

Penjadwalan *Job Dua Stages* dan Penentuan Perakas Potong pada *Flexible Manufacturing System* Menggunakan Metode *Genetic Algorithm*

Teguh E.N. Sitepu^{*1}, Ari Setiawan^{#2}, Antonius Candra K.^{#3}

^{*}Departemen Logistik, Institut Teknologi Harapan Bangsa

¹teguh_sitepu@ithb.ac.id

[#]Departemen Teknik Industri, Institut Teknologi Harapan Bangsa

Jl. Dipatiukur No. 80-84., Bandung., Indonesia

²ari_setiawan@ithb.ac.id

³antonius.chandra13@gmail.com

Abstract— This paper deals with two stages job scheduling and cutting tools determination in an aircraft manufacturing industry that implements Flexible Manufacturing System (FMS). System has four parallel CNC machines with 15 jobs. A job consists one to two stages, and every stage consists several operations. Job scheduling solved using Genetic Algorithm (GA) method. Program was built in C# that has objective function to minimize mean flow time and makespan. Job scheduling solved separately between stage-1 and stage-2 then combined manually into a scheduling gantt-chart. Combination method that only merge job scheduling result of stage-2 into stage-1 produce idle time so an adjustment procedure required for eliminate it. Job scheduling result based on mean flow time minimization criteria is 737,3 minutes and makespan criteria is 1.280 minutes. Mean flow time GA is 6,7 minutes faster than SPT. While makespan GA is 50 minutes faster than SPT. Cutting tool required in machine 1 is 1.260 units, machine 2 is 1.250 units, machine 3 is 1.250 units, and machine 4 is 1.280 units.

Keywords— Flexible Manufacturing System (FMS), genetic algorithm, job scheduling, mean flow time, makespan

Abstrak— Penelitian ini berkaitan dengan penjadwalan *job dua stage* dan penentuan perakas potong pada suatu industri manufaktur pesawat terbang yang menerapkan *Flexible Manufacturing System* (FMS). Sistem terdiri atas 4 mesin CNC paralel dengan 15 *job*. Setiap *job* terdiri atas satu sampai dua *stage*, dan setiap *stage* terdiri atas beberapa operasi. Perancangan penjadwalan dilakukan dengan metode *Genetic Algorithm* (GA). Program dibangun dengan bahasa C# yang memiliki fungsi obyektif minimasi rata-rata *flow time* dan *makespan*. Perancangan penjadwalan dilakukan terpisah antara *stage-1* dan *stage-2* kemudian digabungkan secara manual dalam satu *gant chart* penjadwalan. Metode penggabungan yang hanya menempelkan hasil *stage-2* pada *stage-1* menghasilkan waktu *idle* sehingga suatu prosedur penyesuaian diperlukan untuk menghilangkannya. Hasil penjadwalan dengan kriteria minimasi rata-rata *flow time* mencapai 737,3 menit dan kriteria *makespan* mencapai 1.280 menit. Rata-rata *flow time* metode GA lebih cepat 6,7 menit dibandingkan metode SPT. Sedangkan *makespan* metode GA lebih cepat 50 menit dibandingkan metode SPT. Jumlah perakas potong yang dibutuhkan pada mesin 1 adalah 1.260 unit, mesin 2 adalah 1.250 unit, mesin 3 adalah 1.250 unit, dan mesin 4 adalah 1.280 unit.

Kata Kunci— Flexible Manufacturing System (FMS), algoritma genetika, penjadwalan *job*, rata-rata *flow time*, *makespan*

I. PENDAHULUAN

Dalam menghadapi persaingan pasar dan menjaga kelangsungan bisnisnya, suatu perusahaan perlu memperhatikan beberapa aspek seperti biaya (*cost*), kualitas (*quality*), responsif (*responsiveness*), dan kecepatan (*delivery*) [1]. Dalam bidang manufaktur, salah satu upaya yang dapat dilakukan perusahaan untuk mengoptimalkan beberapa aspek tersebut adalah dengan mengoptimalkan penggunaan sumber daya dalam perusahaan melalui perancangan suatu penjadwalan produksi yang baik.

Berkaitan dengan aspek *responsiveness* yang mengacu kepada fleksibilitas, perusahaan dituntut memiliki kemampuan merespon berbagai perubahan secara efisien. Salah satu upaya untuk mencapainya adalah dengan penerapan teknologi *Flexible Manufacturing System* (FMS) dalam kegiatan produksi. Penerapan FMS sangat sesuai dengan industri manufaktur yang memiliki tingkat volume produksi menengah dan variasi produk menengah [2]. Apabila upaya penerapan FMS dilakukan oleh perusahaan, dalam pembuatan programnya tetap dibutuhkan perancangan penjadwalan yang baik sehingga penggunaan sumber daya yang ada serta waktu pengerjaan produk dapat tetap optimal dan perusahaan dapat mencapai aspek-aspek yang dibutuhkan dalam menghadapi persaingan pasar.

Penelitian ini akan membahas bagaimana perancangan penjadwalan beberapa *job* yang dilakukan pada satu perusahaan manufaktur pesawat terbang yang menggunakan teknologi *Flexible Manufacturing System* (FMS). Penelitian dilakukan pada sebuah *work center* yang terdiri dari empat mesin CNC paralel, di mana masing-masing mesin memiliki konfigurasi perakas potong dan karakteristik yang sama (Identik). Jumlah *part* yang akan dikerjakan berjumlah 15 *part*, di mana dalam satu *job* pengerjaan *part* tersebut terdiri dari satu hingga dua *stage* dan di dalam *stage* tersebut terdapat beberapa operasi. *Job* yang dikerjakan bersifat independen. Dalam pengerjaannya terdapat ketentuan yang tidak boleh

dilanggar yaitu job pada stage-2 baru dapat dikerjakan bila job pada stage-1 telah selesai dikerjakan.

Terdapat beberapa metode yang dapat digunakan dalam perancangan penjadwalan job seperti pendekatan analitik, pendekatan heuristic, dan pendekatan metaheuristic. Perancangan penjadwalan job menggunakan model matematis telah dilakukan oleh Setiawan, dkk. [3] sedangkan untuk penggunaan metode heuristic yaitu Short Processing Time (SPT) telah dilakukan dalam penelitian Setiawan, dkk. [4]. Perancangan penjadwalan dengan metode metaheuristic menggunakan metode Genetic Algorithm sebenarnya sudah dilakukan dalam penelitian Setiawan, dkk. [4]. Namun, penyelesaian yang dilakukan secara manual mengakibatkan waktu komputasi sangat lama sehingga hanya dilakukan selama 3 generasi.

Pada penelitian ini, metode Genetic Algorithm yang digunakan pada penelitian Setiawan, dkk. [4] akan dirancang menggunakan software pemrograman Microsoft Visual Studio C#. Hal ini dilakukan agar proses perhitungan berjalan lebih otomatis menggunakan program komputer sehingga diharapkan diperoleh hasil penjadwalan yang dapat memenuhi kriteria rata-rata flow time minimum dan kriteria makespan minimum. Selain itu penerapan metode Genetic Algorithm pada penelitian ini dikembangkan dan diterapkan untuk penjadwalan job yang lebih spesifik yakni tingkat stage. Selain jadwal produksi, penelitian ini akan menentukan jumlah perkakas potong yang harus disediakan di setiap mesin untuk menjalankan proses produksi.

II. METODOLOGI PENELITIAN

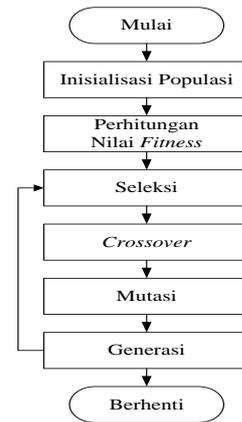
Flexible Manufacturing System (FMS) adalah serangkaian kumpulan mesin yang saling terhubung dengan sistem transportasi yang membawa benda kerja dalam pallet atau unit interface lainnya sehingga proses registrasi benda kerja-mesin menjadi akurat, cepat dan otomatis. Sebuah komputer pusat mengendalikan baik mesin maupun sistem transportasinya.

Penjadwalan dapat dikategorikan sebagai proses pengalokasian sumber daya untuk menyelesaikan sekumpulan tugas [5]. Menurut Morton dan Pentico dalam Sipper, D. dan Bulfin (1997), penjadwalan merupakan proses pengorganisasian, pemilihan, penggunaan waktu untuk menangani aktivitas-aktivitas yang diperlukan untuk memproduksi produk tertentu pada waktu tertentu sesuai dengan jumlah waktu yang tersedia dan keterbatasan antara aktivitas dan sumber daya yang tersedia [6].

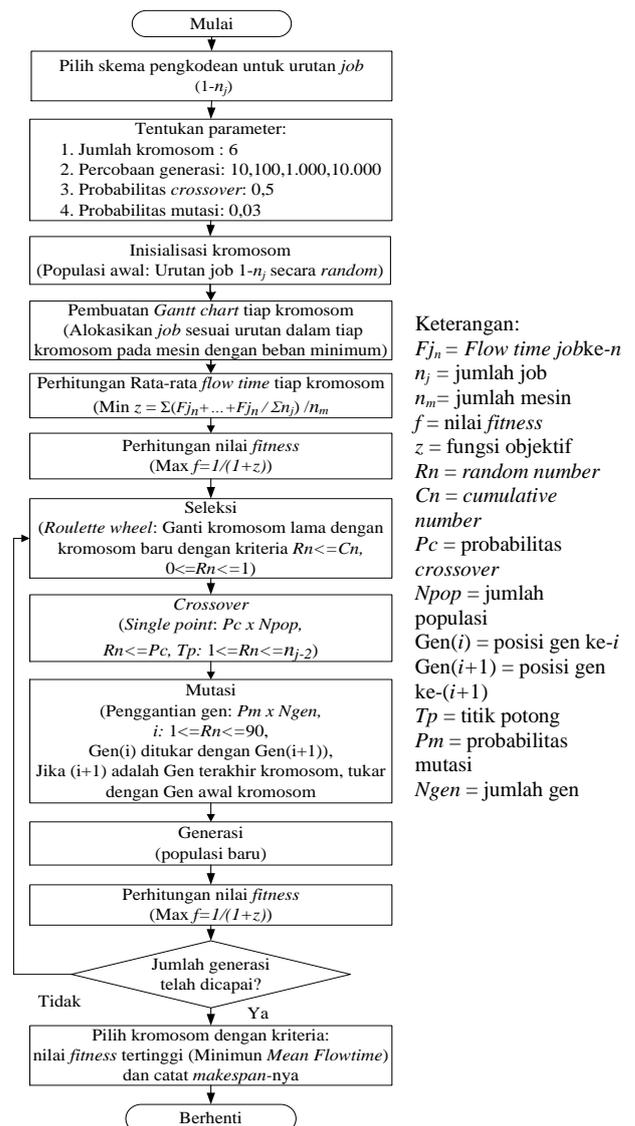
Secara umum proses perancangan penjadwalan job dengan metode GA pada penelitian ini terdiri dari beberapa tahapan proses yang dapat dilihat pada Gambar 1 dan Gambar 2.

III. HASIL DAN PEMBAHASAN

Data-data yang dipakai dalam penelitian ini merupakan data hipotetik berupa data waktu pengerjaan part untuk masing-masing stage seperti tercantum pada Tabel I.



Gambar 1. Metodologi GA secara umum

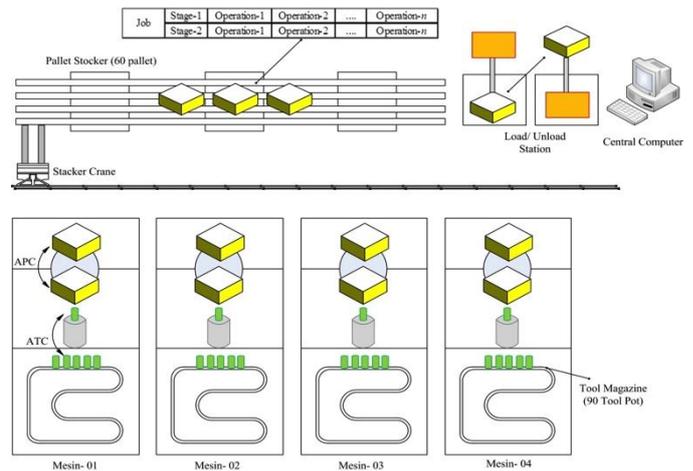


Keterangan:
 Fj_n = Flow time jobke-n
 n_j = jumlah job
 n_m = jumlah mesin
 f = nilai fitness
 z = fungsi objektif
 Rn = random number
 Cn = cumulative number
 Pc = probabilitas crossover
 $Npop$ = jumlah populasi
 $Gen(i)$ = posisi gen ke-i
 $Gen(i+1)$ = posisi gen ke-(i+1)
 Tp = titik potong
 Pm = probabilitas mutasi
 $Ngen$ = jumlah gen

Gambar 2 Perancangan penjadwalan Job GA

TABEL I
DATA JOB DAN WAKTU PROSES

Job	Part Number	Waktu Proses (menit)		Total Waktu Proses (menit)
		Stage-1	Stage-2	
1	PN-01	80	-	80
2	PN-02	240	240	480
3	PN-03	280	280	560
4	PN-04	240	120	360
5	PN-05	90	-	90
6	PN-06	100	-	100
7	PN-07	240	200	440
8	PN-08	240	120	360
9	PN-09	280	280	560
10	PN-10	240	240	480
11	PN-11	160	240	400
12	PN-12	320	200	520
13	PN-13	240	200	440
14	PN-14	90	-	90
15	PN-15	80	-	80



Gambar 3 Tata letak dan konfigurasi FMS [5]

Pengerjaan part dilakukan pada FMS dalam satu work center yang terdiri dari empat mesin CNC parallel yang memiliki konfigurasi perkakas potong dan karakteristik yang sama (Identik). Perkakas potong yang dipakai seperti perkakas potong untuk drilling dan milling disimpan pada tool magazine. Tiap mesin memiliki kapasitas 90 tool pot dalam tool magazine yang dilengkapi dengan Automatic Pallet Changer (APC) dan Automatic Tool Changer (ATC). APC berguna untuk mengganti benda kerja dari luar ke dalam ruangan permesinan dalam mesin CNC. Sementara ATC berguna untuk memindahkan perkakas potong dari spindle ke tool magazine di mana perkakas potong disimpan dalam mesin CNC.

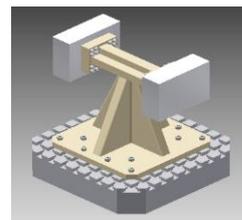
FMS memiliki pallet stocker yang berguna untuk menyimpan material baik itu sedang menunggu dikerjakan maupun sudah selesai dikerjakan pada saat kegiatan pengerjaan dilakukan tanpa manusia. Sistem material handling dilakukan dengan stacker crane. Pada FMS terdapat dua loading dan unloading station yang ditujukan untuk memasang benda kerja yang akan dikerjakan pada fixture. Tata letak dan konfigurasi FMS dapat dilihat pada Gambar 3.

Benda kerja yang dikerjakan mempunyai dua permukaan di mana masing-masing permukaan harus di-setup pada sebuah fixture. Satu job diasumsikan terdiri dari satu benda kerja dengan dua buah permukaan seperti dijelaskan pada Gambar 4. Sebuah permukaan dikerjakan dalam beberapa operasi yang disebut dengan sebuah stage. Benda kerja dengan dua buah permukaan membutuhkan dua stage. Masing-masing permukaan membutuhkan setup pada fixture seperti dijelaskan pada Gambar 5.

Sekumpulan job yang akan dikerjakan pada FMS tersebut dipersiapkan dan dirancang untuk satu periode sehingga dapat beroperasi hampir selama 24 jam tanpa operator dan job yang diberikan pada hari tersebut harus diselesaikan pada hari tersebut juga.



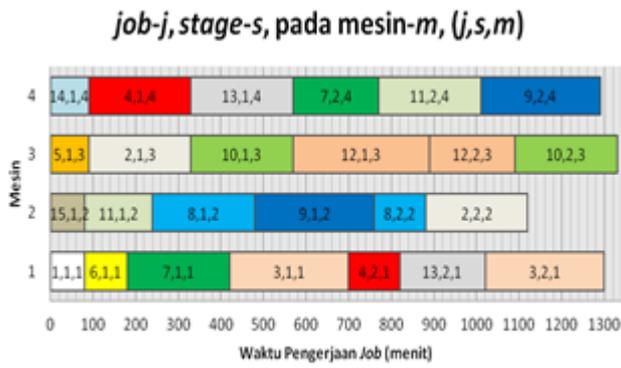
Gambar 4 Dua buah permukaan benda kerja dalam dua stage [8]



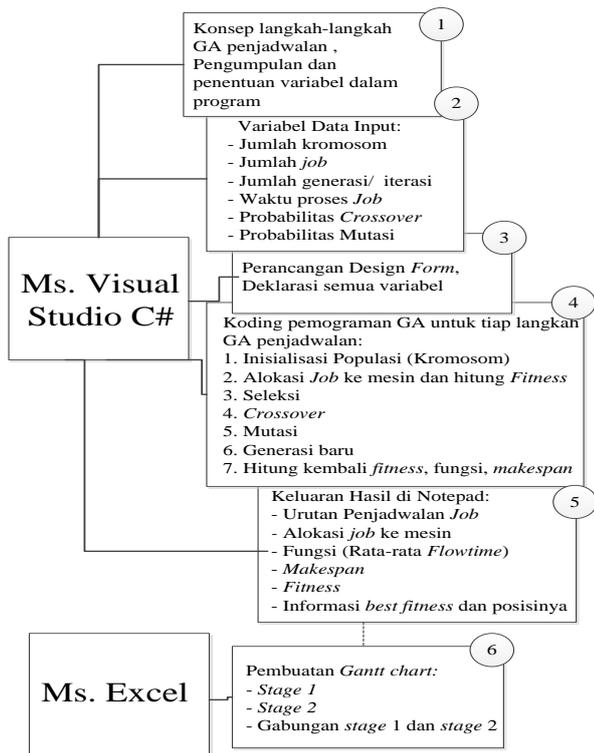
Gambar 5 Dua buah permukaan benda kerja dalam dua stage [8]

Algoritma genetika adalah algoritma komputasi yang diinspirasi teori evolusi yang kemudian diadopsi menjadi algoritma komputasi yang biasa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Algoritma ini didasarkan pada proses genetik yang ada dalam makhluk hidup yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti prinsip seleksi alam atau “siapa yang kuat, dia yang bertahan (survive)”. Dengan meniru teori evolusi ini, algoritma genetika dapat digunakan untuk mencari solusi permasalahan-pemmasalahan dalam dunia nyata [7].

Sebagai bahan perbandingan dengan metode GA, disusunlah sebuah penjadwalan dengan metode heuristik Short Processing Time (SPT). Metode SPT dirancang untuk tiap stage secara terpisah dan setelahnya baru digabungkan menjadi satu penjadwalan gabungan, dan bila ada waktu idle dapat dilakukan beberapa penyesuaian. Hasil perancangan SPT yang didapatkan dengan cara tersebut adalah rata-rata flow time sebesar 744 menit dan makespan sebesar 1330 menit, seperti tercantum pada Gambar 6.



Gambar 6 Penjadwalan job stage-1 dan stage-2 dengan metode SPT yang sudah disesuaikan



Gambar 7 Pengembangan program penjadwalan GA

Pengembangan program penjadwalan GA yang dilakukan dalam Microsoft Visual Studio 2010 dengan bahasa C# dapat dilihat pada Gambar 7.

Perancangan program pada Gambar 7 mengikuti metodologi GA yang dijelaskan sebagai berikut:

1) Inisialisasi populasi (kromosom)

Pada tahap ini akan dimunculkan kromosom sesuai input user. Dalam masing-masing kromosom tersebut terdapat angka dari satu sampai jumlah job yang dimasukkan oleh user dengan urutan acak dan dengan ketentuan tidak terdapat urutan angka yang sama.

Contoh: bila input user untuk jumlah kromosom adalah dua, dan untuk jumlah job adalah 10 maka pada langkah ini akan muncul keluaran debug pada program sebagai berikut:

- Kromosom 1: 6 9 3 10 4 5 1 8 7 2
- Kromosom 2: 5 10 2 7 1 9 4 3 6 8

2) Alokasi job ke mesin dan hitung fitness.

Pada tahap ini waktu proses/pengerjaan job yang telah di-input user akan dialokasikan ke sejumlah mesin yang jumlahnya ditentukan oleh user. Dalam studi kasus empat mesin sesuai dengan urutan yang terdapat dalam kromosom, namun tetap memperhatikan beban waktu minimum yang ada pada mesin. Setelah dialokasikan, barulah dihitung rata-rata flow time sesuai metodologi. Setelah didapatkan rata-rata flow time sebagai fungsi tujuan, lanjutkan dengan menghitung fitness. Contohnya sebagai berikut:

Misalkan waktu proses untuk job ke-1 sampai job ke-10 adalah: 240, 280, 120, 200,120, 280, 240, 240, 200, 200. Jumlah mesin yang ditentukan adalah empat, jumlah kromosom dan urutannya sama dengan contoh pada step 1 yaitu 6 9 3 10 4 5 1 8 7 2. Maka alokasi dalam mesin untuk kromosom 1 adalah sebagai berikut:

- Mesin 1: 6 8
- Mesin 2: 9 5 7
- Mesin 3: 3 4 2
- Mesin 4: 10 1
- Mesin 1: 280 240
- Mesin 2: 200 120 240
- Mesin 3: 120 200 280
- Mesin 4: 200 240

Pada program, perhitungan fungsi dan fitness langsung pada hasil karena telah dimasukkan rumus perhitungan, sehingga agar lebih jelas akan dijabarkan cara perhitungan rata-rata flow time-nya secara manual seperti ditunjukkan Tabel II.

Lalu rata-rata flow time dari masing-masing mesin tersebut dirata-rata kembali sehingga didapatkan rata-rata flow time sebesar 356,67 menit. Kemudian bila dihitung fitness sesuai rumus dalam metodologi, didapatkan fitness sebesar 0,0028.

TABEL II

CONTOH PERHITUNGAN RATA-RATA FLOW TIME

Mesin 1			Mesin 2		
Job	Waktu Proses	Flow Time	Job	Waktu Proses	Flow Time
6	280	280	9	200	200
8	240	520	5	120	320
			7	240	560
Total		800	Total		1080
Rata-Rata		400	Rata-Rata		360
Mesin 3			Mesin 4		
Job	Waktu Proses	Flow Time	Job	Waktu Proses	Flow Time
3	120	120	10	200	200
4	200	320	1	240	440
2	280	600			
Total		1040	Total		640
Rata-Rata		346.67	Rata-Rata		320

3) *Seleksi*

Pada tahap ini, dilakukan seleksi untuk menentukan kromosom-kromosom yang akan digunakan pada proses selanjutnya. Metode yang digunakan ialah metode *Roulette Wheel*. Metode seleksi *Roulette Wheel* dilakukan dengan memilih kromosom-kromosom yang memiliki persentase atau probabilitas kumulatif yang menjadi area dimana bilangan acak tersebut berada. Dalam program, kromosom pengganti dapat dicari dengan cara mencari selisih bilangan acak (*Random number*) dengan kumulatif. Jadi definisi mendekati atau “berada di area” adalah kromosom yang memiliki selisih paling kecil. Setelah semua kromosom mendapatkan kromosom pengganti, maka isi kromosom yang lama segera digantikan dengan isi kromosom pengganti tersebut. Ilustrasinya terdapat pada Tabel III. di mana pada kasus tersebut didapatkan kromosom pengganti untuk kromosom 1 adalah kromosom 2 karena hasil $|Rn_1 - Cn_2|$ lebih kecil dari $|Rn_1 - Cn_1|$, dan begitupun sebaliknya kromosom pengganti untuk kromosom 2 adalah kromosom 1 karena hasil $|Rn_2 - Cn_1|$ lebih kecil dari $|Rn_2 - Cn_2|$.

4) *Crossover*

Tahap ini dilakukan untuk menukarkan beberapa gen yang terdapat pada dua kromosom untuk menghasilkan individu atau kromosom baru. Dalam penelitian ini digunakan metode *Single Point Crossover* dengan probabilitas *crossover* sebesar 50% (0,5).

Metode *Single Point Crossover* dilakukan dengan membangkitkan bilangan acak dari 0 sampai 1 sebanyak jumlah kromosom. Kemudian pilih n kromosom yang memiliki selisih terkecil antara bilangan acak dengan probabilitas *crossover*. Di mana n adalah pembulatan ke atas dari hasil kali probabilitas *crossover* dengan m jumlah kromosom yang ada. Kemudian bangkitkan bilangan acak dari satu sampai sejumlah *job* yang di-*input user* dikurangi dua ($n_j - 2$), sebanyak n jumlah kromosom yang akan dilakukan *crossover*. Bilangan acak tersebut akan menentukan titik persilangan (*crossover*) gen yang akan ditukarkan antara dua kromosom yang akan dikawinkan. Ketentuan dalam program adalah seluruh gen pada kromosom induk pertama sebelum titik potong akan disalin ke dalam kromosom anak, sisa sambungannya diambil dari gen kromosom induk kedua setelah titik potong namun tidak boleh ada gen yang terulang kembali. Jika terdapat gen yang terulang, program akan melewatinya dan tetap membaca sampai akhir urutan, bila masih ada gen yang hilang, program akan meneruskan pembacaan gen dari gen paling awal milik induk kedua. Ilustrasinya sebagai berikut:

TABEL III

ILUSTRASI PENCARIAN KROMOSOM PENGGANTI (SELEKSI)

Kromosom	<i>Fitness</i>	Probabilitas	Kumulatif	<i>Random</i>	$ Rn_1 - Cn_1 $	$ Rn_1 - Cn_2 $	Kromosom
1	0,00280	0,507	0,507	0,913	0,406	0,488	2
2	0,00272	0,493	1,000	0,019	0,087	0,981	1
Total	0,00552	1,000					

Terdapat total dua kromosom, probabilitas *crossover* adalah 0,5 dan jumlah *job* adalah 10 buah, isi urutan pada kromosom adalah sebagai berikut:

- Kromosom 1: 6 9 3 10 4 5 1 8 7 2
- Kromosom 2: 5 10 2 7 1 9 4 3 6 8
- Kromosom 3: 6 8 2 7 9 1 4 3 5 10

Jumlah kromosom yang akan di-*crossover* adalah $0,5 \times 3 = 1,5 \approx 2$ kromosom, maka *random number* untuk titik potong yang dimunculkan adalah dua buah (dengan *range* 1-8), misalkan dipilih bilangan acak 3 dan 7. Ilustrasi *crossover* kromosom ditunjukkan Tabel IV.

Berdasarkan Tabel IV, kromosom yang di-*crossover* ada 2 buah. Dan berikut adalah contoh *crossover*-nya:

Crossover 1:

Induk 1 adalah kromosom 2, dan induk 2 adalah kromosom 3, anaknya adalah *offspring* kromosom 2. Titik potong = 3.

- Induk 1 (Kr 2) : 5 10 2 | 7 1 9 4 3 6 8
- Induk 2 (Kr 3) : 6 8 2 | 7 9 1 4 3 5 10
- Offspring 2* : 5 10 2 7 9 1 4 3 5 6 8

Crossover 2:

Induk 1 adalah kromosom 3, dan induk 2 adalah kromosom 2, anaknya adalah *offspring* kromosom 3. Titik potong = 7.

- Induk 1 (Kr 3) : 6 8 2 7 9 1 4 | 3 5 10
- Induk 2 (Kr 2) : 5 10 2 7 1 9 4 | 3 6 8
- Offspring 3* : 6 8 2 7 9 1 4 3 5 10

Gen *Offspring* akan menggantikan gen kromosom lama, sebagai contoh gen *offspring 2* berarti ia akan menggantikan gen kromosom 2, dan gen *offspring 3* akan menggantikan gen kromosom 3.

5) *Mutasi*

Pada tahap ini dilakukan pemasukan beberapa gen baru untuk menggantikan beberapa gen yang hilang di dalam satu individu atau kromosom akibat proses *crossover* atau perkawinan silang. Probabilitas mutasi pada penelitian ini adalah 3%. Mutasi dilakukan dengan menghitung total gen keseluruhan (m) dengan mengalikan jumlah n kromosom yang di-*input* dengan jumlah n_j *job* yang di-*input*. Selanjutnya hitung jumlah gen yang akan dimutasi (o) dihitung dengan mengalikan hasil total gen keseluruhan (m) dengan probabilitas mutasi. Lalu bangkitkan bilangan acak dari 1 sampai m sebanyak jumlah gen yang akan dilakukan mutasi (o). Bilangan acak tersebut akan menentukan posisi gen (i) yang akan ditukarkan dengan gen setelahnya ($i+1$). Namun apabila gen setelahnya adalah gen yang berasal dari beda kromosom, maka gen (i) akan dipertukarkan dengan gen paling awal dalam kromosom. Hal ini dilakukan agar tidak ada pengulangan *job* dalam suatu kromosom. Ilustrasinya adalah sebagai berikut:

TABEL IV

ILUSTRASI PEMILIHAN KROMOSOM YANG DI-CROSSOVER

Kromosom	<i>Random Number (Rn)</i>	Probabilitas <i>Crossover (Pc)</i>	$ Rn - Pc $
Kromosom 1	0,913	0,5	0,413
Kromosom 2	0,492	0,5	0,008
Kromosom 3	0,667	0,5	0,167

Misalkan terdapat dua kromosom, dengan jumlah job 10, dengan isi sebagai berikut:

Kromosom 1 : 6 9 3 10 4 5 1 8 7 2

Kromosom 2 : 5 10 2 7 1 9 4 3 6 8

Total gen keseluruhan (o) = jumlah kromosom (n) \times jumlah job (n_j) = $2 \times 10 = 20$ gen.

Untuk menghitung jumlah gen yang dimutasi (o) = total gen keseluruhan (m) \times probabilitas mutasi = $20 \times 0,03 = 0,6 \approx 1$.

Bila random number yang muncul (Range 1-20) = 20 maka:

Kromosom 1 : 6 9 3 10 4 5 1 8 7 2

Kromosom 2 : 5 10 2 7 1 9 4 3 6 8

Gen yang dimutasi adalah angka 8 yang dicetak tebal, berarti sesuai ketentuan gen tersebut akan dipertukarkan dengan gen awal kromosomnya yaitu angka 5, maka hasil akhirnya ialah:

Kromosom 1 : 6 9 3 10 4 5 1 8 7 2

Kromosom 2 : 8 10 2 7 1 9 4 3 6 5

6) Generasi

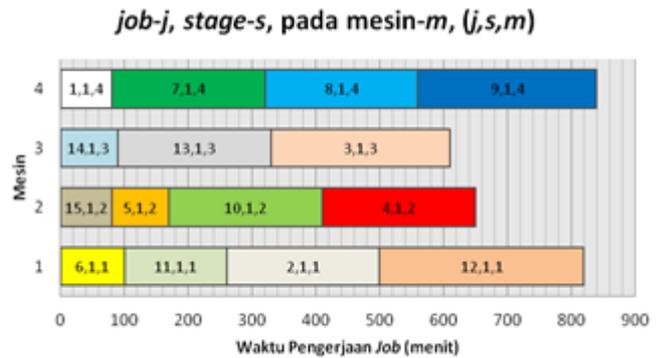
Merupakan individu baru yang merupakan solusi dari permasalahan; dimana solusi pada penelitian tentang penjadwalan mesin adalah urutan penjadwalan job yang akan diproses pada empat mesin dalam FMS. Generasi merupakan hasil urutan penjadwalan yang telah melalui semua proses dari step satu sampai lima. Individu yang telah melalui kelima proses sebelumnya dialokasikan kembali ke dalam mesin dan dihitung kembali nilai rata-rata flow time, fitness, dan makespan-nya.

Program GA yang dirancang dipakai untuk melakukan uji generasi sebanyak 5 kali, masing-masing untuk jumlah generasi 10, 100, 1.000, dan 10.000 untuk tiap stage. Waktu komputasi GA stage 1 dengan bahasa C# yang dieksekusi menggunakan computer dengan prosesor Intel Core i5 @2.25 GHz RAM 4 GB Operating System Windows 8.1 64-bit dapat dilihat pada Tabel V.

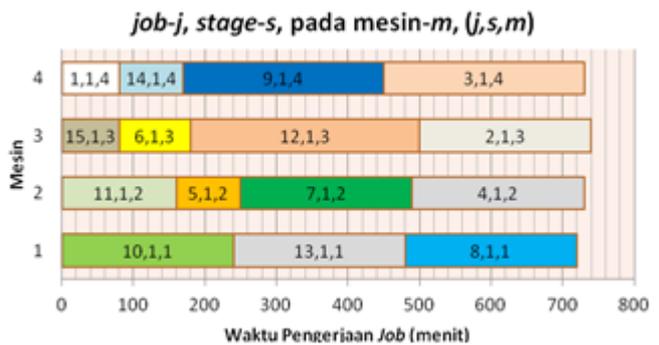
Pada percobaan untuk stage-1 berdasarkan kriteria minimasi rata-rata flow time didapatkan rata-rata flow time 385,208 menit dan makespan 840 menit dengan penjadwalan seperti pada Gambar 8. Sementara untuk kriteria minimasi makespan didapatkan rata-rata flow time 340 menit dan make-

span 640 menit dengan penjadwalan seperti pada Gambar 9.

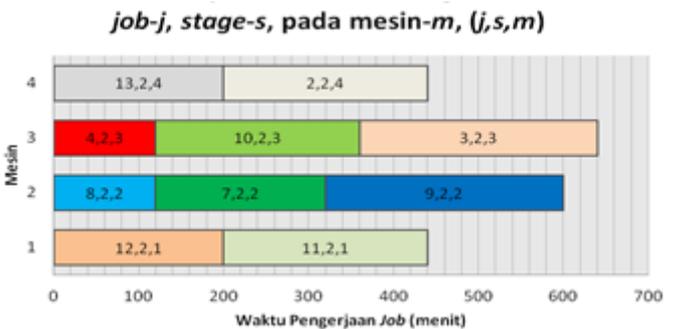
Pada percobaan untuk stage-2 berdasarkan kriteria minimasi rata-rata flow time didapatkan rata-rata flow time 450 menit dan makespan 740 menit dengan penjadwalan seperti pada Gambar 10. Sementara untuk kriteria minimasi makespan didapatkan rata-rata flow time 358,33 menit dan makespan 560 menit dengan penjadwalan seperti pada Gambar 11. Waktu komputasi GA stage 2 dapat dilihat pada Tabel VI.



Gambar 8 Penjadwalan job stage-1 berdasarkan kriteria minimasi rata-rata flow time



Gambar 9 Penjadwalan job stage-1 berdasarkan kriteria minimasi makespan



Gambar 10 Penjadwalan job stage-2 berdasarkan kriteria minimasi rata-rata flow time

TABEL V

WAKTU KOMPUTASI STAGE 1 PROGRAM GA

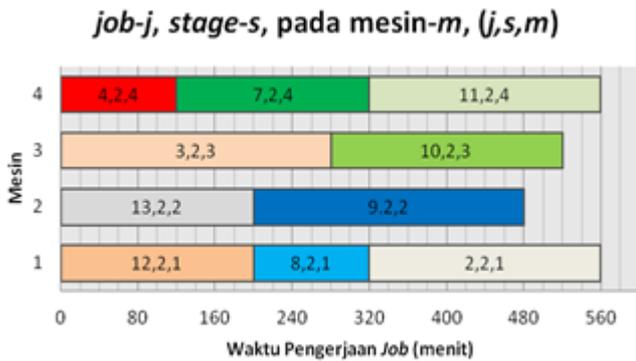
Generasi	Uji-	Waktu (detik)	Generasi	Uji-	Waktu (detik)
10	1	2,28	1.000	1	121,06
	2	2,28		2	111,81
	3	3,27		3	120,00
	4	3,27		4	107,06
	5	2,27		5	111,32
100	1	23,71	10.000	1	820,23
	2	22,45		2	610,27
	3	24,12		3	815,27
	4	23,87		4	846,01
	5	23,76		5	841,15

Tahap selanjutnya adalah menggabungkan penjadwalan stage-1 dan stage-2 secara manual berdasarkan kriterianya. Penjadwalan gabungan untuk kriteria minimasi rata-rata *flow time* dapat dilihat pada Gambar 12, sedangkan penjadwalan gabungan untuk kriteria minimasi *makespan* dapat dilihat pada Gambar 13.

Pada Gambar 12 yaitu penjadwalan gabungan untuk kriteria minimasi rata-rata *flow time* didapatkan rata-rata *flow time* 829,333 menit dan *makespan* sebesar 1480 menit. Sedangkan

Adanya waktu *idle* menyebabkan hasil penjadwalan gabungan untuk dua kriteria tersebut belum optimal, maka diperlukan langkah penyesuaian untuk menghilangkannya. Cara paling sederhana adalah menggeser *job* yang ada di sebelah kanan waktu *idle* ke sebelah kiri untuk mengisi waktu *idle*, namun harus tetap memperhatikan ketentuan bahwa *stage-2* tidak boleh dikerjakan sebelum *stage-1* selesai dikerjakan. Dengan cara penggeseran mungkin saja masih terdapat kemungkinan adanya waktu *idle* yang dapat diatasi dengan pertukaran *job* tertentu yang dapat disesuaikan sehingga waktu *idle* dapat dihilangkan dan ketentuan tidak terlanggar.

Dengan mengikuti langkah tersebut disusun suatu *gant* chart penjadwalan yang baru seperti pada Gambar 14 untuk kriteria minimasi rata-rata *flow time* dan pada Gambar 15 untuk kriteria minimasi *makespan*.



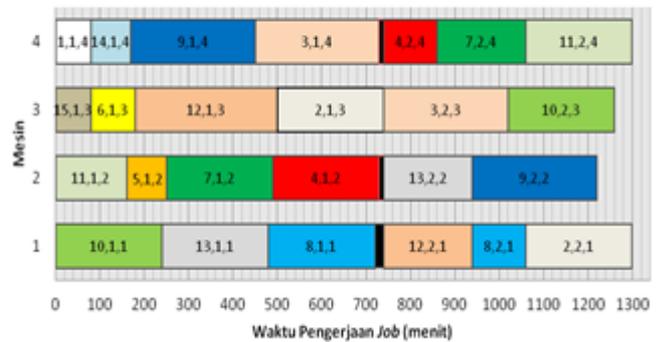
Gambar 11 Penjadwalan job stage-2 berdasarkan kriteria minimasi *makespan*

TABEL VI

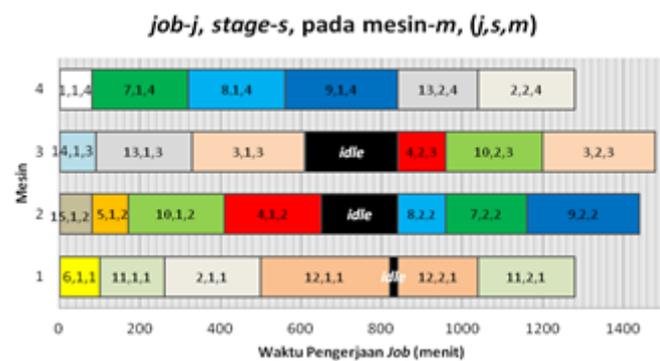
WAKTU KOMPUTASI STAGE 2 PROGRAM GA

Generasi	Uji-	Waktu (detik)	Generasi	Uji-	Waktu (detik)
10	1	1,04	1.000	1	84,13
	2	1,70		2	71,37
	3	0,86		3	66,66
	4	0,62		4	74,75
	5	0,27		5	50,22
100	1	8,06	10.000	1	339,10
	2	8,23		2	330,55
	3	8,75		3	527,72
	4	8,10		4	537,11
	5	7,95		5	547,55

job-j, stage-s, pada mesin-m, (j,s,m)



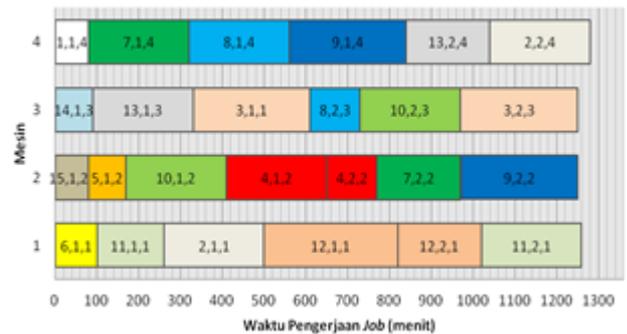
Gambar 13 Penjadwalan job stage-1 dan 2 dengan kriteria minimasi *makespan*



Gambar 12 Penjadwalan job stage-1 dan 2 dengan kriteria minimasi rata-rata *flow time*

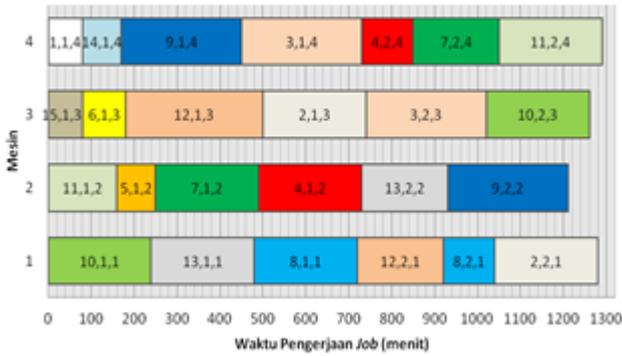
pada Gambar 13 untuk penjadwalan gabungan untuk kriteria *makespan* didapatkan rata-rata 781,333 menit dan *makespan* sebesar 1300 menit.

job-j, stage-s, pada mesin-m, (j,s,m)



Gambar 14 Penjadwalan job stage-1 dan 2 dengan kriteria minimasi rata-rata *flow time* yang sudah disesuaikan

job-j, stage-s, pada mesin-m, (j,s,m)



Gambar 15 Penjadwalan job stage-1 dan 2 dengan kriteria minimasi makespan yang sudah disesuaikan

Pada Gambar 14 penjadwalan dengan kriteria minimasi rata-rata flow time yang telah diperbaharui didapatkan hasil rata-rata flow time sebesar 737,3 menit dan makespan sebesar 1280 menit. Sementara pada Gambar 15 untuk penjadwalan dengan kriteria minimasi makespan yang baru didapatkan hasil rata-rata flow time sebesar 774 menit dan makespan sebesar 1290 menit. Ternyata hasil penjadwalan dengan kriteria rata-rata flow time minimum lebih baik ditinjau dari besarnya rata-rata flow time dan makespan-nya. Hal yang mungkin menyebabkan penjadwalan dengan kriteria minimasi makespan tidak dapat mencapai tujuannya adalah fungsi tujuan (objektif) utama dalam program yang menggunakan fungsi tujuan rata-rata flow time.

Untuk mengukur kinerja penjadwalan dengan metode GA yang telah dilakukan maka hasil penjadwalan tersebut dibandingkan dengan hasil penjadwalan dengan metode SPT yang telah dibuat sebelumnya, dilihat dari dua kriteria yakni rata-rata flow time dan makespan.

Berdasarkan kriteria minimasi rata-rata flow time, rata-rata flow time yang didapatkan dengan metode GA adalah 737,3 menit sedangkan untuk SPT yang telah disesuaikan didapatkan rata-rata flow time sebesar 744 menit sehingga disimpulkan bahwa metode GA memiliki rata-rata flow time lebih baik walaupun tidak terlalu signifikan yaitu sebesar 6,7 menit lebih cepat.

Berdasarkan kriteria minimasi makespan, besarnya makespan yang didapatkan dengan metode GA adalah 1280 menit sedangkan untuk SPT yang telah disesuaikan didapatkan makespan sebesar 1330 menit sehingga disimpulkan bahwa metode GA memiliki besar makespan lebih pendek yaitu sebesar 50 menit lebih cepat.

Penentuan perangkat potong dilakukan untuk penjadwalan job dua stage dengan kriteria minimasi rata-rata flow time yang telah disesuaikan karena penjadwalan ini memiliki hasil rata-rata flow time maupun makespan paling minimum. Penentuan perangkat potong tersebut tercantum pada Tabel VII sampai Tabel X.

TABEL VII

PENENTUAN PERKAKAS POTONG UNTUK MESIN 1

Mesin 1							
Triplets (j,s,m)	6,1,1	11,1,1	2,1,1	12,1,1	12,2,1	11,2,1	Total
tj	100	160	240	320	200	240	
CTj	100	260	500	820	1020	1260	
Endmill Ø 14	CC 1,1,1 20	CC 2,1,1 40	CC 3,1,1 35	CC 4,1,1 50	CC 5,1,1 45	CC 6,1,1 40	230
Endmill Ø 12	CC 1,2,1 15	CC 2,2,1 50	CC 3,2,1 40	CC 4,2,1 40	CC 5,2,1 40	CC 6,2,1 40	145
Facemill Ø 80	CC 1,3,1	CC 2,3,1	CC 3,3,1	CC 4,3,1	CC 5,3,1	CC 6,3,1	
			45				45
Endmill Ø 16 R2	CC 1,4,1 30	CC 2,4,1 30	CC 3,4,1 30	CC 4,4,1 30	CC 5,4,1 30	CC 6,4,1 30	60
Endmill Ø 14 R2	CC 1,5,1 20	CC 2,5,1 25	CC 3,5,1 25	CC 4,5,1 60	CC 5,5,1 60	CC 6,5,1 60	105
Endmill Ø 12 R2	CC 1,6,1	CC 2,6,1	CC 3,6,1 30	CC 4,6,1	CC 5,6,1 45	CC 6,6,1	75
Endmill Ø 10 R2	CC 1,7,1	CC 2,7,1	CC 3,7,1 35	CC 4,7,1 60	CC 5,7,1 60	CC 6,7,1 50	145
Endmill Ø 8 R2	CC 1,8,1	CC 2,8,1	CC 3,8,1	CC 4,8,1 50	CC 5,8,1 30	CC 6,8,1 60	140
Ballnose Ø 16	CC 1,9,1 15	CC 2,9,1	CC 3,9,1	CC 4,9,1	CC 5,9,1	CC 6,9,1 40	55
Ballnose Ø 12	CC 1,10,1 20	CC 2,10,1 20	CC 3,10,1	CC 4,10,1	CC 5,10,1 30	CC 6,10,1	70
Ballnose Ø 10	CC 1,11,1 30	CC 2,11,1	CC 3,11,1	CC 4,11,1 60	CC 5,11,1 50	CC 6,11,1 20	160
Ballnose Ø 8	CC 1,12,1	CC 2,12,1	CC 3,12,1	CC 4,12,1	CC 5,12,1	CC 6,12,1 30	30

TABEL VIII

PENENTUAN PERKAKAS POTONG UNTUK MESIN 2

Mesin 2								
Triplets (j,s,m)	15,1,2	5,1,2	10,1,2	4,1,2	4,2,2	7,2,2	9,2,2	Total
tj	80	90	240	240	120	200	280	
CTj	80	170	410	650	770	970	1250	
Endmill Ø 14	CC 1,1,2 10	CC 2,1,2 40	CC 3,1,2 35	CC 4,1,2 35	CC 5,1,2 35	CC 6,1,2 60	CC 7,1,2 180	
Endmill Ø 12	CC 1,2,2 15	CC 2,2,2 15	CC 3,2,2 50	CC 4,2,2 20	CC 5,2,2 20	CC 6,2,2 50	CC 7,2,2 50	220
Facemill Ø 80	CC 1,3,2	CC 2,3,2	CC 3,3,2 60	CC 4,3,2	CC 5,3,2	CC 6,3,2	CC 7,3,2	60
Endmill Ø 16 R2	CC 1,4,2 20	CC 2,4,2	CC 3,4,2	CC 4,4,2 50	CC 5,4,2	CC 6,4,2	CC 7,4,2	70
Endmill Ø 14 R2	CC 1,5,2 10	CC 2,5,2	CC 3,5,2	CC 4,5,2 40	CC 5,5,2	CC 6,5,2	CC 7,5,2	50
Endmill Ø 12 R2	CC 1,6,2	CC 2,6,2	CC 3,6,2 40	CC 4,6,2 50	CC 5,6,2	CC 6,6,2	CC 7,6,2	90
Endmill Ø 10 R2	CC 1,7,2 15	CC 2,7,2	CC 3,7,2 50	CC 4,7,2	CC 5,7,2	CC 6,7,2 35	CC 7,7,2 100	
Endmill Ø 8 R2	CC 1,8,2 10	CC 2,8,2	CC 3,8,2	CC 4,8,2	CC 5,8,2 15	CC 6,8,2 30	CC 7,8,2 40	95
Ballnose Ø 16	CC 1,9,2 20	CC 2,9,2	CC 3,9,2	CC 4,9,2 35	CC 5,9,2 20	CC 6,9,2 20	CC 7,9,2 45	140
Ballnose Ø 12	CC 1,10,2 20	CC 2,10,2 20	CC 3,10,2	CC 4,10,2 10	CC 5,10,2 30	CC 6,10,2 30	CC 7,10,2 40	150
Ballnose Ø 10	CC 1,11,2	CC 2,11,2	CC 3,11,2	CC 4,11,2	CC 5,11,2 15	CC 6,11,2	CC 7,11,2	15
Ballnose Ø 8	CC 1,12,2	CC 2,12,2 15	CC 3,12,2	CC 4,12,2	CC 5,12,2 20	CC 6,12,2	CC 7,12,2 45	80

TABEL IX

PENENTUAN PERKAKAS POTONG UNTUK MESIN 3

Mesin 3							
Triplets (j,s,m)	14,1,3	13,1,3	3,1,3	8,2,3	10,2,3	3,2,3	Total
tj	90	240	280	120	240	280	
CTj	90	330	610	730	970	1250	
Endmill Ø 14	CC 1,1,3	CC 2,1,3	CC 3,1,3	CC 4,1,3	CC 5,1,3	CC 6,1,3	
		30	50		50	30	160
Endmill Ø 12	CC 1,2,3	CC 2,2,3	CC 3,2,3	CC 4,2,3	CC 5,2,3	CC 6,2,3	
	10	40	50	30	60	55	245
Facemill Ø 80	CC 1,3,3	CC 2,3,3	CC 3,3,3	CC 4,3,3	CC 5,3,3	CC 6,3,3	
		30	60				90
Endmill Ø 16 R2	CC 1,4,3	CC 2,4,3	CC 3,4,3	CC 4,4,3	CC 5,4,3	CC 6,4,3	
	25	40					65
Endmill Ø 14 R2	CC 1,5,3	CC 2,5,3	CC 3,5,3	CC 4,5,3	CC 5,5,3	CC 6,5,3	
		25	35				60
Endmill Ø 12 R2	CC 1,6,3	CC 2,6,3	CC 3,6,3	CC 4,6,3	CC 5,6,3	CC 6,6,3	
	20		35	20			75
Endmill Ø 10 R2	CC 1,7,3	CC 2,7,3	CC 3,7,3	CC 4,7,3	CC 5,7,3	CC 6,7,3	
	15	25				25	65
Endmill Ø 8 R2	CC 1,8,3	CC 2,8,3	CC 3,8,3	CC 4,8,3	CC 5,8,3	CC 6,8,3	
				15	45	40	100
Ballnose Ø 16	CC 1,9,3	CC 2,9,3	CC 3,9,3	CC 4,9,3	CC 5,9,3	CC 6,9,3	
			50	25	45	50	170
Ballnose Ø 12	CC 1,10,3	CC 2,10,3	CC 3,10,3	CC 4,10,3	CC 5,10,3	CC 6,10,3	
	20	50				30	100
Ballnose Ø 10	CC 1,11,3	CC 2,11,3	CC 3,11,3	CC 4,11,3	CC 5,11,3	CC 6,11,3	
				25	40		65
Ballnose Ø 8	CC 1,12,3	CC 2,12,3	CC 3,12,3	CC 4,12,3	CC 5,12,3	CC 6,12,3	
				5		50	55

TABEL X

PENENTUAN PERKAKAS POTONG UNTUK MESIN 4

Mesin 4							
Triplets (j,s,m)	1,1,4	7,1,4	8,1,4	9,1,4	13,2,4	2,2,4	Total
tj	80	240	240	280	200	240	
CTj	80	320	560	840	1040	1280	
Endmill Ø 14	CC 1,1,4	CC 2,1,4	CC 3,1,4	CC 4,1,4	CC 5,1,4	CC 6,1,4	
	10	45	40		45	45	185
Endmill Ø 12	CC 1,2,4	CC 2,2,4	CC 3,2,4	CC 4,2,4	CC 5,2,4	CC 6,2,4	
	10	30	20			30	90
Facemill Ø 80	CC 1,3,4	CC 2,3,4	CC 3,3,4	CC 4,3,4	CC 5,3,4	CC 6,3,4	
		25	50	30			105
Endmill Ø 16 R2	CC 1,4,4	CC 2,4,4	CC 3,4,4	CC 4,4,4	CC 5,4,4	CC 6,4,4	
		30	40	40			110
Endmill Ø 14 R2	CC 1,5,4	CC 2,5,4	CC 3,5,4	CC 4,5,4	CC 5,5,4	CC 6,5,4	
		25	40	40			105
Endmill Ø 12 R2	CC 1,6,4	CC 2,6,4	CC 3,6,4	CC 4,6,4	CC 5,6,4	CC 6,6,4	
	15	35		30			80
Endmill Ø 10 R2	CC 1,7,4	CC 2,7,4	CC 3,7,4	CC 4,7,4	CC 5,7,4	CC 6,7,4	
	10			50			60
Endmill Ø 8 R2	CC 1,8,4	CC 2,8,4	CC 3,8,4	CC 4,8,4	CC 5,8,4	CC 6,8,4	
	10				45	35	90
Ballnose Ø 16	CC 1,9,4	CC 2,9,4	CC 3,9,4	CC 4,9,4	CC 5,9,4	CC 6,9,4	
	25		50	50	30	40	195
Ballnose Ø 12	CC 1,10,4	CC 2,10,4	CC 3,10,4	CC 4,10,4	CC 5,10,4	CC 6,10,4	
		50		40		50	140
Ballnose Ø 10	CC 1,11,4	CC 2,11,4	CC 3,11,4	CC 4,11,4	CC 5,11,4	CC 6,11,4	
					40	20	60
Ballnose Ø 8	CC 1,12,4	CC 2,12,4	CC 3,12,4	CC 4,12,4	CC 5,12,4	CC 6,12,4	
					40	20	60

IV. KESIMPULAN

Berdasarkan penelitian mengenai perancangan penjadwalan job dua stage menggunakan metode Genetic Algorithm (GA) yang telah dilakukan dalam program komputer dapat ditarik kesimpulan sebagai berikut:

1. Hasil penjadwalan dengan kriteria minimasi rata-rata flow time menghasilkan flow time paling minimum, yaitu sebesar 737,3 menit dan makespan paling minimum, yaitu 1280 menit.
2. Rata-rata flow time minimum yang didapatkan dengan metode GA adalah 737,3 menit sedangkan rata-rata flow time minimum yang diperoleh dengan metode SPT adalah 744 menit sehingga dapat disimpulkan bahwa rata-rata flow time dengan metode GA lebih cepat 6,7 menit. Sedangkan untuk makespan, dengan metode GA diperoleh makespan minimum sebesar 1280 menit, sedangkan dengan menggunakan SPT yang telah disesuaikan diperoleh makespan minimum sebesar 1330 menit sehingga dapat disimpulkan makespan yang diperoleh dengan metode GA lebih cepat sekitar 50 menit.
3. Jumlah perkakas potong yang dibutuhkan pada mesin 1 sebesar 1.260 unit, mesin 2 sebesar 1.250 unit, mesin 3 sebesar 1.250 unit, dan mesin 4 sebesar 1.280 unit.

DAFTAR REFERENSI

- [1] E. Purnamawati. "Evaluasi Fleksibilitas Manufaktur: Studi Kasus Perusahaan Industri Kecil dan Menengah", *Proceeding Seminar Nasional Teknik Industri 2013*, Yogyakarta, Universitas Pembangunan Nasional, 2013.
- [2] A. Setiawan, L. Qashmal, R. Wangsaputra. "An Object-oriented Modelling of Production Scheduling for Flexible Manufacturing System", *Journal of Applied Mechanics and Materials*, **842**, 345-354. 2016.
- [3] A. Setiawan, R. Wangsaputra, A. Hakim H., Y. Yuwana M. "A Production Scheduling Model Considering Cutting Tools for an FMS to Minimize Makespan", *Proceeding of the Asia Pacific Industrial Engineering & Management System Conference*, APIEMS 2015, Ho Chi-Minh City, Vietnam, 2015.
- [4] A. Setiawan, Susan, E.K. Asih. "Penjadwalan Job Shop pada Empat Mesin Identik dengan Menggunakan Metode Shortest Processing Time dan Genetic Algorithm". *Jurnal Telematika ITHB*. **9**. 19-24. 2014.
- [5] K. R. Baker dan Trietsch. *Principles of Sequencing and Scheduling*, New York: John Wiley and Sons, Inc, 2009.
- [6] D. Sipper dan R. L. Bulfin, JR. *Production: Planning, Control, and Integration*, New York: McGraw-Hill Companies, Inc, 1997.
- [7] Sam'ani. "Rancang Bangun Sistem Penjadwalan Perkuliahan dan Ujian Akhir Semester Dengan Pendekatan Algoritma Genetika", *Thesis Magister Strata Dua*, Universitas Diponegoro, 1-46, 2012.
- [8] L. Qashmal. "Penjadwalan Produksi Pada Flexible Manufacturing System Dengan Metode Pemodelan Berorientasi Obyek", *Laporan Tugas Akhir*, Fakultas Teknik Mesin dan Dirgantara Institut Teknologi Bandung, 2015.

Teguh Ersada Natail Sitepu, lahir pada tahun 1991 di Medan, menerima gelar Sarjana Teknik dari USU jurusan Teknik Industri pada tahun 2014 dan gelar Magister Teknik dari ITB jurusan Teknik dan Manajemen Industri pada tahun 2017. Saat ini aktif sebagai Dosen Tetap di Departemen Logistik ITHB. Minat penelitian: Riset Operasi, Statistika Industri, dan Manajemen Inovasi.

Ari Setiawan, lahir pada tahun 1966 di Bandung, menerima gelar Sarjana Teknik dari ITB jurusan Teknik Mesin pada tahun 1990 dan gelar Magister Teknik dari ITB jurusan Teknik dan Manajemen Industri pada tahun 1997. Saat ini aktif sebagai Dosen Tetap di Departemen Teknik Industri ITHB. Minat penelitian: CIM, Proses Produksi, dan *Maintenance*.

Antonius Candra, lahir di Bandung pada 13 Oktober 1994. Jenjang pendidikan S1 di Teknik Industri ITHB dan lulus pada tahun 2016. Minat penelitian: penjadwalan, produksi.