.70	487.04	130.84	2369
TB60-CG	89.06	131.84	419.34	155.12	284.79	288.60	76.38	1445

Hasil dari gantt chart pada Gambar 2 dapat diketahui bahwa menggunakan metode *dispatching priority rules* LPT, bahwa *makespan* penjadwalan yaitu sebesar 8.394 menit. Berikut ini merupakan *gantt chart* metode LPT, tersaji pada Gambar 2.



Gambar 2. *Gantt chart* penjadwalan metode LPT

3.3 Penjadwalan dengan Algoritma *Ant Colony Optimization* (ACO)

Pada penelitian ini metode ACO menggunakan bantuan *software* MATLAB dengan hasil pengujian algoritma didapatkan dari hasil *running coding* yang telah dibuat dan akan dilakukan proses *running* sebanyak 3 kali dengan masing-masing 10 iterasi untuk mendapatkan hasil yang berbeda-beda untuk mendapatkan solusi yang paling optimal (Sivakumar *et al.*, 2015; Luo *et al.*, 2020). Parameter jumlah siklus yaitu 50 siklus. Hal ini menunjukkan bahwa semakin banyak jumlah siklus, maka nilai *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 (Liliani, 2014). Sedangkan untuk parameter jumlah semut yaitu 7 semut. Ketika jumlah semut semakin banyak, maka alternatif solusi yang dapat dipilih pada tiap iterasi semakin banyak (Liliani, 2014). Berikut ini *pseudocode* atau panduan untuk membuat *coding* pada *software* MATLAB. Hasil MATLAB tersaji pada Gambar 3.

Berikut ini merupakan langkah-langkah dalam metode algoritma *Ant Colony Optimization* dengan menggunakan bantuan *software* MATLAB.

1. Penentuan Parameter Awal

Tahapan parameter awal diperlukan dalam pengolahan data yang digunakan dan tidak ada aturan pasti mengenai penentuan nilai parameter ini. Adapun hasil untuk penentuan parameter awal setelah proses *running* pertama tersaji pada Gambar 4.

```
Algoritma ACO
Output : rute terbaik,
PenentuanParameterAwal
Input : jumlah semut, jumlah job, matriks proses, hingga menentukan
        alpha, beta, intensitas pheromone (t) didapatkan dari rumus
        txy=0.01*rand(1, jumlahPekerjaan)

inisialisasi
hitung nilai visibility yang dibutuhkan.

PerhitunganProbabilitas
Hitung probabilitas setiap operasi yang terpilih pada job.
xy = zeros(1, jumlahjob);
xz = zeros(1, jumlahjob);
for j=1:jumlahjob
    if j == 1
        elseif j == 5
        else
    end
end
end
nProb(1,:) = xy./sum(xz);
nProb(2,:) = xz./sum(xy);
sumnProb = [sum(nProb); sum(nProb)];
probabilitas = nProb./sumnProb

kalkulasiposisi&penentuanRuteTerbaik
Mencari posisi terbaik dan menemukan hasil makespan terkecil

StoppingCriteria
if faktor terminasi < faktor terminasi max,
    maka kembali ke langkah 2
end if
if faktor terminasi > faktor terminasi max,
    maka tampilkan hasil makespan terkecil.
end if
selisihTimeSpan = abs(timeSpan - timeSpanSebelumnya);
timeSpanSebelumnya = timeSpan;
end
```

Gambar 3. Pseudocode penjadwalan algoritma ACO

2. Inisialisasi

Tahapan kedua melakukan proses inisialisasi. Inisialisasi merupakan tahap penentuan awal bagi *pheromone* pada waktu ke-t. Pada penelitian ini posisi awal dari kondisi eksisting yaitu 1-2-3-4-5 kemudian menghitung hasil *timespan* yang didapatkan dari posisi tersebut tersaji pada Gambar 5.

3. Perhitungan Probabilitas

Tahapan ketiga melakukan proses perhitungan probabilitas. Menghitung probabilitas setiap operasi yang terpilih. Berikut hasil yang diperoleh untuk tahapan ini setelah proses *running* tersaji pada Gambar 6.

4. Kalkulasi Posisi

Tahapan keempat melakukan proses kalkulasi posisi. Pada tahap ini dilakukan kalkulasi posisi menghitung nilai *makespan* untuk masing-masing *job* dan dipilih penjadwalan yang memiliki hasil *makespan* terkecil. Berikut hasil yang diperoleh untuk tahapan ini tersaji pada Gambar 7.

5. Penentuan Rute Terbaik

Tahapan kelima melakukan penentuan rute terbaik. Dimana menghitung berdasarkan nilai *makespan* terkecil dan memperbaiki jalan yang dipilih oleh *pheromone* sehingga mendapatkan posisi rute terbaik. Berikut hasil yang diperoleh untuk tahapan ini tersaji pada Gambar 8.

6. Stopping Criteria

Tahapan yang terakhir yaitu *stopping criteria*. Pada tahapan ini melakukan penentuan rute terbaik yang dilihat dari faktor terminasi. Jika faktor terminasi $\max < \text{faktor terminasi max}$, maka kembali ke langkah 2. Jika tidak, maka akan menghasilkan *makespan* terkecil yang tersaji pada Gambar 9.

Selain itu, dapat dilihat pergerakan nilai *makespan* dari kombinasi penjadwalan dengan metode ACO untuk 10 iterasi, tersaji pada Gambar 10.

```
1. Penentuan Nilai Parameter Inisiasi
Alpha = 1.00000
Beta = 2.00000
Faktor terminasi max = 1.00000
```

Gambar 4. Hasil *running* MATLAB pertama untuk penentuan parameter awal

```
2. Inisialisasi
Sample inisial = 1-2-3-4-5
Timespan inisial = 9.117680e+03
```

Gambar 5. Hasil *running* MATLAB pertama untuk inisialisasi

```
3. Perhitungan Probabilitas
0.0111 0.0001 0 0.0222 0
0.9889 0.9999 1.0000 0.9778 1.0000
```

Gambar 6. Hasil *running* MATLAB pertama untuk perhitungan probabilitas

```
4. Kalkulasi Posisi
posisi baru : 5-3-4-1-2
makespan : 8.393960e+03
```

Gambar 7. Hasil *running* MATLAB pertama tahapan kalkulasi posisi

```
5. Penentuan Rute Terbaik
posisi baru : 5-3-4-1-2
```

Gambar 8. Hasil *running* MATLAB pertama tahapan rute terbaik

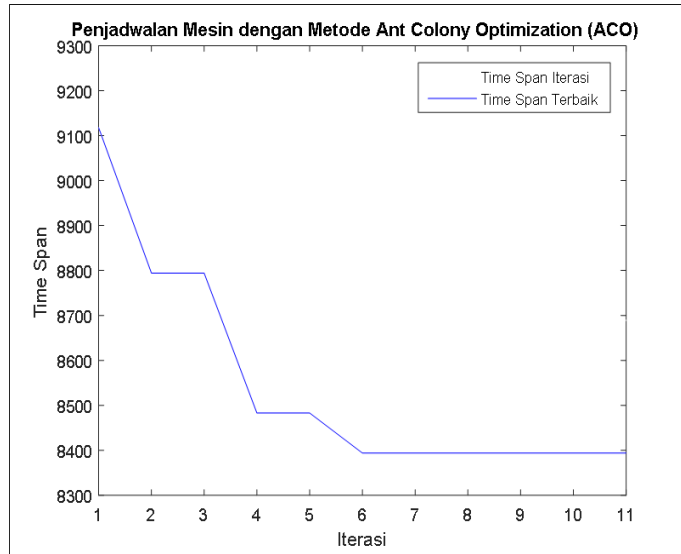
```
6. Stopping Criteria
Faktor terminasi = 1.605600e+02
```

Gambar 9. Hasil *running* MATLAB pertama tahapan *stopping criteria*

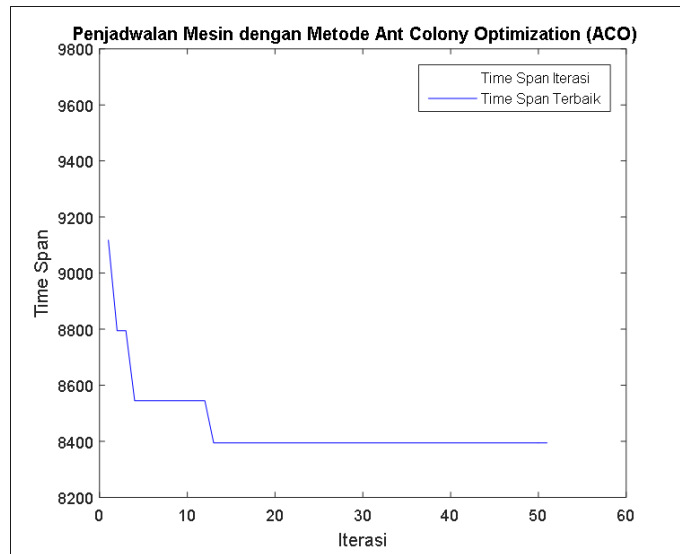
Berdasarkan Gambar 10 bahwa iterasi 1 dengan inisialisasi jadwal 1-2-3-4-5 dengan hasil *makespan* sebesar 9100 jam, Iterasi 2 terjadi penurunan nilai *makespan* menjadi 8800 jam, pada iterasi 3 tidak terjadi perubahan nilai *makespan*. Pada saat iterasi 4 mengalami penurunan *makespan* menjadi 8500 jam, pada iterasi 5 tidak mengalami perubahan kembali. Pada iterasi 6 mengalami penurunan *makespan* dengan nilai 8394 jam hingga iterasi 10 mempunyai nilai *makespan* tetap konstan dan tidak ada perubahan kembali dan menghasilkan *makespan* terkecil. Berikut ini merupakan gambar hasil dari iterasi 50 untuk membandingkan dengan hasil iterasi 10 tersaji pada Gambar 11.

Berdasarkan hasil yang diperoleh pada iterasi 1 grafik mengalami penurunan sampai iterasi 12 dan kemudian iterasi 12 hingga iterasi ke 50 mempunyai nilai *makespan* tetap konstan dan tidak ada perubahan kembali yang menghasilkan *makespan* terkecil yaitu 8.394 menit. Bahwa

tidak adanya perubahan yang signifikan pada iterasi 10 ataupun menggunakan iterasi 50, yang dapat dilihat dari hasil nilai *makespan*.



Gambar 10. Grafik penjadwalan metode ACO hasil *running* MATLAB iterasi 10



Gambar 11. Grafik penjadwalan metode ACO hasil *running* MATLAB iterasi 5

3.4 Perbandingan Nilai *Makespan* Kondisi Eksisting dan Metode Usulan

Berikut ini merupakan tabel perbandingan nilai *makespan* yang didapatkan dari ketiga metode penjadwalan. Berdasarkan hasil pada Tabel 3 bahwa metode usulan yaitu LPT dan ACO memiliki nilai *makespan* yang baik.

Tabel 3. Perbandingan nilai *makespan*

Metode Penjadwalan	Nilai <i>Makespan</i> (Menit)
Eksisting	9.118
LPT (<i>Longest Processing Time</i>)	8.394
ACO (<i>Ant Colony Optimization</i>)	8.394

4. Kesimpulan

Kesimpulan yang dihasilkan dari penelitian ini adalah sebagai berikut. Usulan penjadwalan *flow shop* dalam proses produksi SB B-60 di PT ABC menggunakan metode LPT (*Longest Processing Time*). Hasil nilai *makespan* minimum dari proses produksi SB B-60 dengan metode *Longest Processing Time* (LPT) dan *Ant Colony Optimization* (ACO) lebih kecil dibandingkan dengan nilai *makespan* metode eksisting dengan selisih perbedaan 724 menit. Nilai *makespan* yang didapat dari metode *eksisting* sebesar 9.118 menit dan nilai *makespan* metode usulan yaitu LPT dan ACO sebesar 8.394 menit.

Keterbatasan penelitian ini belum melibatkan waktu perawatan mesin, sehingga dapat menjadi usulan penelitian lebih lanjut dengan mempertimbangkan waktu perawatan mesin. Serta dapat menggunakan metode usulan yang lain seperti metode *Hybrid Particel Swarm Optimization-Ant Colony Optimization* (PSO-ANT) supaya didapatkan perbandingan hasil yang lebih optimal.

5. Daftar Pustaka

de Abreu, A.P. and Fuchigami, H.Y. (2022) 'An efficiency and robustness analysis of warm-start mathematical models for idle and waiting times optimization in the flow shop', *Journal Computers and Industrial Engineering*, 166. Available at: <https://doi.org/10.1016/j.cie.2022.107976>.

Baker, K.R. and Trietsch, D. (2018) *Principles of sequencing and scheduling*. 2nd edn. Wiley.

Blum, C. (2005) 'Ant colony optimization: Introduction and recent trends', *Journal Physics of Life Reviews*. 2(4), pp 353-373. Available at: <https://doi.org/10.1016/j.plprev.2005.10.001>.

Della Croce, F. and Scatamacchia, R. (2020) 'The longest processing time rule for identical parallel machines revisited', *Journal of Scheduling*, 23(2), pp. 163–176.

Demir, H.I.C. (2020) 'Dynamic integrated process planning, scheduling and due-date assignment using ant colony optimization', *Journal Computer & Industrial Engineering*, 149.

Dorigo, M. and Blum, C. (2005) 'Ant colony optimization theory: a survey', *Journal Theoretical Computer Science*, 344(2–3), pp. 243-278. Available at: <https://doi.org/10.1016/j.tcs.2005.05.020>.

Febianti, E., Muharni, Y. and Azhary, M.N. (2020) 'Penjadwalan produk menggunakan teknik enumerasi', *Journal of Integrated System*, 3(2), pp. 180-193. Available at: <https://doi.org/10.28932/jis.v3i2.2900>.

Fera, M. *et al.* (2013) 'Production scheduling approaches for operations management', in M.M. Schiraldi (ed.) *Operations management*. InTech. Available at: <https://doi.org/10.5772/55431>.

Liliani and Alfian, A. (2014) 'Usulan penjadwalan produksi dengan algoritma Ant Colony (studi kasus PT Shima Prima Utama Palembang)', in *Simposium Nasional RAPI XIII - FT UMS*. Available at: <chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://publikasiilmiah.ums.ac.id/xmlui/bitstream/handle/11617/5517/2.Liliani.pdf?sequence=1&isAllowed=y>.

Lopez, P. and Roubellat, F. (2010) *Production scheduling*. Wiley.

Luo, Y. *et al.* (2020) 'Charging scheduling strategy for different electric vehicles with

optimization for convenience of drivers, performance of transport system and distribution network', *Journal Energy*, 194. Available at: <https://doi.org/10.1016/j.energy.2019.116807>.

Muharni, Y. *et al.* (2020) 'The application of LPT, CEGA, and PSO method on flow shop scheduling with parallel machine', in *IOP Conference Series: Materials Science and Engineering*. Available at: <https://doi.org/10.1088/1757-899X/909/1/012054>.

Muharni, Y., Febianti, E. and Sofa, N.N. (2019) 'Minimasi makespan pada penjadwalan flow shop mesin paralel produk steel Bridge B-60 menggunakan metode Longest Processing Time dan Particle Swarm Optimization', *Journal Industrial Servicess*, 4(2). Available at: <https://doi.org/10.36055/jiss.v4i2.5154>.

Muharni, Y., Saeful M, A.I. and Rubyanti, T.E. (2020) 'Penjadwalan flow shop mesin paralel menggunakan metode Longest Processing Time dan Cross Entropy-Genetic Algorithm pada pembuatan produk steel Bridge B-60', *Jurnal Ilmiah Teknik Industri*, 7(3). Available at: <https://doi.org/10.24912/jitiuntar.v7i3.6338>.

Panggabean, J.F.R. (2018) 'Hybrid ant colony optimization-genetics algorithm to minimize makespan flow shop scheduling', *International Journal of Engineering and Technology (UAE)*, 7(2.2). Available at: <https://doi.org/10.14419/ijet.v7i2.2.11868>.

Panjaitan, A.H. and Ginting, R. (2019) 'Usulan penjadwalan produksi dengan algoritma Ant Colony (studi kasus PT. KLM Medan)', in *Talenta Conference Series: Energy and Engineering (EE)*. Available at: <https://doi.org/10.32734/ee.v2i3.760>.

Pinedo, M.L. (2016) *Scheduling: theory, algorithms, and systems*. 4th edn. New York: Springer.

Rui, Z. *et al.* (2014) 'An ant colony algorithm for job shop scheduling problem with tool flow', in *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 228(8). Available at: <https://doi.org/10.1177/0954405413514398>.

Sivakumar, P. *et al.* (2015) 'Real-time task scheduling for distributed embedded system using MATLAB toolboxes', *Indian Journal of Science and Technology*, 8(15), pp. 1-7. Available at: <https://doi.org/10.17485/ijst/2015/v8i15/55680>.

Toksari, M.D. (2016) 'A hybrid algorithm of Ant Colony Optimization (ACO) and Iterated Local Search (ILS) for estimating electricity domestic consumption: case of Turkey', *International Journal of Electrical Power and Energy Systems*, 78, pp. 776-782.