

BAB 4

PENGHITUNGAN, HASIL AKHIR DAN ANALISA

Bab ini berisi hasil dari *run* program yang telah dibuat kemudian dilakukan analisa terhadap hasil yang diperoleh.

4.1. Penyusunan Algoritma

Data penelitian ini diolah dengan menggunakan bahasa pemrograman VBA pada Microsoft Excel. Bahasa pemrograman ini dipilih karena penggunaannya yang tidak memerlukan pemasangan *software* baru melainkan sudah ada di *microsoft*. Disamping itu ada kemudahan-kemudahan lainnya bila memerlukan pengolahan lebih lanjut. Microsoft Visual Basic for Applications (VBA) adalah sebuah turunan bahasa pemrograman *Visual Basic* yang dikembangkan oleh *Microsoft* dan dirilis pada tahun 1993. VBA adalah hasil kombinasi yang terintegrasi antara lingkungan pemrograman (*Visual Basic Editor*) dengan bahasa pemrograman (*Visual Basic*) yang memudahkan *user* untuk mendesain dan membangun program *Visual Basic* dalam aplikasi utama Microsoft Office, serta aplikasi-aplikasi tertentu. VBA didesain untuk melakukan beberapa tugas, seperti halnya membuat alternatif spesifikasi sebuah aplikasi seperti *Microsoft Office* atau *Microsoft Visual Studio*. Kegunaan VBA adalah mengotomatisasi pekerjaan-pekerjaan kompleks yang dilakukan secara berulang-ulang²³

4.1.1. Langkah-Langkah Penyusunan Algoritma

Dalam penelitian ini, Algoritma DE diterapkan dalam VBA Excel dan mengevaluasi kinerja mengenai penerapan umum DE sebagai teknik optimasi numerik. Sebagai parameter yang sesuai algoritma DE sangat penting untuk hasil yang akurat, sehingga parameter-parameter ini harus dipilih dengan hati-hati. Rincian pengaturan parameter ini akan dibahas dalam hasil analisa parameter DE melalui analisa DOE (*design of experiment*) dengan bantuan *minitab 14.0*, yang rinciannya akan dibahas dalam bab ini juga. Parameter kontrol yang adalah faktor yang memengaruhi konvergensi algoritma DE, meliputi ukuran populasi (NP),

²³ http://id.wikipedia.org/wiki/Visual_Basic_for_Applications halaman ini terakhir diubah pada 03:03, 10 Maret 2010. diakses 15 Juni 2010

faktor mutasi (F) dan probabilitas *crossover* (CR), sehingga pemilihan parameter tersebut menjadi penting untuk memperoleh hasil yang dapat diandalkan. Program VBA yang telah dibuat untuk menyelesaikan permasalahan *job shop* dapat dilihat pada lampiran 1, 2, dan 3. Penentuan parameter-parameter kontrol untuk optimasi DE dapat dijelaskan sebagai berikut:

- Inisialisasi populasi awal

- Melakukan penyetelan pada generasi ke-0.

Penentuan parameter-parameter yang akan digunakan menjadi input bagi algoritma DE adalah dengan mempertimbangkan nilai yang terbaik dari dua skenario DOE yang telah dilakukan, di samping dengan memperhatikan anjuran dari berbagai peneliti pada berbagai jurnal, yaitu: $NP = 22$; $F = 0,6$; $CR = 0,5$; dan jumlah iterasi = 5000.

- Menentukan populasi awal

Ukuran populasi dari algoritma DE sebaiknya sedang. Jika ukuran populasi terlalu kecil, DE cenderung konvergen pada optima lokal karena kurang luasnya area pencarian; sedangkan bila terlalu besar DE memerlukan fungsi evaluasi dalam jumlah yang besar agar konvergen; dengan begitu akan memerlukan waktu komputasi yang lama.

- Menentukan vektor permutasi

Permutasi adalah penyusunan kembali suatu kumpulan objek dalam urutan yang berbeda dari urutan yang semula²². Dalam hubungan ini, hasil bilangan acak yang disesuaikan dengan urutan jumlah *job* dan jumlah operasi, dibuat urutannya dimulai dari yang terkecil dengan memperhatikan posisi sebelumnya. Vektor pada tiap individu bilangan acak ini nantinya akan menjadi nomor urutan pengerjaan 85 pesanan sesuai data penelitian. Sebagai contoh, jika ada empat *job* dengan empat dimensi bilangan acak, yaitu 0,0712; 0,2454; 0,3105 dan 0,3456; maka masing-masing diberi dimensi 1, dimensi 2, dimensi 3, dan dimensi 4, dimana urutan proses pengerjaan *job* dimulai dari nilai yang terkecil dilanjutkan dengan yang lebih besar sampai yang paling besar.

²² <http://id.wikipedia.org/wiki/Permutasi>, diubah pada 10:01, 30 Juni 2010. diakses pada 8:10, 25 Juni 2010

- Mengevaluasi setiap individu

Setelah mendapatkan vektor-vektor urutan *job* setiap individu, kemudian dibuat fungsi objektifnya (seperti total biaya produksi). Evaluasi dilakukan dengan cara menghubungkan vektor tersebut dengan data waktu operasi, sehingga fungsi objektif dapat dihitung. Evaluasi ini dimaksudkan untuk mengetahui individu pada generasi (iterasi) awal yang memiliki total biaya produksi terkecil yang selanjutnya akan diturunkan pada generasi selanjutnya.

- Memperbaharui generasi (iterasi)

Populasi individu pada iterasi awal akan berevolusi membentuk populasi individu iterasi baru. Individu-individu mengalami evolusi melalui serangkaian proses, yang dimulai dengan proses mutasi, proses pindah silang, dan proses penyeleksian. Jika generasi awal disimbolkan sebagai $t = 0$, maka iterasi baru disimbolkan sebagai $t = t + 1$.

- Proses Mutasi

Faktor mutasi seharusnya tidak kurang dari nilai tertentu untuk mencegah konvergensi prematur (solusi yang tidak optimal). Nilai faktor mutasi yang lebih besar memberikan kemungkinan keluar dari solusi lokal minimum. Namun, jika nilai faktor mutasi lebih dari 1^{23} , kecepatan konvergensi menurun. Pemilihan nilai faktor mutasi yang tepat adalah hal yang sulit, dan karena itu harus dipilih secara hati-hati dengan menggunakan pengalaman, atau *best practice* dari percobaan yang telah dilakukan.

- Proses *Crossover*

Crossover memengaruhi jumlah variabel yang akan diubah dalam vektor ($U_i(G)$) dibandingkan dengan target vektor ($X_i(G)$). Jika nilai *crossover* besar, variabel lebih banyak diambil dari vektor mutan ($V_i(G)$) daripada vektor target ($X_i(G)$). Probabilitas *crossover* besar sering mempercepat konvergensi, tapi populasi mungkin saling bertemu sebelum waktunya. Di sisi lain, lebih banyak variabel

²³ R. Gamperle, S. D. Muller and P. Koumoutsakos, "A parameter study for differential evolution," *Advances in Intelligent Systems, Fuzzy Systems, Evolution Computation*, WSEAS Press, pp. 293-298, 2002.

diambil daripada vektor target ($X_i (G)$) dari vektor mutan ($V_i (G)$) apabila nilai probabilitas *crossover* kecil. Apabila probabilitas *crossover* sama dengan 0, maka semua vektor *trial* ($U_i (G)$) tetap menjadi sama dengan anggota di vektor target ($X_i (G)$) dan tidak ada perbaikan dalam hasil pencarian. Jika probabilitas *crossover* sama dengan 1, maka semua variabel dalam vector *trial* ($U_i (G)$) diambil dari vektor mutan ($V_i (G)$). Hal ini berarti tidak ada perpindahan komponen antara vektor mutan ($V_i (G)$) dan vektor target ($X_i (G)$), sehingga menurunkan keragaman populasi. Oleh karena itu pengguna harus memilih nilai yang tepat dari probabilitas *crossover* dengan hati-hati antara 0 dan 1.

- Proses Seleksi

Seleksi merupakan mekanisme untuk menentukan apakah vektor *trial* menggantikan vektor target atau tidak. Jika nilai fungsi objektif vektor *trial* lebih kecil atau sama dengan vektor target, maka vektor *trial* akan menjadi populasi generasi berikutnya. Sedangkan bila tidak maka vektor target tetap menjadi populasi generasi berikutnya.

- Kriteria Terminasi DE

Karena perilaku stokastiknya, tidak mungkin algoritma DE menjamin solusi optimal telah tercapai setelah beberapa iterasi, dalam prakteknya kondisi terminasi yang dapat digunakan dalam algoritma DE, seperti; jumlah maksimum iterasi tertentu, berhenti ketika tidak ada *improvement* terhadap solusi terbaik atau berhenti jika keragaman seluruh populasi menjadi terlalu rendah.

4.1.2. Verifikasi dan Validasi Program

Verifikasi program dilakukan untuk mengetahui apakah program sudah sesuai dengan konsep model yang dibangun sebelumnya, yaitu dengan memasukkan input data acak ke dalam program dan program berjalan sesuai konsepnya. Hasil dari output program akan keluar berupa total biaya produksi yang merupakan jumlah biaya produksi keseluruhan ditambah dengan biaya penalti akibat keterlambatan penyelesaian produk. Dari hasil verifikasi terhadap 10 iterasi yang dilakukan terdapat program dengan data yang sama dengan data yang akan

menjadi penelitian, hal ini menunjukkan terjadi perubahan dari solusi awal sampai didapat solusi akhir, yaitu sebagai berikut pada tabel 4.1.;

Tabel 4.1. Hasil 10 Iterasi untuk Verifikasi Program

Solusi Awal	528,975.20
<i>Makespan</i> Awal	1154.15
Solusi Akhir	510,971.50
<i>Makespan</i> Akhir	1064.35

Dari tabel terlihat total biaya solusi awal adalah sebesar 528,975.2 dengan *makespan* 1154.15 dan solusi akhir sebesar 510,971.5 dengan *makespan* 1064.35. Karena terjadi perubahan terhadap *outputnya* maka program telah terverifikasi.

Validasi program dilakukan dengan memeriksa apakah hasil program sudah sesuai dengan yang seharusnya, yaitu dengan memeriksa *presedence constraint* dari setiap *job*. Data yang digunakan hasilnya adalah sebagai berikut pada tabel 4.2.; dimana jumlah populasi NP adalah 5, nilai F adalah 0.5 dan nilai CR adalah 0.3, dengan jumlah iterasi maksimum adalah 10.

Tabel 4.2. Parameter yang Digunakan dalam Validasi

No	Data	Nilai
<u>Data Problem</u>		
1	Jumlah Job	5
2	Jumlah Mesin	8
<u>Data Algoritma</u>		
3	Jumlah Populasi	5
4	Faktor Mutasi	0.5
5	Tingkat Crossover	0.3
6	Jumlah Local search	0
<u>Data Terminasi</u>		
7	Jumlah Iterasi Maksimum	10

Selanjutnya hasil validasi terhadap program sebagai berikut:

- Inisialisasi populasi individual sesuai dengan solusi acak.

Populasi awal berisi bilangan acak antara 0 dan 1 sesuai dengan rumus yang telah dibuat pada program, yaitu solusi bilangan acak. Populasi target ditentukan sesuai dengan jumlah setiap proses dari *job*.

Tabel 4.3. Solusi Awal (Hasil *Run* Program)

Nomor Job	Mesin Ke		
1	0.46616	0.83558	0.92111
2	0.52065	0.84109	0.46004
3	0.04991	0.37765	0.78988
4	0.30434	0.69499	0.15849
5	0.54916	0.97108	0.11103

- Melakukan pengurutan (permutasi) pada setiap individu dari populasi awal. Dilakukan dengan cara mencari nilai bilangan acak yang terkecil kemudian periksa pada *job* nomor berapa. Selanjutnya tempatkan pada baris atau *job* yang sesuai pada kolom pertama atau pada mesin pertama, begitu selanjutnya sampai semua bilangan menempati posisinya masing-masing. Jadi pada waktu menugaskan suatu *job* pada mesin tertentu, bisa dilihat langsung pada tabel 4.4. permutasi populasi awal, dimana urutan operasinya pada mesin tersebut dimulai dari yang nilainya terkecil. Sebagai contoh, pada baris pertama, urutan bilangan acak dari yang terkecil hingga terbesar adalah 0,46616; 0,83558; dan 0,92111, demikian pula baris ke 2, sampai ke 5, masing-masing diurutkan, sehingga didapatlah Tabel 4.4. yang menyatakan permutasi (urutan pengerjaan *job*) populasi awal untuk setiap mesin.

Tabel 4.4. Permutasi Populasi Awal

Nomor Job	Mesin Ke		
1	0.46616	0.83558	0.92111
2	0.46004	0.52065	0.84109
3	0.04991	0.37765	0.78988
4	0.15849	0.30434	0.69499
5	0.11103	0.54916	0.97108

Kemudian dari tabel 4.4. setiap kolomnya diurutkan lagi mulai dari yang terkecil sampai yang terbesar; sebagai contoh kolom pertama 0.04991, 0.11103, 0.15849, 0.46004, dan 0.46616. Jadi urutan pengerjaan *job* pada mesin 1 adalah nomor *job* 3, 5, 4, 2 dan 1, demikian seterusnya untuk kolom berikutnya. Setelah melakukan pengurutan operasi setiap *job* terhadap populasi awal, untuk setiap individu dicari fungsi objektifnya, yaitu nilai total

biaya produksinya ditambah dengan biaya penalti bila terjadi keterlambatan pengerjaan. Tahap-tahap penyelesaiannya adalah sebagai berikut:

- Mengeset waktu mula-mula sama dengan nol.
- Memasukkan waktu proses untuk operasi yang mendapatkan jadwal pertama kali dilanjutkan dengan operasi *job* berikutnya sesuai dengan hasil permutasi populasi awal, yaitu untuk operasi berikutnya ditambahkan waktu proses di setiap proses yang bersangkutan dengan waktu penyelesaian pendahulunya (*predecessor*).
- Setelah mendapatkan total waktu penyelesaian masing-masing *job*, kemudian dibandingkan dengan *due date* dari *job* yang bersangkutan, untuk memperoleh nilai keterlambatan. Apabila nilai keterlambatan bernilai negatif (*earliness*), nilai tersebut akan berubah menjadi nol.
- Selanjutnya, keterlambatan setiap *job* dikalikan dengan penalti (bobot) masing-masing sehingga didapatkan biaya keterlambatan untuk setiap *job*. Lalu biaya keterlambatan ditambahkan dengan biaya produksi seluruh *job* untuk mendapatkan total biaya produksi.
- Setelah mendapatkan solusi awal, kemudian dilanjutkan dengan melakukan proses iterasi DE, dimana bilangan acak dari solusi awal digunakan sebagai data untuk proses mutasi, *crossover* dan seleksi mengikuti kaidah algoritma DE. Hasil akhirnya dapat dilihat pada tabel 4.5 di bawah ini yang menunjukkan proses urutan pengerjaan *job* yang akan dikerjakan pada mesin-mesin 1, 2 dan 3.

Tabel 4.5. Perhitungan Waktu Produksi setiap *Job* Solusi Akhir

Nomor Mesin	Mesin Ke	Item					
		Job number	3	5	4	2	1
1	1	Start Time	0	6.2	12.4	18.6	24.8
		End Time	6.2	12.4	18.6	24.8	31
		Job number	4	3	2	5	1
2	1	Start Time	18.6	22.7	26.8	30.9	35
		End Time	22.7	26.8	30.9	35	39.1
		Job number	4	3	2	1	5
3	1	Start Time	22.7	27.2	31.7	39.1	43.6
		End Time	27.2	31.7	36.2	43.6	48.1
		Job number	4	3	2	1	5

Perhitungan waktu produksi masing-masing *job* dapat dilihat pada Tabel 4.5 dengan penjelasan sebagai berikut:

- Dimulai dari mesin 1 (karena semua proses harus melalui mesin 1) untuk pertama kali, dipilih *job* no 3 dengan waktu proses 6.2 jam, diikuti oleh *job* no 5 dengan waktu 6.2 jam, lalu diikuti *job* 4, 2 dan 1 total waktu $0+6.2+6.2+6.2+6.2+6.2 = 31$ jam.
- Kemudian mesin no 2, urutan *job* 4-3-2-5-1, di mana masing-masing operasi pada mesin ini menghabiskan waktu 4.1 jam. Karena *job* 4 baru bisa dimulai setelah operasi pada mesin 1, yaitu jam 18.6, maka penghitungan total waktu adalah $18.6+4.1+4.1+4.1+4.1+4.1 = 39.1$; hasil ini sesuai dengan hasil program.
- Selanjutnya mesin no 3 dengan urutan *job* 4-3-2-1-5, dimana masing-masing waktunya adalah 4.5 jam. Karena *job* 4 baru bisa dimulai setelah jam ke-22.7, maka waktu penyelesaiannya adalah $22.7+4.5 +4.5 +4.5 = 36.2$. Karena mesin 1 baru selesai pada jam ke-39.1, maka total waktu penyelesaian adalah $39.1+4.5+4.5 = 48.1$; hasilnya sesuai dengan hasil program.

Tabel 4.6. Perhitungan Total Biaya Produksi Solusi Akhir

Nomor Job	Biaya Produksi	DueDate	EndTime	Jumlah Waktu Keterlambatan	Biaya Penalty persatuan waktu	Biaya Penalty	Total Biaya
1	1680.25	92.04	43.6	0	1	0	1680.25
2	1680.25	92.04	36.2	0	1	0	1680.25
3	1680.25	92.04	31.7	0	1	0	1680.25
4	1680.25	92.04	27.2	0	1	0	1680.25
5	1680.25	230.1	48.1	0	1	0	1680.25

Kemudian dilakukan penghitungan total biaya produksi, yaitu dengan menjumlahkan semua biaya penalti dan biaya produksi. Waktu akhir penyelesaian masing-masing *job* diperoleh dari tabel 4.5, yang digunakan untuk menghitung jumlah waktu keterlambatan. Tabel 4.6 menunjukkan biaya produksi, *due date*, waktu selesai, jumlah waktu keterlambatan, biaya penalti dan total biaya produksi masing-masing *job*. Setelah memeriksa perhitungan waktu produksi setiap *job* dan total biaya produksi pada solusi akhir, ternyata sesuai dengan konsep modelnya, maka program telah tervalidasi.

4.2. Input

4.2.1. Input Data

Data yang telah dikumpulkan kemudian diolah dan dimasukkan pada program yang telah dibuat pada Microsoft Excel dengan menggunakan VBA Excel. Data-data pada Tabel 3.6 dan Tabel 3.7, yaitu data biaya produksi, biaya penalti dan *due date* dimasukkan pada *sheet* "DataBiaya"; data waktu proses dimasukkan ke *sheet* "DataWaktuProses"; data urutan operasi pengerjaan dimasukkan ke *sheet* "DataUrutanProses"; dan data mesin dimasukkan ke *sheet* "DataMesin".

4.2.2. Penentuan Input Parameter Kontrol

Terlebih dahulu dilakukan *Design of Experiments* (DOE) sebelum mendapatkan nilai terbaik untuk menentukan parameter kontrol algoritma DE. Karena perilaku stokastik dari DE, maka dilakukan dua skenario untuk menguji faktor utama yang penting dalam proses DE, masing-masing faktor terdiri dari 3 level, yaitu level 1, 2, dan 3. DOE yang dipakai berjenis *full factorial design*, sehingga ada $(3^4) = 81$ kombinasi yang mungkin untuk dilakukan percobaan. Setiap kombinasi akan direplikasi sebanyak 5 kali, agar diperoleh hasil yang lebih akurat. Jadi terdapat $81 \times 5 = 405$ percobaan.

Program MINITAB 14 digunakan untuk melihat hasil DOE dengan tingkat kepercayaan 95%, sehingga nilai $\alpha = 5\%$. Nilai-nilai pada tabel 4.7 adalah nilai yang digunakan pada skenario 2, setelah dilakukan percobaan sebanyak 405 kali dengan *output* berupa total biaya produksi (hasil percobaan ini dapat dilihat pada lampiran 5).

Tabel 4.7. Skenario 2 untuk DOE

Faktor	Level		
	1	2	3
Ukuran Populasi NP	15	20	25
Jumlah Iterasi JI	15	20	25
Faktor Mutasi F	0.5	0.6	0.7
Tingkat Crossover CR	0.4	0.5	0.6

Percobaan yang telah dilakukan dengan me *run* program sesuai dengan skenario 1 dan 2 (hasil skenario 1 dapat dilihat pada lampiran 4). Diperoleh kesimpulan untuk nilai-nilai terbaik yang akan dipakai pada penjadwalan, dengan fungsi tujuan meminimumkan total biaya produksi, seperti terlihat pada Tabel 4.8 berikut;

Tabel 4.8. Parameter DE berdasarkan Hasil DOE

No	Data	Nilai
Data Problem		
1	Jumlah Job	85
2	Jumlah Mesin	8
Data Algoritma		
3	Jumlah Populasi	22
4	Faktor Mutasi	0.6
5	Tingkat Crossover	0.5
6	Jumlah Local search	0
Data Terminasi		
7	Jumlah Iterasi Maksimum	5000

Parameter kontrol algoritma DE berdasarkan hasil DOE yang akan dipakai adalah: jumlah populasi 22, faktor mutasi 0.6, tingkat *crossover* 0.5 dan jumlah iterasi maksimum 5000. Nilai-nilai tersebut akan dijelaskan pada bagian analisis skenario untuk parameter DE.

4.3. Pengolahan Data dan Hasil

Program VBA Excel ini dijalankan pada komputer HP xw 4600 Workstation Intel (R) Core(TM)2 Quad CPU 9400 @ 2.66GHz, 2.00 GB of RAM. *Script Macro* program untuk perhitungan urutan penjadwalan dapat dilihat pada lampiran 1, lampiran 2 dan lampiran 3.

4.3.1. Hasil Penjadwalan dengan Algoritma DE

Parameter yang digunakan adalah sebagai berikut: ukuran Populasi 22, jumlah Iterasi 5000, faktor mutasi 0.6, dan faktor pindah silang 0,5. Setelah dilakukan *run* sebanyak 5 kali, didapat ringkasan hasil sebagai terlihat pada Tabel 4.9.

Tabel 4.9. Hasil *Run* Program

	Run 1	Run 2	Run 3	Run 4	Run 5
Total Biaya	446418.23	439277.45	446893.11	442606.23	442137.77
<i>Makespan</i>	751.9	649	700.45	702.6	754.7
Jumlah Pesanan yang tidak terlambat	31	34	27	30	28
Waktu komputasi	94	97	95	94	93

Dari Tabel 4.9. diatas dilihat, bahwa untuk hasil solusi terbaik adalah *run 2* dimana total biaya produksi terkecil mencapai 439,277.45 USD dengan total waktu penyelesaian 649 jam, dan waktu komputasi 97 menit. Hasil komputasi penjadwalan untuk mesin 1, urutan *job* yang akan dikerjakan adalah sebagai berikut seperti dilihat di tabel 4.10.

Tabel 4.10. Penjadwalan sebagai Solusi Akhir

Nomor Mesin	Mesin Ke	Item (Urutan <i>Job</i> Ke)												
		Job number	58	85	68	84	83	74	66	64	44	62	75	
1	1	Start Time	0	8.1	13.6	20.6	26.1	31.6	38.6	47.1	55.6	61.1	69.6	
		End Time	8.1	13.6	20.6	26.1	31.6	38.6	47.1	55.6	61.1	69.6	76.6	
		Job number	46	70	82	73	65	49	24	40	71	51	67	
		Start Time	76.6	82.1	89.1	94.6	101.6	110.1	115.6	120.1	125.4	132.4	137.9	
		End Time	82.1	89.1	94.6	101.6	110.1	115.6	120.1	125.4	132.4	137.9	146.4	
		Job number	9	50	28	77	80	47	36	54	76	81	34	
		Start Time	146.4	152.6	158.1	162.6	169.6	175.1	180.6	185.9	190.4	197.4	202.9	
		End Time	152.6	158.1	162.6	169.6	175.1	180.6	185.9	190.4	197.4	202.9	208.2	
		Job number	57	29	3	55	39	53	25	63	41	45	56	
		Start Time	208.2	212.7	218	224.2	228.7	234	239.5	244	252.5	258	263.5	
		End Time	212.7	218	224.2	228.7	234	239.5	244	252.5	258	263.5	268	
		Job number	17	27	8	48	16	72	31	33	10	22	59	
		Start Time	268	272.5	277	283.2	288.7	295.2	302.2	307.5	312.8	319	323.5	
		End Time	272.5	277	283.2	288.7	295.2	302.2	307.5	312.8	319	323.5	331.6	
		Job number	11	18	38	32	61	26	78	14	37	13	30	
Start Time	331.6	337.8	342.3	347.6	352.9	361	365.5	372.5	379	384.3	390.8			
End Time	337.8	342.3	347.6	352.9	361	365.5	372.5	379	384.3	390.8	396.1			
Job number	7	79	6	35	2	1	23	19	69	5	42			
Start Time	396.1	402.3	409.3	415.5	420.8	427	433.2	437.7	442.2	449.2	455.4			
End Time	402.3	409.3	415.5	420.8	427	433.2	437.7	442.2	449.2	455.4	460.9			
Job number	21	4	15	20	52	43	60	12						
Start Time	460.9	465.4	471.6	478.1	482.6	488.1	493.6	501.7						
End Time	465.4	471.6	478.1	482.6	488.1	493.6	501.7	507.9						
2	1	Job number	85	68	83	84	74	85	66	58	75	68	44	
		Start Time	13.6	20.6	31.6	35.1	38.6	42.6	51.6	62.1	76.6	80.6	88.85	
		End Time	17.1	24.6	35.1	38.6	42.6	51.6	62.1	76.3	80.6	88.85	91.15	
		Job number	64	82	62	70	73	49	40	46	65	71	67	
		Start Time	91.15	101.7	105.2	115.7	119.7	123.7	126	130.1	132.4	142.9	146.9	
		End Time	101.7	105.2	115.7	119.7	123.7	126	130.1	132.4	142.9	146.9	157.4	
		Job number	24	74	75	50	80	51	84	47	76	36	28	
		Start Time	157.4	158.9	167.1	175.4	177.7	181.2	183.5	192.5	197.4	201.4	205.5	
		End Time	158.9	167.1	175.4	177.7	181.2	183.5	192.5	194.8	201.4	205.5	207	
		Job number	83	77	57	81	34	29	39	54	55	3	71	
		Start Time	207	216	220	225.5	229	233.1	237.2	241.3	246.8	252.3	256.4	
		End Time	216	220	225.5	229	233.1	237.2	241.3	246.8	252.3	256.4	264.7	
		Job number	25	63	82	45	9	70	56	41	73	31	16	
		Start Time	264.7	266.2	276.7	285.7	288	292.1	300.3	305.8	308.1	316.4	320.5	
		End Time	266.2	276.7	285.7	288	292.1	300.3	305.8	308.1	316.4	320.5	331	

Tabel 4.10. (sambungan)

Nomor Mesin	Mesin Ke	Item (Urutan Job Ke)											
		Job number	33	53	18	77	10	72	22	61	48	17	27
		Start Time	331	335.1	342.3	343.8	352.1	356.2	360.2	361.7	375.9	378.2	379.7
		End Time	335.1	337.4	343.8	352.1	356.2	360.2	361.7	375.9	378.2	379.7	381.2
		Job number	26	38	81	76	8	11	32	37	78	7	59
		Start Time	381.2	382.7	386.8	395.8	404	408.1	412.2	416.3	420.4	424.4	428.5
		End Time	382.7	386.8	395.8	404	408.1	412.2	416.3	420.4	424.4	428.5	442.7
		Job number	23	80	42	4	21	5	69	6	52	2	35
		Start Time	442.7	444.2	460.9	471.6	475.7	477.2	481.3	485.3	489.4	491.7	495.8
		End Time	444.2	453.2	463.2	475.7	477.2	481.3	485.3	489.4	491.7	495.8	499.9
		Job number	79	43	60	12	20	15	30	72	13	14	1
		Start Time	499.9	503.9	506.2	520.4	524.5	526	536.5	540.6	548.9	559.4	569.9
		End Time	503.9	506.2	520.4	524.5	526	536.5	540.6	548.9	559.4	569.9	574
		Job number	69	78	79	19							
		Start Time	574	582.2	590.5	598.7							
		End Time	582.2	590.5	598.7	600.2							
3	1	Job number	85	84	68	74	85	75	58	68	83	66	62
		Start Time	17.1	38.6	42.6	47.1	51.6	80.6	85.1	91.4	101.9	105.9	118.2
		End Time	21.1	42.6	47.1	51.6	79.6	85.1	91.4	101.9	105.9	118.2	130.4
		Job number	64	82	70	71	73	67	74	75	84	65	77
		Start Time	130.4	142.7	146.7	151.2	155.7	160.2	172.4	182.9	193.4	221.4	233.7
		End Time	142.7	146.7	151.2	155.7	160.2	172.4	182.9	193.4	221.4	233.7	238.2
		Job number	83	81	57	71	76	54	63	55	70	82	56
		Start Time	238.2	266.2	270.2	273.7	284.2	288.7	292.2	304.4	307.9	318.4	346.4
		End Time	266.2	270.2	273.7	284.2	288.7	292.2	304.4	307.9	318.4	346.4	349.9
		Job number	3	73	16	9	80	61	72	77	76	81	10
		Start Time	349.9	354.4	364.9	382.4	386.9	390.9	397.2	401.7	412.2	422.7	450.7
		End Time	354.4	364.9	382.4	386.9	390.9	397.2	401.7	412.2	422.7	450.7	455.2
		Job number	80	59	7	69	6	78	4	5	79	12	60
		Start Time	455.2	483.2	489.5	494	498.5	503	507.5	512	516.5	524.5	529
		End Time	483.2	489.5	494	498.5	503	507.5	512	516.5	521	529	535.3
		Job number	15	72	13	2	11	69	14	78	1	79	8
		Start Time	536.5	554	564.5	582	586.5	591	601.5	619	629.5	634	644.5
		End Time	554	564.5	582	586.5	591	601.5	619	629.5	634	644.5	649
4	1	Job number	85	68	74	75	84	83	71	70	82	73	77
		Start Time	21.1	47.1	77.1	107.1	137.1	157.1	177.1	207.1	237.1	257.1	287.1
		End Time	41.1	77.1	107.1	137.1	157.1	177.1	207.1	237.1	257.1	287.1	317.1
		Job number	81	76	80	72	69	78	79				
		Start Time	317.1	337.1	390.9	410.9	498.5	528.5	558.5				
		End Time	337.1	367.1	410.9	440.9	528.5	558.5	588.5				
5	1	Job number											
		Start Time											
		End Time											
6	1	Job number											
		Start Time											
		End Time											
7	1	Job number	49	44	40	62	58	24	46	28	36	66	50
		Start Time	126	132.1	138.2	154.4	158.9	162.1	168.6	207	213.5	229.7	234.2
		End Time	132.1	138.2	154.4	158.9	162.1	168.6	174.7	213.5	229.7	234.2	240.3

Tabel 4.10. (sambungan)

Nomor Mesin	Mesin Ke	Item (Urutan <i>Job</i> Ke)											
		Job number	34	47	29	64	65	45	39	25	63	57	54
		Start Time	240.3	256.5	262.6	278.8	283.3	288	294.1	310.3	316.8	321.3	323.8
		End Time	256.5	262.6	278.8	283.3	287.8	294.1	310.3	316.8	321.3	323.8	326.3
		Job number	67	55	18	41	33	56	48	53	22	61	51
		Start Time	326.3	330.8	343.8	350.3	356.4	372.6	378.2	384.3	390.4	397.2	400.4
		End Time	330.8	333.3	350.3	356.4	372.6	375.1	384.3	390.4	396.9	400.4	406.5
		Job number	38	17	31	37	27	23	59	42	52	43	32
		Start Time	406.5	422.7	429.2	445.4	461.6	468.1	489.5	492.7	498.8	506.2	512.3
		End Time	422.7	429.2	445.4	461.6	468.1	474.6	492.7	498.8	504.9	512.3	528.5
		Job number	60	20	26	35	21	30	19				
		Start Time	535.3	538.5	545	551.5	567.7	574.2	600.2				
		End Time	538.5	545	551.5	567.7	574.2	590.4	606.7				
8	1	Job number	68	85	74	75	84	71	83	70	73	82	77
		Start Time	101.9	103.4	182.9	193.4	221.4	284.2	285.7	328	364.9	366.4	412.2
		End Time	103.4	145.7	184.4	194.9	263.7	285.7	328	329.5	366.4	408.7	413.7
		Job number	76	81	80	72	69	78	79				
		Start Time	422.7	450.7	493	564.5	601.5	629.5	644.5				
		End Time	424.2	493	535.3	566	603	631	646				

Mesin 5 dan 6 kosong karena memang mesin tersebut adalah *dummy* dari mesin 2 dan 3, jadi bila ada pekerjaan yang masuk ke mesin 5 maka akan dikerjakan di mesin 2 sepanjang tidak melanggar *presedence constraint* dari *job* tersebut, begitu juga *job* yang akan dikerjakan pada mesin 6 akan dikerjakan di mesin 3.

4.3.2. Langkah-Langkah Logis Penjadwalan

Penjelasan langkah-langkah logis dari hasil penelitian terhadap 85 pesanan dari 10 macam *job*, dilakukan dengan mengambil beberapa contoh, yaitu terutama untuk *job* jenis 9 pada tabel 4.10, sebagai berikut;

Contoh diambil adalah pesanan yang ke 68, pesanan ini mewakili semua pesanan jenis *job* 9 dan 10. Pesanan ke 68 (warna hijau muda dan oranye - jenis *job* 9), pertama kali dikerjakan pada mesin 1 jam ke 13.6 selesai pada jam ke 20.6; kemudian dikerjakan pada mesin 2 jam ke 20.6, selesai jam ke 24.6; kemudian dikerjakan pada mesin 3 pada jam ke 42.6 dan selesai pada jam ke 47.1; kemudian dikerjakan pada mesin 4 pada jam ke 47.1 dan selesai jam ke 77.1, kemudian balik kembali ke mesin 2 pada jam ke 80.6 dan selesai pada jam ke 88.85, dilanjutkan pada mesin 3 pada jam ke 91.4 dan selesai pada jam ke 101.9, terakhir

dikerjakan pada mesin 8 pada jam ke 101.9 dan selesai pada jam ke 103.4. Tabel 4.11 berikut ini menunjukkan ringkasan operasi *job* ke 68.

Tabel 4.11. Urutan Operasi *Job* ke - 68

No Job 68	Mesin Ke						
	1	2	3	4	5=2	6=3	8
Waktu Mulai	13,6	20,6	42,6	47,1	80,6	91,4	101,9
Waktu Selesai	20,6	24,6	47,1	77,1	88,85	101,9	103,4

Dari tabel bisa dilihat bahwa tidak ada proses yang bersamaan dikerjakan terutama pada mesin 5 dan 6 yang merupakan *dummy* dari mesin 2 dan 3.

Selanjutnya Tabel 4.12. berikut ini merupakan ringkasan dari biaya produksi, *job*, *due date*, *end time* (data diambil dari tabel 4.10), jumlah waktu keterlambatan, dan biaya penalti masing-masing *job*, setelah semua dijumlahkan didapatkan total biaya produksi untuk keseluruhan *job*. Total biaya produksi diperoleh sebesar 439,277.45 USD.

Tabel 4.12. Total Biaya Produksi

Nomor Job	Biaya Produksi	DueDate	EndTime	Jumlah Waktu Keterlambatan	Biaya Penalti persatuan waktu	Biaya Penalti	Total Biaya
1	1680.25	92.04	634	541.96	1	542.0	2222.2
2	1680.25	92.04	586.5	494.46	1	494.5	2174.7
3	1680.25	92.04	354.4	262.36	1	262.4	1942.6
4	1680.25	92.04	512	419.96	1	420.0	2100.2
5	1680.25	230.1	516.5	286.4	1	286.4	1966.7
6	1680.25	230.1	503	272.9	1	272.9	1953.2
7	1680.25	230.1	494	263.9	1	263.9	1944.2
8	1680.25	230.1	649	418.9	1	418.9	2099.2
9	1680.25	460.2	386.9	0	1	0	1680.3
10	1680.25	460.2	455.2	0	1	0	1680.3
11	1680.25	460.2	591	130.8	1	130.8	1811.1
12	1680.25	460.2	529	68.8	1	68.8	1749.1
13	3501.83	460.2	582	121.8	2.08	253.3	3755.2
14	3501.83	460.2	619	158.8	2.08	330.3	3832.1
15	3501.83	460.2	554	93.8	2.08	195.1	3696.9
16	3501.83	460.2	382.4	0	2.08	0	3501.8
17	2389.41	199.29	429.22	229.93	1.42	326.5	2715.9
18	2389.41	199.29	350.3	151.01	1.42	214.4	2603.8
19	2389.41	536.9	606.7	69.8	1.42	99.1	2488.5
20	2389.41	536.9	545	8.1	1.42	11.5	2400.9
21	2389.41	115.05	574.2	459.15	1.42	652.0	3041.4

Tabel 4.12. (sambungan)

Nomor Job	Biaya Produksi	DueDate	EndTime	Jumlah Waktu Keterlambatan	Biaya Penalti persatuan waktu	Biaya Penalti	Total Biaya
22	2389.41	115.05	396.89	281.84	1.42	400.2	2789.6
23	2389.41	115.05	474.62	359.57	1.42	510.6	2900.0
24	2389.41	115.05	168.59	53.54	1.42	76.0	2465.4
25	2389.41	345	316.77	0	1.42	0	2389.4
26	2389.41	345	551.5	206.5	1.42	293.2	2682.6
27	2389.41	345	468.12	123.12	1.42	174.8	2564.2
28	2389.41	345	213.5	0	1.42	0	2389.4
29	3031.5	536.9	278.84	0	1.8	0	3031.5
30	3031.5	536.9	590.4	53.5	1.8	96.3	3127.8
31	3031.5	536.9	445.42	0	1.8	0	3031.5
32	3031.5	536.9	528.52	0	1.8	0	3031.5
33	3031.5	92.04	372.62	280.58	1.8	505.0	3536.5
34	3031.5	92.04	256.52	164.48	1.8	296.1	3327.6
35	3031.5	92.04	567.7	475.66	1.8	856.2	3887.7
36	3031.5	92.04	229.7	137.66	1.8	247.8	3279.3
37	3031.5	230.1	461.62	231.52	1.8	416.7	3448.2
38	3031.5	230.1	422.72	192.62	1.8	346.7	3378.2
39	3031.5	230.1	310.27	80.17	1.8	144.3	3175.8
40	3031.5	230.1	154.39	0	1.8	0	3031.5
41	4075	460.2	356.42	0	2.43	0	4075.0
42	4074.65	115.05	498.82	383.77	2.43	932.6	5007.2
43	4074.65	115.05	512.32	397.27	2.43	965.4	5040.0
44	4074.65	115.05	138.19	23.14	2.43	56.2	4130.9
45	4074.65	115.05	294.07	179.02	2.43	435.0	4509.7
46	4074.65	268	174.71	0	2.43	0	4074.7
47	4074.65	268	262.64	0	2.43	0	4074.7
48	4074.65	268	384.27	116.27	2.43	282.5	4357.2
49	4074.65	268	132.07	0	2.43	0	4074.7
50	4074.65	460.2	240.32	0	2.43	0	4074.7
51	4074.65	460.2	406.52	0	2.43	0	4074.7
52	4074.65	460.2	504.94	44.74	2.43	108.7	4183.4
53	4074.65	460.2	390.39	0	2.43	0	4074.7
54	4206	345	326.27	0	2.5	0	4206.0
55	4206	345	333.27	0	2.5	0	4206.0
56	4206	536.9	375.12	0	2.5	0	4206.0
57	4206	536.9	323.77	0	2.5	0	4206.0
58	4511.375	199.29	162.09	0	2.68	0	4511.4
59	4511.375	199.29	492.7	293.41	2.68	786.3	5297.7
60	4511.375	536.9	538.5	1.6	2.68	4.3	4515.7
61	4511.375	536.9	400.4	0	2.68	0	4511.4
62	7517.69	199.29	158.89	0	4.47	0	7517.7
63	7517.69	199.29	321.27	121.98	4.47	545.3	8062.9
64	7517.69	345	283.34	0	4.47	0	7517.7
65	7517.69	345	287.84	0	4.47	0	7517.7
66	7517.69	536.9	234.2	0	4.47	0	7517.7
67	7517.69	536.9	330.77	0	4.47	0	7517.7

