

PENJADWALAN PERKULIAHAN DENGAN PENDEKATAN *EVOLUTIONARY ALGORITHM* (STUDI KASUS: SISTEM INFORMASI AKADEMIK (SIKAD) PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER UNIVERSITAS BRWIJAYA)

Satrio Agung Wicaksono¹, R. Arief Setiyawan¹, Budi Darma Setiyawan¹,
Ari Hernawan¹, Rizal Setya Perdana¹

¹Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya
Email: ¹satrio@ub.ac.id, ¹rarief@ub.ac.id, ¹s.budidarma@ub.ac.id, -, ¹rizalespe@ub.ac.id

(Naskah masuk: 11 Juni 2014, diterima untuk diterbitkan: 22 Juli 2014)

Abstrak

Untuk menyusun jadwal kuliah bukanlah sesuatu yang mudah karena terkait aturan-aturan yang ada. Penjadwalan perkuliahan jika dilakukan dengan cara manual tentu saja akan memakan waktu cukup lama. Oleh karena itu pada penelitian ini mencoba untuk melakukan pendekatan menggunakan *evolutionary algorithm* untuk mempermudah dalam pembuatan jadwal kuliah dengan menerapkan aturan yang berlaku. Kromosom disusun dalam bentuk representasi *string* dengan susunan yang mewakili hari, jam perkuliahan, ruang dan gedung. Dari beberapa percobaan parameter yang digunakan, diperoleh hasil optimal pada jumlah individu 100 dan peluang crossover sebesar 75%.

Kata kunci: *algoritma evolusi, algoritma genetik, penjadwalan mata kuliah.*

Abstract

It is not an easy task to arrange academic schedule because it is affected by many constraints. If this scheduling is done manually, it will consume many times. Therefore, this research tries to use the evolutionary algorithm approach to do scheduling by applying the applicable rules. Chromosomes are represented as string, which each of them consist of days, times, rooms, and the buildings. From some experiments which are used in this research, optimal result obtained when use 100 individu in one population and 75% chance of crossover.

Keywords: *evolution algorithm, genetic algorithm, class scheduling.*

1. PENDAHULUAN

Dalam pembuatan jadwal kuliah ada sejumlah aturan-aturan yang harus dipertimbangkan ketika jadwal tersebut akan dibuat. Pertimbangan tersebut antara lain jadwal kuliah diharapkan bisa sesuai dengan ketersediaan waktu dosen untuk bisa mengajar, penggunaan ruang kuliah tidak boleh bersinggungan dengan jadwal yang lain, kapasitas perkuliahan harus sesuai dengan kapasitas ruang kuliah, tidak boleh ada jadwal kuliah yang beririsan dengan jadwal angkatan sebelumnya maupun sesudahnya, sehingga mahasiswa dapat mengambil mata kuliah angkatan sebelumnya maupun sesudahnya. Aturan-aturan tersebut yang menyebabkan pembuatan jadwal menjadi rumit atau memakan waktu cukup lama ketika proses pembuatan jadwal.

Pada penelitian skripsi oleh Mawaddah, 2006, penjadwalan dengan algoritma genetik di lingkungan brawijaya sudah diteliti, namun masih memiliki keterbatasan, misalnya pada penelitian tersebut dibatasi bahwa setiap mata kuliah memiliki

jumlah sks yang sama, dan hanya bias dilaksanakan satu minggu sekali. Batasan lainnya adalah, pada penelitian tersebut, ruangan masih belum diperhatikan ketika merepresentasikan kromosom.

Penelitian lainnya Fatimaa dan Hosni tahun 2011, menjelaskan tentang penggunaan algoritma genetik pada penjadwalan dengan memfokuskan pada teknik representasi genetik, proses mutasi dan crossover.

Pada penelitian ini dilakukan pendekatan dengan menggunakan *evolutionary algorithm* untuk menyusun jadwal perkuliahan dengan harapan dapat membantu dalam pembuatan jadwal perkuliahan dengan cepat dan tetap menerapkan aturan-aturan dalam pembuatan jadwal perkuliahan yang lebih kompleks.

Yang ingin dilakukan dalam penelitian ini adalah bagaimana merepresentasikan kromosom untuk penjadwalan mata kuliah, bagaimana menerapkan aturan-aturan pembuatan jadwal mata kuliah pada *evolutionary algorithm*, dan bagaimana menentukan nilai penalti bagi jadwal yang melanggar aturan.

2. ALGORITMA EVOLUSI

Algoritma evolusi adalah istilah yang diberikan pada *evolutionary computing* yang mengkhusus pada algoritma. Algoritma evolusi memiliki beberapa subbidang, yaitu: programing evolusi (*evolutionary programming*), strategi evolusi (*evolution strategies*), algoritma genetik (*genetic algorithm*), dan programing genetik (*genetic programming*). Pada penelitian ini, solusi yang diberikan akan difokuskan pada algoritma genetik.

Algoritma genetika adalah algoritma pencarian heuristik yang didasarkan atas mekanisme seleksi alami dan genetika alami. Algoritma genetika pertama kali dikembangkan pada tahun 1975 oleh John Holland bersama rekan kerja dan murid-muridnya dari Universitas Michigan (Gen, 1997).

Secara umum, dalam algoritma genetik, akan dilakukan perulangan sejumlah n buah generasi, dalam setiap generasi, langkah-langkah yang dilakukan adalah:

1. Menghitung nilai fitness
2. Seleksi calon induk
3. Perkawinan silang
4. Mutasi
5. Seleksi untuk masuk ke generasi selanjutnya dengan mencampur anak dan induk, kemudian di ranking.

2.1 KROMOSOM

Langkah pertama pada algoritma evolusi adalah menerjemahkan/ merepresentasikan masalah riil menjadi terminologi biologi dalam bentuk kromosom. Kromosom menyatakan salah satu solusi yang mungkin. Kromosom merupakan kumpulan gen (Setiadi, 2001).

Kromosom disusun dalam bentuk representasi *string* dengan susunan yang mewakili hari, jam perkuliahan, ruang dan gedung. Pada tabel 1 menunjukkan ilustrasi penyusunan kromosom, dimana kromosom disusun dengan *header* secara

berurut yaitu kode mata kuliah, tahun mata kuliah, kelas, kode dosen (dosen ketua), dan SKS(Satuan Kredit Semester) pelaksanaan.

Dari gabungan [(hari-ke, jam-ke); (ruang, gedung)] maka diperoleh kromosom sebagai berikut:
 [(1,1);(A2.21,TIF1)] [(1,2);(A2.21,TIF1)]
 [(4,3);(A2.24,TIF1)] [(4,4);(A2.24,TIF1)]
 ...dst.

2.2 FITNESS

Nilai *fitness* merupakan suatu ukuran baik tidaknya suatu solusi yang dinyatakan sebagai satu individu, atau dengan kata lain nilai *fitness* menyatakan nilai dari fungsi tujuan. Nilai *fitness* ini yang dijadikan sebagai acuan dalam mencapai nilai optimal dalam algoritma genetika.

Fitness dihitung pada setiap individu yang terbentuk, dari setiap individu akan diambil sejumlah individu yang terbaik untuk bertahan pada generasi berikutnya. Penghitungan dilakukan dengan memberikan pinalti untuk setiap aturan yang digunakan dalam penjadwalan. Semakin wajib aturan dilaksanakan, maka akan semakin besar nilai pinalti yang diberikan. Berikut aturan penghitungan fungsi *fitness*:

$$f(g) = 1 / (1 + pinalti) \quad (2.1)$$

dimana

$$pinalti = \sum P_i v_i(g) \quad (2.2)$$

dimana P_i adalah bobot pinalti yang diberikan untuk aturan ke- i , dan $v_i(g) = 1$ jika jadwal g melanggar aturan ke- i , dan bernilai 0 jika sebaliknya.

Dari penghitungan nilai *fitness* dapat diketahui bahwa semakin sedikit aturan yang dilanggar, maka akan semakin besar nilai *fitness*-nya. Jadwal yang sempurna akan memiliki nilai *fitness* 1, karena nilai total aturan yang dilanggar adalah 0.

Pada penjadwalan mata kuliah yang akan dibuat, diberikan sejumlah aturan beserta bobot pinalti dari setiap aturan yang terlihat pada tabel 2. Nilai bobot pinalti ini diberikan berdasarkan tingkat kepentingan dari masing-masing aturan.

Tabel 1 Struktur Kromosom

[illegible]

Tabel 2 Aturan dan nilai penalti

Aturan	Bobot Pinalti
Tabrakan mata kuliah satu semester	5
Ruang yang digunakan kurang dari kapasitas maksimum peserta kelas	5
Bentrok jadwal mengajar dosen	3
Ruang yang sama digunakan dalam waktu yang bersamaan	10
Mata kuliah yang seharusnya dilaksanakan pada satu kesatuan pertemuan tetapi dilaksanakan di hari yang berbeda atau pada jam yang tidak berurut	5

2.3 SELEKSI

Seleksi digunakan untuk memilih individu-individu mana saja yang akan dipilih untuk proses *crossover* dan mutasi. Seleksi digunakan untuk mendapatkan calon induk yang baik. "Induk yang baik akan menghasilkan keturunan yang baik". Semakin tinggi nilai *fitness* suatu individu semakin besar kemungkinannya untuk terpilih.

Langkah pertama yang dilakukan dalam seleksi ini adalah pencarian nilai *fitness*. Nilai *fitness* ini yang nantinya akan digunakan pada tahap-tahap seleksi berikutnya. Masing-masing individu dalam wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua individu dalam wadah seleksi tersebut.

2.4 PERKAWINAN SILANG

Apabila proses seleksi telah dilaksanakan dan sudah terpilih induk baru, maka operator berikutnya adalah perkawinan silang. Perkawinan silang adalah cara mengkombinasikan gen-gen induk untuk menghasilkan keturunan baru. Perkawinan silang yang digunakan adalah perkawinan silang satu titik (*one point crossover*). Pada perkawinan ini dilakukan dengan cara menukar nilai gen pada posisi gen yang sama dari kedua induk. Penukaran gen tersebut juga harus dilakukan pengecekan apakah individu baru yang terbentuk tidak ilegal atau tetap sesuai dengan aturan yang berlaku.

Pada proses perkawinan silang, kromosom anak yang dihasilkan merupakan kombinasi gen-gen yang dimiliki oleh kromosom induk. Secara umum, mekanisme kawin silang ini adalah sebagai berikut (Al-Mili, 2010):

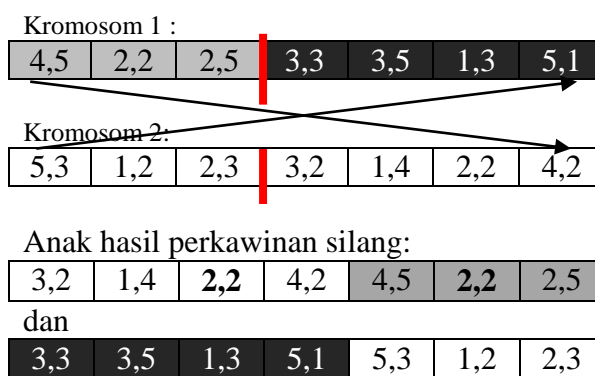
1. memilih dua buah kromosom sebagai induk.

2. memilih secara acak posisi dalam kromosom, biasa disebut titik perkawinan silang, sehingga masing-masing kromosom induk terpecah menjadi dua segmen.
3. lakukan pertukaran antar segmen kromosom induk untuk menghasilkan kromosom anak.

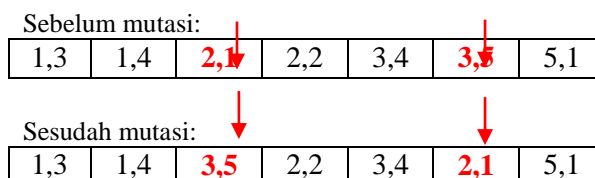
Ilustrasi proses perkawinan silang, bisa dilihat pada gambar 1.

2.5 MUTASI

Mutasi dilakukan untuk mencegah terjadinya konvergensi prematur. Mutasi dilakukan dengan cara menukar nilai gen dengan nilai gen lainnya atau dikenal dengan istilah *swap mutation*. Ilustrasinya seperti terlihat pada gambar 2.



Gambar 1 Ilustrasi perkawinan silang



Gambar 2 Ilustrasi mutasi cara swap

3. HASIL PENELITIAN

Dalam penerapan algoritma evolusi, ada beberapa parameter yang harus diperhatikan, seperti, jumlah individu yang digunakan, generasi, dan prosentase *crossover* dan mutasi yang digunakan. Dalam penelitian ini, akan dicari beberapa nilai parameter yang mengoptimalkan proses penyusunan jadwal kuliah.

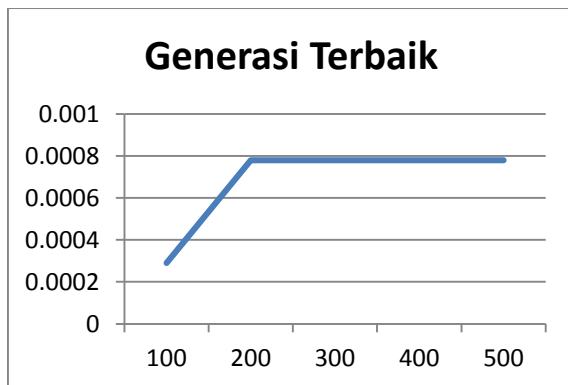
3.1 PENGARUH JUMLAH GENERASI PADA PROSES GENETIK

Pada proses ini dilakukan dengan tujuan untuk mendapatkan nilai generasi terbaik dalam proses genetika yaitu dengan mencari pada generasi ke berapa nilai *fitness* mencapai titik konvergen. Tabel 3 menunjukkan hasil percobaan dengan menggunakan 100 individu dan 50% *crossover* dengan berbagai variasi jumlah generasi.

Dari grafik pada gambar 3 diperoleh titik konvergen pada generasi ke 200. Sehingga pada percobaan selanjutnya jumlah generasi yang digunakan sebanyak 200.

Tabel 3 Hasil percobaan dengan variasi jumlah generasi

Individu	Generasi	Crossover(%)	Fitness Terbaik	Total Pinalti
100	100	50	0,00029	3490
100	200	50	0,00078	1280
100	300	50	0,00078	1280
100	400	50	0,00078	1280
100	500	50	0,00078	1280



Gambar 3 Generasi untuk titik konvergensi

3.2 PENGARUH JUMLAH INDIVIDU PADA PROSES GENETIK.

Dalam proses ini dilakukan beberapa percobaan dengan menggunakan jumlah individu yang berbeda. Dari proses ini, didapat hasil seperti pada tabel 4, dan grafiknya pada gambar 4.

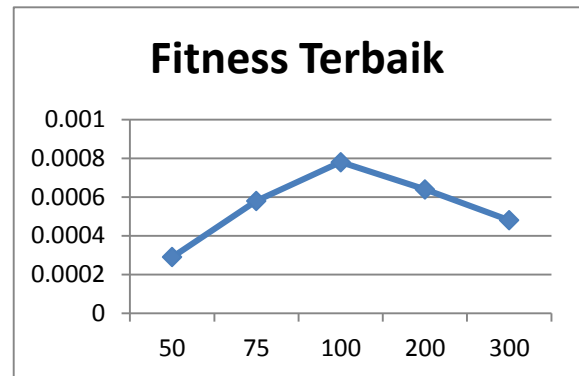
Percobaan ini menggunakan 200 generasi dengan nilai peluang crossover sebesar 50% dan mutasi sebesar 10%.

Dari grafik pada gambar 4 terlihat generasi dengan fitness terbaik diperoleh dengan menggunakan 100 individu. Dengan demikian percobaan selanjutnya menggunakan 100 individu, untuk menentukan berapa peluang crossover yang optimal.

Tabel 4 Hasil percobaan dengan variasi jumlah individu

Individu	Generasi	Crossover(%)	Fitness Terbaik	Pinalti
50	200	50	0,00029	3490
75	200	50	0.00058	1720
100	200	50	0,00078	1280
200	200	50	0.00064	1570

300	200	50	0.00048	2070
-----	-----	----	---------	------



Gambar 4 Generasi untuk titik konvergensi

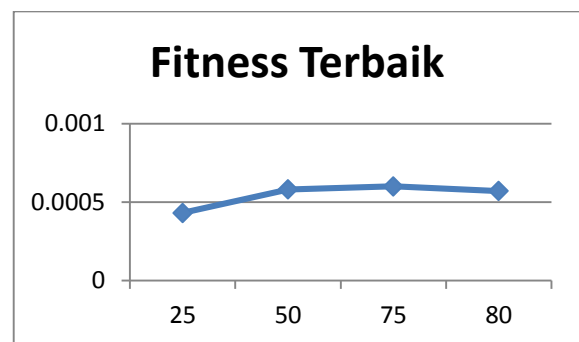
3.3 PENGARUH PELUANG CROSSOVER

Dari percobaan sebelumnya didapat nilai jumlah individu sebesar 100. Dan berikutnya diujikan berapa besar peluang crossover. Hasilnya dapat dilihat pada tabel 5 dan gambar 5.

Tabel 5 Hasil percobaan dengan variasi jumlah generasi

Individu	Generasi	Crossover(%)	Fitness Terbaik	Pinalti
100	200	25	0,00043	2340
100	200	50	0.00058	1720
100	200	75	0.00060	1680
100	200	80	0.00057	1740

Data lain yang digunakan adalah jumlah generasi sebesar 200, dan nilai mutasi sebesar 10%.



Gambar 5 Generasi untuk titik konvergensi

Hasil ini menunjukkan bahwa diperoleh nilai optimal ketika nilai peluang crossover yang dilakukan sebesar 75%.

4. KESIMPULAN

Dari implementasi dan percobaan yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut:

Komputasional II 16-17 Oktober 2001, Universitas Indonesia, Jakarta.

1. Kromosom direpresentasikan dengan menggunakan kombinasi hari, jam, ruangan, dan gedung kuliah.
2. Dari beberapa percobaan parameter yang digunakan, diperoleh hasil optimal pada jumlah individu 100 dan peluang crossover sebesar 75%.
3. Aturan atau batasan yang digunakan adalah meminimalkan
 - a. Tabrakan mata kuliah satu semester
 - b. Penggunaan ruang yang kurang dari kapasitas maksimum peserta kelas
 - c. Bentrok jadwal mengajar dosen
 - d. Ruang yang sama digunakan dalam waktu yang bersamaan
4. Nilai pinalti digunakan untuk mengurangi nilai fitness pada individu dengan jumlah pelanggaran batasan yang besar.

5. SARAN

Beberapa hal masih perlu diteliti, karena jadwal yang dihasilkan masih belum sempurna. Beberapa hal tersebut ialah:

1. Proses pembentukan populasi awal yang sebaiknya juga menggunakan batasan.
2. Menggunakan metode mutasi, crossover dan seleksi lain yang mungkin bisa meningkatkan performa penjadwalan.

6. DAFTAR PUSTAKA

AL-MILLI, N. R., 2010, *Hybrid Genetic Algorithms with Great Deluge for Course Timetabling*, IJCSNS International Journal of Computer Science and Network Security.

FATIMAA, S & HOSNY, M., 2011, *A Survey of Genetic Algorithms for the University Timetabling Problem*.

GEN, M. & RUNWEI, C., 1997, *Genetic Algorithms And Engineering Design*. John Wiley & Sons, Inc. New York.

MAWADDAH, N. K., 2006. *Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika*. Skripsi Ilmu Komputer Brawijaya. Malang.

SETIADI, R., 2001, *Pemecahan Masalah Penjadwalan Kuliah dengan Menggunakan Teknik Intelligent Search*. Seminar Nasional Kecerdasan