

PENJADWALAN *FLOW SHOP* MENGGUNAKAN *HYBRID GENETIC ALGORITHM (HGA)*

Rachmad Hidayat
Jurusan Teknik Industri
Fakultas Teknik Universitas Trunojoyo

ABSTRAK

Studi ini memfokuskan pada masalah penyediaan sistem dan informasi penjadwalan *flow shop* yaitu lamanya waktu makespan untuk keseluruhan operasi *flow shop*. Operasi-operasi setiap *job* diproses dalam beberapa urutan. Paling banyak satu operasi dapat diproses pada setiap mesin, dan paling banyak satu operasi dari setiap *job* dapat diproses. Studi ini mengaplikasikan sebuah *Hybrid Genetic Algorithm (HGA)* dengan memasukkan *Tabu Search (TS)* sebagai prosedur perbaikan lokal ke dalam *Basic Algoritma Genetika*. *Algoritma Genetika* digunakan untuk melakukan global eksplorasi diantara populasi, sedangkan prosedur perbaikan lokal digunakan untuk melakukan lokal eksploitasi di sekitar kromosom. Penggabungan prosedur perbaikan lokal memungkinkan algoritma *HGA* melakukan pencarian genetika pada *subspace* optima lokal. Hasil perhitungan kondisi riil diperoleh waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir (*makespan*) sebesar 1022.1 menit. Hasil perhitungan pendekatan *Hybrid Genetic Algorithm (HGA)* diperoleh waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir (*makespan*) sebesar 862.4 menit, sehingga dapat ditarik kesimpulan bahwa pendekatan *Hybrid Genetic Algorithm (HGA)* mampu mengurangi waktu total operasi.

Keywords : Sistem dan informasi, *Flow Shop*, Waktu *Makespan*, *Tabu Search (TS)*, *Hybrid Genetic Algorithm (HGA)*

1. PENDAHULUAN

Tugas pokok manajemen adalah membuat keputusan. Tepat tidaknya suatu keputusan yang diambil akan tergantung pada kualitas informasi yang di peroleh, ketepatan waktu diperolehnya informasi dan memadai atau tidaknya informasi tersebut. Manajemen yang mempunyai keinginan untuk maju selalu mengembangkan dan membarui sistem informasi manajemen perusahaannya.

Sistem Informasi Manajemen adalah jaringan prosedur pengolahan data yang di kembangkan dalam suatu organisasi dan disatukan apabila di pandang perlu, dengan maksud memberikan data kepada manajemen setiap waktu di perlukan, baik data yang bersifat intern maupun yang bersifat ekstern, untuk dasar pengambilan keputusan dalam rangka mencapai tujuan organisasi. Secara umum tujuan dari pengembangan sistem informasi adalah sebagai berikut :

1. Mendapatkan keseragaman data dan informasi.
2. Mendapatkan informasi yang dapat dipercaya, valid dan benar.
3. Mendapatkan informasi dengan tepat waktu.

Sebenarnya sistem informasi manajemen bukanlah hal yang baru, hanya teknologi informasinya saja yang di perbarui yaitu dengan digunakannya perangkat keras komputer. Sebenarnya sebelum ada komputer teknik sistem informasi manajemen ini telah digunakan para manajer mengolah informasi untuk merencanakan dan mengendalikan operasi organisasinya. Komputer hanya menambah satu atau dua dimensi seperti kecepatan, ketelitian, volume data dengan mempertimbangkan alternatif-alternatif yang lebih banyak dalam pengambilan keputusan.

Pengembangan sistem (*system development*) dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki atau diganti disebabkan karena beberapa hal, antara lain :

1. Adanya permasalahan-permasalahan (*problem*) yang timbul pada sistem yang lama. Permasalahan yang timbul dapat berupa:

a. Ketidakberesan

Ketidakberesan dalam sistem yang lama menyebabkan sistem yang lama tidak dapat beroperasi sesuai dengan yang di harapkan. Ketidakberesan ini dapat berupa :

- ◆ Kecurangan-kecurangan yang disengaja.
- ◆ Kesalahan-kesalahan yang tidak sengaja yang juga dapat menyebabkan
- ◆ Kebenaran dari data kurang terjamin
- ◆ Tidak efisiennya operasi perusahaan.
- ◆ Tidak ditaatinya kebijaksanaan manajemen yang telah di tetapkan

b. Pertumbuhan Organisasi

Pertumbuhan organisasi yang menyebabkan harus disusunya sistem yang baru. Pertumbuhan organisasi di antaranya adalah kebutuhan informasi yang semakin luas dan volume pengolahan yang semakin meningkat. Karena adanya perubahan ini maka menyebabkan sistem lama tidak efektif lagi, sehingga sistem yang lama sudah tidak dapat memenuhi lagi semua kebutuhan informasi yang dibutuhkan manajemen.

2. Untuk meraih kesempatan-kesempatan (*opportunities*)

Teknologi informasi telah berkembang dengan cepatnya. Perangkat keras komputer, perangkat lunak dan teknologi komunikasi telah begitu cepat berkembang. Organisasi mulai merasakan bahwa teknologi informasi ini perlu digunakan dalam proses pengambilan keputusan yang akan di lakukan oleh manajemen. Dalam keadaan pasar persaingan bebas, kecepatan informasi atau efisiensi waktu sangat menentukan berhasil atau tidaknya strategi rencana-rencana yang telah disusun untuk meraih kesempatan-kesempatan yang ada. Jika tidak maka pesaing dapat memanfaatkannya, sedangkan perusahaan yang tidak dapat memanfaatkan teknologi, menyebabkan kesempatan-kesempatan akan jatuh ketangan pesaing. Kesempatan-kesempatan ini dapat berupa peluang-peluang pasar, pelayanan yang meningkat kepada konsumen dan lain sebagainya.

3. Adanya instuksi-instruksi (*directive*)

Penyusunan sistem yang baru dapat terjadi karena instruksi-instruksi dari pimpinan ataupun dari luar organisasi, seperti misalnya peraturan pemerintah. Secara umum indikator adanya permasalahan dan kesempatan yang dapat diraih sehingga menyebabkan sistem yang ada harus dikembangkan (diperbaiki/ditingkatkan/diganti) antara lain :

- ◆ Keluhan dari konsumen
- ◆ Pengiriman barang yang sering tertunda
- ◆ Produktifitas tenaga kerja yang rendah
- ◆ Banyaknya pekerja yang menganggur
- ◆ Kegiatan yang tumpang tindih
- ◆ Tanggapan yang lambat terhadap konsumen
- ◆ Kehilangan kesempatan kompetisi pasar
- ◆ Kesalahan-kesalahan manual yang tinggi

- ◆ Persediaan barang yang terlalu tinggi
- ◆ Pemesanan barang yang tidak efisien
- ◆ Biaya operasi yang tinggi
- ◆ File-file yang kurang teratur
- ◆ Keluhan dari supplier karena tertundanya pembayaran
- ◆ Bertumpuknya *back order* (tertundanya pengiriman karena kurangnya persediaan barang)
- ◆ Investasi yang tidak efisien
- ◆ Ramalan penjualan dan produksi yang tidak tepat
- ◆ Kapasitas produksi yang menganggur (*idle capacities*)

Pada umumnya penjadwalan flow shop mempertimbangkan kriteria tertentu sebagai fungsi tujuan dari permasalahan. Dan berawal dari sinilah dikembangkan metode-metode pemecahan masalah yang diharapkan dapat memberikan solusi yang lebih baik. Salah satu fungsi tujuan yang umum dipakai adalah *makespan*, yaitu keseluruhan waktu operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir.

Studi ini memfokuskan pada masalah penjadwalan flow shop yaitu lamanya waktu *makespan* untuk keseluruhan operasi flow shop. Masalah penjadwalan flow shop dengan fungsi tujuan *makespan* dapat diuraikan sebagai berikut: ada sejumlah n job yang harus diproses pada sejumlah m mesin. Setiap job terdiri dari l operasi, dan setiap operasi diproses pada mesin yang berbeda selama periode waktu tertentu. Operasi-operasi setiap job diproses dalam beberapa urutan. Paling banyak satu operasi dapat diproses pada setiap mesin, dan paling banyak satu operasi dari setiap job dapat diproses. Masalahnya adalah bagaimana menemukan sebuah penjadwalan operasi pada mesin yang dapat mengurangi waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir.

Disini peneliti mengaplikasikan sebuah hybrid genetic algorithm (HGA) dengan memasukkan tabu search (TS) sebagai prosedur perbaikan lokal ke dalam basic algoritma genetika. Algoritma genetika digunakan untuk melakukan global eksplorasi diantara populasi, sedangkan prosedur perbaikan lokal digunakan untuk melakukan lokal exploitasi di sekitar kromosom. Penggabungan prosedur perbaikan lokal memungkinkan algoritma HGA melakukan pencarian genetika pada subspace optima lokal.

2. TUJUAN

Tujuan yang ingin dicapai dari studi ini adalah :

1. Menyusun suatu model penjadwalan flow shop.
2. Menentukan urutan operasi dari setiap job pada setiap mesin agar dapat mengurangi waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir.
3. Membuat sistem informasi penjadwalan flow shop.
4. Melakukan simulasi penjadwalan flow shop.

3. PENJADWALAN FLOW SHOP DENGAN HYBRID GENETIC ALGORITHM

Dalam banyak fasilitas manufaktur dan perakitan, terdapat sejumlah operasi yang harus dilakukan pada setiap pekerjaan dalam urutan yang sama yang mengakibatkan pekerjaan harus mengikuti rute yang sama. Mesin-mesin diasumsikan terpasang secara seri dalam kondisi ini sebagai sebuah flow shop.

Problem flow shop sequencing dideskripsikan sebagai berikut :

1. Terdapat sejumlah m mesin dan n job;
2. Setiap job terdiri dari m operasi;

3. Setiap operasi diproses pada mesin yang berbeda;
4. n job harus diproses dengan urutan yang sama pada m mesin;
5. Waktu operasi job i pada mesin j dilambangkan dengan t_{ij} . ($i = 1, 2, \dots, n$; $j = 1, 2, \dots, m$).

Algoritma genetika merupakan cabang dari algoritma evolusi yang merupakan metode adaptive yang digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Algoritma ini didasarkan pada proses genetika 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 akan bertahan (survive). Dengan meniru proses ini algoritma genetika dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam dunia nyata (Chen, C-L., Vempati, V.S. and Aljaber, N.: 1995)

Algoritma genetika menggunakan analogi secara langsung kebiasaan alami yaitu solusi alam. Algoritma ini bekerja dengan sebuah populasi yang terdiri dari individu-individu yang masing-masing individu merepresentasikan sebuah solusi yang mungkin bagi persoalan yang ada. Dalam kaitan ini individu dilambangkan dengan sebuah nilai fitness yang akan digunakan untuk mencari solusi terbaik dari persoalan yang ada (Ching-Fang, Liaw : 1999)

Pertahanan yang tinggi dari individu memberikan kesempatan untuk melakukan reproduksi melalui perkawinan silang dengan individu yang lain dalam populasi tersebut. Individu baru yang dihasilkan dalam hal ini dinamakan keturunan, yang membawa beberapa sifat dari induknya. Sedangkan individu dalam populasi yang tidak terseleksi dalam reproduksi akan mati dengan sendirinya. Dengan jalan ini, beberapa generasi dengan karakteristik yang bagus akan bermunculan dalam populasi tersebut, untuk kemudian dicampurkan dan ditukar dengan karakter yang lain. Dengan mengawinkan sebanyak mungkin individu, maka akan semakin banyak kemungkinan terbaik yang dapat diperoleh (Chen, C-L., Vempati, V.S. and Aljaber, N.: 1995)

Algoritma genetika merupakan teknik pencarian general yang didasari pada mekanisme seleksi dan genetika alami. Tidak sama halnya dengan algoritma pencarian lokal seperti tabu search yang didasari pada manipulasi satu feasible solution. Sedangkan algoritma genetika mempertahankan dan memanipulasi populasi feasible solution (kromosom). Kromosom ini terdiri dari gen-gen yang merupakan ciri atau karakter kromosom. Kromosom-kromosom ini dirubah atau dimodifikasi dengan menggunakan operator-operator genetika yang meniru prinsip evolusi dan secara turun-temurun sampai dimana populasi baru tercipta. Proses ini berulang-ulang sampai kondisi penghentian khusus sebelumnya terpenuhi (Lam F.S.C., Lin B.C., C. Sriskandarajah, and Yan H. : 1999).

Meskipun algoritma genetika telah terbukti menjadi sebuah teknik pencarian yang efektif dan versatile untuk memecahkan masalah optimasi, tetapi masih ada beberapa situasi dimana algoritma genetika sederhana tidak bekerja secara baik. Oleh karena itu, strategi penggabungan disarankan, yaitu dengan memasukkan konvensional heuristic sebagai prosedur perbaikan lokal ke dalam basic algoritma genetika (Lam F.S.C., Lin B.C., C. Sriskandarajah, and Yan H. : 1999). Salah satu bentuk yang biasa digunakan adalah hybrid genetic algorithm yaitu dengan memasukkan sebuah prosedur perbaikan lokal sebagai ekstra tambahan dalam pengulangan basic algoritma genetika pada rekombinasi dan seleksi. Prosedur perbaikan lokal diaplikasikan pada setiap offspring yang baru dihasilkan ke dalam sebuah lokal optimum sebelum menyisipkannya ke dalam populasi. Algoritma genetika digunakan untuk melakukan eksplorasi global diantara populasi, sementara prosedur perbaikan lokal digunakan untuk melakukan eksploitasi lokal disekitar kromosom (Chen, C-L., Vempati, V.S. and Aljaber, N.: 1995).

Persoalan penting dalam menerapkan algoritma genetika adalah merancang sebuah representasi kromosom yang tepat pada pemecahan masalah bersama dengan operator genetika agar semua kromosom yang dihasilkan selama proses evolusioner akan menghasilkan feasible solution. Dalam studi ini, diusulkan sebuah operasi yang didasari representasi kromosom untuk masalah penjadwalan flow shop. Representasi ini mengkodekan sebuah penjadwalan sebagai rangkaian operasi yang telah diurutkan, dimana setiap gen menunjukkan satu operasi. Dalam representasi ini, operasi didaftar dalam Penjadwalan Flow Shop Menggunakan Hybrid Genetic Algorithm (Hga) (Rachmad Hidayat)

urutan yang telah dijadwal. Diberikan sebuah kromosom, jadwal aktual disimpulkan sebagai berikut; Operasi pertama dalam kromosom dijadwal pertama, dan kemudian operasi kedua dalam kromosom dipertimbangkan, dan seterusnya. Setiap operasi di bawah pertimbangan dialokasikan ke posisi paling awal yang tersedia pada mesin yang sesuai dengan yang diperlukan pada waktu operasi. Proses ini diulang sampai semua operasi dalam kromosom terjadwal. Ini dapat dilihat dengan mudah, jadwal aktual yang dihasilkan dalam cara ini dijamin menjadi sebuah jadwal aktif, yaitu tidak ada operasi yang dapat diproses lebih awal tanpa menunda operasi lain (Chen, C-L., Vempati, V.S. and Aljaber, N.: 1995)

Inisial populasi dapat dihasilkan secara random. Pembibitan inisial populasi dengan solusi kualitas tinggi dapat membantu algoritma HGA menemukan solusi yang lebih baik secara cepat. Pada proses evolusi algoritma genetika, keragaman populasi dan tekanan seleksi (selective pressure) memegang peranan yang penting. Keduanya sangat berkaitan erat. Meningkatnya tekanan seleksi akan berakibat pada minimnya keragaman populasi. Sebaliknya, tekanan seleksi yang terlalu longgar membuat proses pencarian menjadi tidak efisien. Seleksi merupakan proses yang bertanggung jawab atas pemilihan kromosom dalam proses reproduksi.

Operator tukar silang pada dua parent solusi menghasilkan offspring solusi dengan rekombinasi kedua karakter parent solusi tersebut. Jika karakter baik dari solusi yang berbeda dikombinasikan secara tepat, solusi offspring yang dihasilkan mungkin mempunyai karakter yang lebih baik.

Mutasi mengoperasikan satu parent solusi dan menghasilkan sebuah solusi offspring dengan memodifikasi secara random karakter solusi parent. Ini membantu memelihara level dari populasi yang bermacam-macam, dan menyediakan sebuah mekanime untuk keluar dari optima lokal.

Tukar silang beroperasi pada dua kromosom dan menghasilkan offspring dari kombinasi gen-gen dari parents. Tingkat tukar silang yang tinggi memungkinkan eksplorasi daerah solusi dan mengurangi resiko pencarian pada lokal optima. Namun, hal ini juga memungkinkan eksplorasi daerah solusi yang terlalu luas. Tingkat tukar silang (dinyatakan dengan P_c) didefinisikan sebagai rasio antara jumlah offspring pada suatu generasi dan jumlah populasi (pop_size). Tingkat mutasi (P_m) yang rendah dapat mengakibatkan gen-gen yang berpotensi tidak dicoba. Sebaliknya, tingkat mutasi yang tinggi dapat menyebabkan offspring kehilangan kemiripan dengan parentnya. Salah satu proses mutasi adalah melakukan inversi terhadap nilai gen (Lam F.S.C., Lin B.C., C. Sriskandarajah, and Yan H. : 1999).

4. METODE STUDI

4.1 Teknik pengumpulan data

Dalam studi ini, pengumpulan data dilakukan dengan mengamati seluruh operasi yang diproses pada setiap mesin untuk kemudian diolah dan digunakan untuk menentukan parameter dari model penjadwalan flow shop. Selain itu, jadwal riil flow shop digunakan sebagai data untuk menguji dan mengevaluasi model dan sistem penjadwalan flow shop yang telah dikembangkan. Data studi terdiri dari:

1. Jumlah job (job)
2. Jumlah mesin (buah)
3. Operasi setiap job (kali)
4. Operasi yang harus diproses pada setiap mesin (kali)
5. Banyaknya operasi yang dapat diproses pada setiap mesin (kali)
6. Banyaknya operasi yang dapat diproses dari setiap job (kali)
7. Waktu yang diperlukan setiap operasi pada setiap mesin (menit)
8. Waktu yang diperlukan untuk menyelesaikan keseluruhan operasi (menit).

4.2 Pembuatan model dan perancangan program HGA

Tahapan pembuatan model dimulai dari formulasi model penjadwalan flow shop yang dapat mengurangi waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir. Ada sejumlah n job yang harus diproses pada sejumlah m mesin. Setiap job terdiri dari l operasi, dan setiap operasi diproses pada mesin yang berbeda selama periode waktu tertentu. Operasi-operasi setiap job diproses dalam beberapa urutan. Paling banyak satu operasi dapat diproses pada setiap mesin, dan paling banyak satu operasi dari setiap job dapat diproses. Pada tahap perancangan hybrid genetic algorithm, prosedur perbaikan lokal dimasukkan ke dalam basic algoritma genetika. Algoritma genetika digunakan untuk melakukan eksplorasi global diantara populasi, sementara prosedur perbaikan lokal digunakan untuk melakukan eksploitasi lokal disekitar kromosom.

Tahap ini melakukan perancangan dan pembuatan perangkat lunak, yang meliputi kebutuhan sistem, deskripsi sistem, perancangan obyek dan perancangan tampilan. Kebutuhan sistem menjelaskan bahasa pemrograman yang digunakan dan konfigurasi minimum yang diperlukan agar perangkat lunak dapat bekerja secara layak. Deskripsi sistem menjelaskan gambaran umum sistem yang dibuat. Perancangan obyek menjelaskan kelas-kelas yang dibuat, dan kolaborasi antar kelas. Perancangan tampilan menunjukkan tampilan-tampilan dalam perangkat lunak beserta penjelasannya.

4.3 Validasi Model dan Implementasi Software

Suatu pengembangan model harus melalui tahapan validasi. Validasi terhadap model penjadwalan flow shop ini dilakukan dengan membandingkan hasil yang diperoleh dari running program dengan hasil yang diperoleh dengan menggunakan metode Nawaz, Ensore, dan Ham (NEH) serta hasil perhitungan pada kondisi riil. Kriteria keberhasilan hybrid genetic algorithm adalah mampu menghasilkan jadwal yang lebih baik dari pada hasil perhitungan yang dibuat dengan menggunakan metode NEH dan hasil perhitungan pada kondisi riil.

Tahap ini mengimplementasikan hasil perancangan model dan sistem penjadwalan flow shop yang berupa software dengan beberapa iterasi sampai ditemukan solusi yang dapat mengurangi waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir.

5. HASIL STUDI DAN PEMBAHASAN

Perancangan model dilakukan dengan mempertimbangkan waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir. Hasil yang diperoleh kemudian dilanjutkan dengan pembuatan algoritma dan software penjadwalan flow shop.

Karena problem flow shop secara umum adalah sebuah penjadwalan permutasi, kita dapat memakai job sebagai representasi skema dari kromosom.

Contoh kromosom ke-28 adalah sebagai berikut:

$$V_k = [5 \ 10 \ 6 \ 2 \ 9 \ 8 \ 1 \ 4 \ 7 \ 3]$$

Ini berarti urutan job adalah $J_5, J_{10}, J_6, J_2, J_9, J_8, J_1, J_4, J_7, J_3$

Untuk mengevaluasi kromosom-kromosom menggunakan fungsi penilaian sebagai berikut:

$$\text{Eval}(V_k) = \frac{1}{C_{k \max}}$$

Variabel $C_{k \max}$ adalah makespan untuk kromosom ke k .

Ada banyak operasi tukar silang yang telah dikenal, antara lain PMX (Partially Mapped Crossover), OX (Order Crossover), CX (Cycle Crossover) dan Uniform. Disini peneliti menggunakan Partially Mapped Crossover (PMX) dan uniform.

Pada dasarnya PMX merupakan tukar silang dua titik yang dilanjutkan dengan proses untuk memperbaiki kromosom. Prosedur Partially Mapped Crossover (PMX) adalah sebagai berikut :

Step 1 : Pilih dua titik pada kromosom. Substring ini dinamakan daerah pemetaan.

Penjadwalan Flow Shop Menggunakan Hybrid Genetic Algorithm (Hga) (Rachmad Hidayat)

Step 2 : Pertukaran daerah pemetaan pada kedua parents untuk menghasilkan proto-children

Step 3 : Tentukan relasi pemetaan dari daerah pemetaan.

Step 4 : Perbaiki offsprings dengan relasi pemetaan.

Prosedur PMX dilustrasikan pada gambar 1 sebagai berikut :

Step 1 : Pilih dua titik potong secara random

Parent 1

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

Parent 2

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

Step 2 : Pertukarkan daerah pemetaan

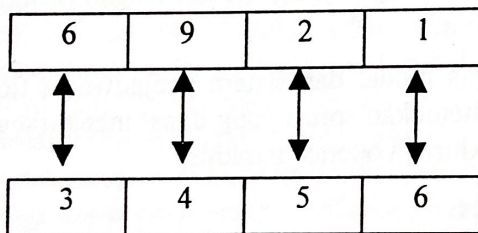
Proto-child 1

1	2	6	9	2	1	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Proto-child 2

5	4	3	4	5	6	7	8	3	10
---	---	---	---	---	---	---	---	---	----

Step 3 : Tentukan relasi pemetaan



1 ↔ 6 ↔ 3

2 ↔ 5

9 ↔ 4

Step 4 : Perbaiki offspring

Offspring 1

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

Offspring 2

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

Gambar 1 Operasi Tukar Silang

Pada Proto-child 1, terdapat duplikasi pekerjaan 1, 2, dan 9, sementara pekerjaan 3, 4, 5 hilang. Sesuai dengan relasi pemetaan pada step 3 maka pekerjaan 1, 2, dan 9 digantikan dengan pekerjaan 3, 4, dan 5. Sedangkan Uniform merupakan tukar silang dua titik yang dilakukan dengan selang seling, kemudian dilanjutkan dengan proses untuk memperbaiki kromosom.

Mutasi dirancang untuk menampilkan Random Exchange, yaitu menyeleksi dua gen secara random di dalam sebuah kromosom dan menukar posisi mereka. Contoh seperti gambar 2 berikut :

Memilih 2 gen

Parent

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

Offspring

1	2	7	4	5	6	3	8	9
---	---	---	---	---	---	---	---	---

Menukar posisi mereka

Gambar 2 Operasi Mutasi

5.1 Seleksi

Seleksi yang digunakan merupakan sampling stokastik dengan daerah sampling yang tetap. Sampling stokastik diimplementasikan roulette wheel. Metode ini sangat sering digunakan, dan merupakan metode seleksi yang cukup baik. Berikut ini adalah penjelasannya :

Prosedur Pra-seleksi

Step 1 : Hitung nilai fungsi untuk masing-masing kromosom.

$$\text{Eval}(V_k) = f(x), k = 1, 2, \dots, \text{pop_size}$$

Step 2 : Hitung total nilai fungsi $F = \sum_{k=1}^{\text{pop_size}} \text{eval}(V_k)$

Step 3 : Hitung probabilitas seleksi P_k untuk setiap kromosom. $P_k = \frac{\text{eval}(V_k)}{F}$, $k = 1, 2, \dots, \text{pop_size}$

Step 4 : Hitung probabilitas kumulatif q_k untuk masing kromosom., $k = 1, 2, \dots, \text{pop_size}$

Prosedur Seleksi

Step 1 : Bangkitkan bilangan random r antara 0 dan 1.

Step 2 : Jika $r \leq q_1$, maka pilih kromosom V_1 . Jika tidak, pilih kromosom ke k (V_k) sedemikian rupa sehingga $q_{k-1} < r \leq q_k$.

5.2 Pengembangan model dan algoritma Hybrid Genetic Algorithm

Model penjadwalan job pada problem flow shop ini menggunakan fungsi tujuan makespan. Studi ini memanfaatkan kelebihan algoritma genetika dan tabu search dengan mempertimbangkan waktu dari mulainya operasi pertama sampai berakhirnya operasi terakhir (makespan). Model matematis dari penjadwalan flow shop ini dapat ditulis sebagai berikut:

$$\text{Minimasi } F_1 = \text{Makespan} = C_{\max} = \max \{C_{mk}\}$$

Dengan fungsi pembatas:

$$C_{il} = \sum_{i=1}^m P_{il} \quad \text{untuk } i=2, \dots, m$$

$$C_{ik} = \sum_{k=1}^n P_{ik}$$

untuk $k = 2, \dots, n$

Jika $P_{ik} \neq 0$, $C_{ik} = \max \{C_{i,j,k}; C_{i,k-1}\}$

Jika $P_{ik} = 0$, $C_{ik} = C_{i-1,k}$

} untuk $i = 2, \dots, m$; $k = 2, \dots, n$

Dimana:

i = indeks mesin

k = indeks job

C_{ik} = saat pengerjaan job ke- k pada mesin i

P_{ik} = waktu pengerjaan job ke- k pada mesin i

Tahapan-tahapan dalam hybrid genetic algorithm adalah sebagai berikut:

- Step 1. Inisialisasi (Initialization) : Membangun sebuah inisial populasi dari pop_size solutions.
- Step 2. Perbaikan (Improvement) : Mempergunakan prosedur perbaikan lokal untuk menggantikan setiap solusi dengan satu optimum lokal.
- Step 3. Rekombinasi (Recombination) : Menggabungkan kembali solusi dalam populasi sekarang dengan menggunakan operator genetika, tukar silang dan mutasi, untuk menghasilkan offspring.
- Step 4. Perbaikan (Improvement) : Mempergunakan prosedur perbaikan lokal untuk menggantikan setiap offspring dengan sebuah optimum lokal.
- Step 5. Seleksi (Selection) : Menyeleksi solusi pop_size dari solusi dalam populasi sekarang dan solusi baru yang dihasilkan pada step 4 untuk membentuk generasi baru.
- Step 6. Iterasi (Iteration) : Ulangi step 3 sampai dengan 5 sampai solusi optimum atau jumlah maksimum generasi, max_gen, dicapai.

5.3 Perhitungan dengan metode NEH.

Metode NEH mempunyai asumsi bahwa job dengan waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir yang lebih besar, mempunyai prioritas yang lebih tinggi.

Algoritma NEH

Step 1 : Urutkan mulai dari yang terbesar sampai yang terkecil job berdasarkan waktu total operasi pada semua mesin.

Step 2 : Ambil job pertama dan atur sehingga makespannya minimum

Step 3 : For $k = 3$ to n , masukkan job k diantara posisi 1 sampai k , sehingga makespannya minimum.

Berdasarkan urutan job yang mempunyai waktu total operasi terbesar, maka urutan jobnya adalah J10 (719.8 menit) – J9 (622.3 menit) – J8 (527.4 menit) – J5 (479.9 menit) – J4 (479.9 menit) – J6 (428.1 menit) – J7 (406.2 menit) – J3 (387.7 menit) – J2 (328.9 menit) – J1 (277.8 menit).

Makespannya sebagai berikut :

Perhitungan Makespan :

Jika $C(j_i, k)$ menyatakan waktu penyelesaian pekerjaan j_i pada mesin k_i dan $\{j_1, j_2, j_3, \dots, j_n\}$ adalah urutan pekerjaan, maka waktu penyelesaian untuk n job dan m mesin adalah sebagai berikut :

$$C(j_1, 1) = t_{j_1 1}$$

$$C(j_1, k) = C(j_1, k-1) + t_{j_1 k}, \quad k=2,3,\dots,m$$

$$C(j_i, 1) = C(j_{i-1}, 1) + t_{ji1}, \quad i=2,3,\dots,n$$

$$C(j_i, k) = \max(C(j_{i-1}, k), C(j_i, k-1)), \quad i=2,3,\dots,n; k=2,3,\dots,m$$

Makespan dihitung sebagai berikut : $C_{\max} = (j_n, m)$.

5.4 Perhitungan dengan Hybrid Genetic Algorithm

Untuk menguji hasil perhitungan perangkat lunak dilakukan beberapa kali running menggunakan data yang sama dengan data perhitungan metode NEH. Parameter yang digunakan bervariasi untuk melihat pengaruhnya terhadap pencapaian solusi optimal.

Proses perhitungan :

- Jumlah pekerjaan : 10
- Jumlah mesin : 27
- Jumlah populasi per generasi : 85
- Jumlah maksimum generasi : 85
- Metode crossover : Partially Mapped Crossover
- Presentasi mutasi : 0.5
- Iterasi maksimum pada tabu search : 10

Tabel 1 Proses Perhitungan Makespan Dengan HGA

Presentasi Crossover	Kromosom Terbaik	Makespan
0.1	5-10-1-6-9-8-4-2-3-7	865.5
0.2	5-10-6-2-1-9-8-4-3-7	865.5
0.3	4-10-6-2-1-9-8-5-3-7	865.5
0.4	5-10-2-1-6-9-8-4-3-7	865.5
0.5	5-10-2-6-1-9-8-4-3-7	865.5
0.6	10-6-1-9-8-5-4-7-2-3	862.4
0.7	5-10-2-1-6-9-8-4-3-7	865.5
0.8	5-10-1-2-6-9-8-4-3-7	865.5
0.9	5-10-1-6-9-8-2-4-3-7	865.5
1.0	10-1-6-9-4-8-5-7-2-3	862.4

5.5. Analisis keberhasilan HGA

Dengan melihat hasil pengujian maka dapat disimpulkan bahwa secara konseptual, perangkat lunak berhasil memenuhi fungsinya sebagai alat untuk memecahkan masalah penjadwalan produksi flow shop. Pada perhitungan dengan menggunakan metode NEH diperoleh hasil makespan 883.5 menit, dan dari hasil running program diperoleh hasil makespan 862.4 menit. Sedangkan hasil perhitungan kondisi riil makespan 1022.1 menit.

Secara umum perubahan parameter akan berpengaruh terhadap kecepatan pencapaian solusi optimal. Bila ditinjau dari masing-masing parameter akan dapat diuraikan seperti analisa berikut :

1. Pengaruh terhadap ukuran populasi

Ukuran populasi menunjukkan jumlah kromosom yang terbentuk pada setiap generasi. Semakin banyak jumlah kromosom maka calon solusi yang dianalisa pada setiap generasi akan semakin banyak. Artinya probabilitas untuk menemukan solusi yang lebih baik akan semakin besar.

2. Pengaruh terhadap jumlah generasi
Jumlah generasi menunjukkan perubahan (iterasi) yang diinginkan agar populasi awal berkembang menghasilkan solusi yang lebih baik. Jumlah generasi yang besar sangat bermanfaat untuk mengarahkan populasi ke arah solusi yang lebih baik.
3. Pengaruh terhadap probabilitas crossover
Probabilitas crossover yang tinggi akan memberikan kesempatan kepada perangkat lunak untuk mengeksplorasi lebih banyak kromosom yang ada dalam populasi. Namun jika probabilitas terlalu besar, akan mengakibatkan terjadinya perkembangan solusi optimum yang prematur. Artinya perkembangan solusi optimum terjadi terlalu dini, karena pada crossover dihasilkan perubahan kromosom yang sama, maka populasi akan segera terisi oleh kromosom-kromosom yang homogen sehingga operasi-operasi genetika berikutnya tidak memberikan peningkatan yang berarti pada fitness.
4. Pengaruh terhadap probabilitas mutasi
Prosedur mutasi memberikan pengaruh yang lebih besar terhadap lahirnya kromosom anak yang berbeda dari kromosom induknya. Jika probabilitas mutasi terlalu besar maka gen yang ditukarkan semakin banyak dan dapat menimbulkan atau merusak kestabilan kromosom yang kurang memberikan nilai tambah kepada peningkatan fitness. Tetapi jika nilai probabilitas permutasi terlalu kecil maka pertukaran gen yang dievaluasi terlalu sedikit sehingga banyak gen yang sebenarnya memiliki potensi untuk memberikan peningkatan fitness tetapi tidak memperoleh kesempatan penukaran gen.

6. KESIMPULAN

Dari hasil perhitungan kondisi riil, metode Nawaz, Enscore, dan Ham (NEH), dan pendekatan hybrid genetic algorithm (HGA) pada analisa dan pembahasan dapat diambil kesimpulan bahwa :

1. Hasil perhitungan kondisi riil diperoleh waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir (makespan) sebesar 1022.1 menit, dan urutan operasi dari setiap job adalah job₁-job₂-job₃-job₄-job₅-job₆-job₇-job₈-job₉-job₁₀ (J1-J2-J3-J4-J5-J6-J7-J8-J9-J10).
2. Hasil perhitungan metode Nawaz, Enscore, dan Ham (NEH) diperoleh waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir (makespan) sebesar 883.5 menit, dan urutan operasi dari setiap job adalah job₁₀-job₉-job₈-job₅-job₄-job₆-job₇-job₃-job₂-job₁ (J10-J9-J8-J5-J4-J6-J7-J3-J2-J1).
3. Hasil perhitungan pendekatan hybrid genetic algorithm (HGA) diperoleh waktu total operasi dari mulainya operasi pertama sampai berakhirnya operasi terakhir (makespan) sebesar 862.4 menit, dan urutan operasi dari setiap job adalah job₁₀-job₆-job₁-job₉-job₈-job₅-job₄-job₇-job₂-job₃ (J10-J6-J1-J9-J8-J5-J4-J7-J2-J3), job₁₀-job₁-job₆-job₉-job₄-job₈-job₅-job₇-job₂-job₃ (J10-J1-J6-J9-J4-J8-J5-J7-J2-J3), job₁₀-job₆-job₁-job₉-job₄-job₈-job₅-job₇-job₂-job₃ (J10-J6-J1-J9-J4-J8-J5-J7-J2-J3), job₁₀-job₁-job₆-job₉-job₈-job₅-job₄-job₇-job₂-job₃ (J10-J1-J6-J9-J8-J5-J4-J7-J2-J3), job₁₀-job₁-job₆-job₉-job₅-job₈-job₄-job₇-job₂-job₃ (J10-J1-J6-J9-J5-J8-J4-J7-J2-J3), job₁₀-job₁-job₆-job₉-job₈-job₅-job₄-job₇-job₂-job₃ (J10-J1-J6-J9-J8-J5-J4-J7-J2-J3).
4. Hybrid Genetic Algorithm (HGA) mampu memberikan hasil penjadwalan produksi flow shop yang lebih baik dibandingkan dengan hasil perhitungan metode Nawaz, Enscore, dan Ham (NEH) dan perhitungan kondisi riil.
5. Hybrid Genetic Algorithm (HGA) mampu mengeksplorasi dan mengeksploitasi serta mengevaluasi banyak titik solusi secara serentak.

7. DAFTAR PUSTAKA

1. Baker, Kenneth R., 1974, *Introduction to Sequencing and Scheduling*, John Wiley and Sons, USA.

2. Conway, R.W., Maxwell, W.L. and Miller, L.W., 1967, *Theory of Scheduling*, (Reading, MA: Addison-Wesley).
3. Chen, C-L., Vempati, V.S. and Aljaber, N., 1995, An application of genetic algorithms for flowshop problems, *European Journal of Operational Research*, 80, 389-396.
4. Ching-Fang, Liaw, 1999, A hybrid genetic algorithm for the open shop scheduling problem, *European Journal of Operational Research*, 124, 28-42.
5. Goldberg, David E., 1989, *Genetic Algorithms: In Search, Optimization and Machine Learning*, The University of Alabama.
6. Dafis, Gordon B. And Margaretha K. Olson, *Management Information System: Conceptual Foundation, Structure and Development*, McGraw Hill International, Second Edition
7. Krober, Donald W. And Hugh J. Watson, *Computer Based Information Systems: A Management Approach*, MacMillan Publishing Company, 1990.
8. Lam F.S.C., Lin B.C., C. Sriskandarajah, and Yan H., 1999, Scheduling to minimize product design time using a genetic algorithm, *European Journal of Operational Research*, vol.37, no.6, hal.1369-1386
9. Mohanti, R.P. and Deskmukh, *An AHP Application For Manpower Evaluation*, Industrial Engineering Journal, 1996.
10. Morton, Thomas E., Pentico, David W., 1993, *Heuristic Scheduling Systems: with Applications to Production Systems and Project Management*, John Wiley and Sons, USA.
11. Pinedo, Michael., 1995, *Scheduling: Theory, Algorithms and System*, Prentice Hall, New Jersey.
12. Ralph H. Sprague, Jr and Hugh J. Watson, *Decision Support System Putting Theory and Control, A case Study Approach*, Prentice Hall Inc., 1982
13. Sitansu S. Mitra, *Decision Support System: Tools and Techniques*, John Wiley & Son Inc, 1986.
14. Turban, Efraim, *Decision support and Expert System : Management Support System* . (Englewood Cliffs, N.J. Prentice-Hall, Inc. 1995.