

ALGORITMA GENETIK TABU SEARCH DAN MEMETIKA PADA PERMASALAHAN PENJADWALAN KULIAH

Moh. Ali Albar¹⁾

¹⁾ Teknik Elektro Universitas Gadjah Mada
Jl. Grafika No. 2, Fakultas Teknik, UGM, Yogyakarta 55283
email : albar.elektrougm@gmail.com¹⁾

Abstrak

Penyampaian informasi penjadwalan kuliah di lembaga akademik merupakan hal yang sangat penting. Masalah penjadwalan kuliah merupakan permasalahan yang sangat kompleks karena adanya berbagai komponen yang terdiri dari mahasiswa, dosen, ruang kelas, dan waktu kegiatan perkuliahan dengan memperhatikan batasan dan syarat tertentu. Penjadwalan kuliah yang di susun secara manual sudah semakin jarang digunakan karena adanya aplikasi penjadwalan yang berbasis komputer dan bekerja secara otomatis. Aplikasi ini menggunakan berbagai metode pendekatan heuristic dalam menyelesaikan permasalahan penjadwalan. Ada beberapa metode heuristic yaitu Algoritma Memetika dan Algoritma Genetik Tabu Search dimana dengan dua metode tersebut pembuatan solusi jadwal yang optimal dapat dilakukan dengan waktu yang tidak lama walaupun menggunakan data yang banyak sekalipun.

Penelitian ini bertujuan untuk membandingkan kedua Algoritma tersebut dalam sistem penjadwalan kuliah yang menghasilkan solusi jadwal yang optimal. Variabel pengujianya adalah Soft Constraint, Hard Constraint, Total Soft Constraint, Total Hard Constraint dan Waktu Eksekusi. Berdasarkan hasil penelitian, Algoritma Genetik Tabu Search lebih unggul daripada Algoritma Memetika pada slot yang lebih rendah, sedangkan untuk slot yang lebih besar Algoritma Memetika lebih unggul daripada Algoritma Genetik Tabu Search.

Kata kunci :

Memetika, Genetik Tabu Search, Heuristic

1. Pendahuluan

1.1 Latar Belakang Permasalahan

Penyampaian informasi yang berkaitan dengan penjadwalan kuliah di lembaga akademik merupakan hal yang sangat penting. Masalah penjadwalan kuliah merupakan permasalahan yang sangat kompleks karena adanya berbagai komponen yang terdiri dari mahasiswa, dosen, ruang kelas, dan waktu kegiatan perkuliahan dengan memperhatikan batasan dan syarat tertentu.

Aplikasi ini menggunakan berbagai metode pendekatan heuristic dalam menyelesaikan permasalahan penjadwalan. Metode heuristic merupakan suatu teknik untuk penyelesaian permasalahan yang tidak menekankan pada pembuktian apakah solusi yang didapatkan adalah benar, tetapi lebih menekankan pada performa komputasi dan kesederhanaan. Metode

heuristic merupakan suatu metode penyelesaian yang menggunakan konsep pendekatan. Pendekatan heuristic menggunakan suatu algoritma yang secara interaktif akan menghasilkan solusi yang akan mendekati optimal. Pendekatan heuristic menghasilkan perhitungan yang cepat karena dilakukan dengan membatasi pencarian dengan mengurangi jumlah alternatif yang ada. Pendekatan heuristic lebih dapat diterapkan ke permasalahan nyata dimana permasalahan melibatkan jumlah input data yang besar dan kompleks. Beberapa algoritma heuristic telah dikembangkan antara lain Algoritma Genetik, Tabu Search, dan Algoritma Memetika. Belum adanya penelitian yang membahas perbandingan kinerja antara Algoritma Genetik Tabu Search dan Algoritma Memetika, maka di dalam penelitian ini penulis ingin menganalisa kinerja dari Algoritma Genetik Tabu Search dan Memetika khususnya pada permasalahan penjadwalan kuliah.

1.2 Tujuan Penelitian

Membandingkan sistem penjadwalan kuliah yang diimplementasikan menggunakan algoritma Genetik Tabu Search dengan algoritma Memetika sehingga dapat diketahui algoritma mana yang lebih unggul.

1.3 Batasan Penelitian

Penelitian ini di uji dan di analisa khusus menggunakan data masukan yang di dapat dari Fakultas Pertanian Universitas Mataram. Setiap matakuliah direpresentasikan sebesar 2 SKS. Variabel yang di analisa antara lain Soft Constraint, Hard Constraint, Total Soft Constraint, Total Hard Constraint dan Waktu Eksekusi yang dibutuhkan aplikasi tersebut hingga menghasilkan solusi yang optimal..

1.4 Hasil Penelitian Sebelumnya

Pada paper yang ditulis oleh Iing Mutakhiroh (2007) dengan menggunakan metode heuristic berupa algoritma genetik dan algoritma semut mengemukakan bahwa hasil yang diperoleh dari metode heuristic lebih variatif dibandingkan metode konvensional, dimana waktu perhitungan yang diperlukan lebih cepat 30% dibandingkan dengan menggunakan metode konvensional. Selain itu dikatakan bahwa untuk kasus yang berbeda, algoritma akan memberikan hasil yang berbeda, tidak dapat dipastikan bahwa algoritma semut atau genetik yang terbaik. [1]

Penelitian yang dilakukan oleh Muhammad Aria (2008) mengenai aplikasi algoritma genetik untuk optimasi penjadwalan mata kuliah menyatakan bahwa performansi akhir cukup baik dan menunjukkan bahwa

algoritma *genetik* telah berhasil diaplikasikan untuk penjadwalan mata kuliah. [2].

Penelitian yang dilakukan oleh Irfan Darmawan (2011) menyatakan bahwa metode *Incorporation Genetic-Tabu Search Algorithm* (IGTS) memiliki performansi yang relatif lebih baik terutama pada kasus *Scheduling Jobs on Multiple Machines* (SJMM) berukuran kecil, walaupun performansinya semakin menurun seiring dengan bertambahnya ukuran kasus. [9].

Penelitian lainnya yang dilakukan oleh Thamilselvan (2009) membandingkan penggunaan Algoritma *Genetik* (GA), *Tabu Search* (TS), dan kombinasi antara Algoritma *Genetik* dan *Tabu Search* pada permasalahan penjadwalan *job shop*. Kombinasi ini dikenal dengan sebutan Algoritma *Genetik Tabu Search* (GTS). Tujuan penelitiannya yaitu menemukan jadwal yang optimal dan dapat meminimalkan waktu penyelesaian keseluruhan mesin dari semua kegiatan yang ada. Hasil penelitian menunjukkan bahwa GTS lebih baik dalam memecahkan masalah penjadwalan daripada GA atau TS. [3].

Penelitian yang dilakukan oleh Fulya Altiparmak dan Berna Dengiz (2003) pada kasus mendisain secara optimal jaringan komputer pada suatu universitas. Penelitian ini membandingkan kinerja tiga prosedur *metaheuristic* klasik dan bentuk kombinasi antara ketiganya untuk memecahkan masalah secara realistik: *hill climbing* (HC), *simulated annealing* (SA) dan algoritma *genetik* (GA). Tiga perubahan yang menggunakan pencarian lokal untuk meningkatkan solusi selama iterasi juga dibandingkan. Hal ini menunjukkan bahwa melakukan pencarian lokal selama evolusi dari algoritma *genetik*, yang disebut algoritma *Memetika*, menghasilkan desain jaringan terbaik dan melakukannya dengan biaya yang wajar. Hasil penelitiannya yaitu perbandingan algoritma lokal antara HC, SA, dan GA menyatakan bahwa GA memiliki kinerja yang lebih baik dari HC dan SA untuk kasus yang semakin besar, sedangkan perbandingan kombinasi algoritma antara SA-HC, GA-HC, dan MA (*Memetic Algorithm*) menyatakan bahwa MA lebih baik dari SA-HC dan GA-HC dalam menggunakan pencarian lokal untuk solusi baru yang diciptakan selama evolusi dan merupakan strategi yang paling sukses. Ini memberikan tidak hanya desain terbaik untuk masalah yang lebih besar, tetapi juga upaya komputasi terendah dari *heuristic* global. [4]

Penelitian yang diusulkan berbeda dengan penelitian yang dilakukan oleh peneliti-peneliti di atas dalam hal kasus penyelesaian optimasi pada masalah penjadwalan kuliah yang bersifat lebih kompleks dan jenis algoritma yang diterapkan pada sistem penjadwalan kuliah. Pada penelitian ini menggunakan Algoritma *Genetik Tabu Search* (GTS) dan Algoritma *Memetika* (MA), dimana *setting* parameter untuk Algoritma *Genetik Tabu Search* yaitu model populasinya menggunakan *generational replacement* dengan *elitisme*, seleksi orang tua berupa metode algoritma *Tabu Search*, rekombinasinya menggunakan *uniform crossover*, proses mutasinya menggunakan *swap mutation*, sedangkan pada

Algoritma *Memetika* yaitu model populasinya menggunakan *generational replacement* dengan *elitisme*, seleksi orang tua berupa *tournament selection*, rekombinasinya menggunakan *uniform crossover*, proses mutasinya menggunakan *swap mutation*, dan ada penambahan proses yaitu pencarian lokal (*local search*). Penelitian ini akan menguji manakah antara Algoritma *Genetik Tabu Search* dengan Algoritma *Memetika* yang lebih unggul digunakan dalam permasalahan penjadwalan kuliah dilihat dari efisiensi pencarian solusinya dan batasan-batasan yang dilanggar dan belum terpenuhi, waktu eksekusi, serta mengetahui kelebihan dan kekurangan antara Algoritma *Genetik Tabu Search* dan Algoritma *Memetika*.

1.5 Kontribusi Makalah

Hasil penelitian tesis ini diharapkan dapat menjadi acuan dalam menentukan algoritma yang cocok untuk diaplikasikan ke sistem penjadwalan kuliah. Selain itu sebagai acuan bagi penelitian lainnya dalam melakukan perbandingan terhadap algoritma *heuristic* yang unggul lainnya.

2. Tinjauan Pustaka

Penelitian yang dilakukan oleh Iing Mutakhirah (2007) pada metode algoritma *genetik* menggunakan peluang *crossover* sebesar 0.500, peluang mutasi sebesar 0.100, peluang pelestarian sebesar 0.100, maksimum jumlah generasi sebanyak 50, panjang kromosom adalah 5, dan proses seleksi menggunakan metode Roda Roulette. [1]. Penelitian yang dilakukan oleh Muhammad Aria (2008) mengenai aplikasi algoritma *genetik* untuk optimasi penjadwalan mata kuliah menyatakan bahwa performansi akhir cukup baik dan menunjukkan bahwa algoritma *genetik* telah berhasil diaplikasikan untuk penjadwalan mata kuliah. Algoritma *genetik* yang diterapkan menggunakan metode seleksi berupa *good fitness*, pemilihan mutasi dapat dilakukan dengan metode *good fitness*, random maupun *roulette wheel*. [2]. Penelitian yang dilakukan oleh Irfan Darmawan (2011) menyatakan bahwa metode *Incorporation Genetic-Tabu Search Algorithm* (IGTS) yang merupakan penggabungan dari *tabu search* dan algoritma *genetik* memiliki performansi yang relatif lebih baik pada kasus *Scheduling Jobs on Multiple Machines* (SJMM). Kaidah *tabu search* digunakan sebagai kaidah dasar dalam algoritma, sedangkan kaidah algoritma *genetik* hanya terbatas penggunaannya pada proses pembangkitan sampel. Pada penelitian ini terdapat lima pekerjaan dan lima mesin dengan ukuran populasi sebanyak 10, jumlah iterasi untuk mutasi adalah 100, dan *crossover* adalah 200. Algoritma akan berakhir setelah pembentukan 200 generasi. [9].

Penelitian oleh Thamilselvan (2009) membandingkan penggunaan Algoritma *Genetik* (GA), *Tabu Search* (TS), dan kombinasi antara Algoritma *Genetik* dan *Tabu Search* (GTS) pada permasalahan penjadwalan *job shop*. Penelitian menguji besarnya waktu dan biaya hasil solusi dari ketiga algoritma dengan empat ukuran masalah (pekerjaan x mesin) yaitu 5x5, 5x10, 10x5, dan 5x10. Hasil penelitian menunjukkan bahwa GTS lebih

baik dalam memecahkan masalah penjadwalan daripada GA atau TS. [3].

Penelitian yang dilakukan oleh Fulya Altiparmak dan Berna Dengiz (2003) pada kasus mendisain secara optimal jaringan komputer pada suatu universitas. Penelitian ini membandingkan kinerja tiga prosedur metaheuristic klasik dan bentuk kombinasi antara ketiganya untuk memecahkan masalah secara realistis: hill climbing (HC), simulated annealing (SA) dan algoritma *genetik* (GA). Implementasi Algoritma *genetik* pada penelitian ini yaitu generasi dengan populasi awal di acak sebanyak 50 kromosom. Seleksi orang tua dilakukan menggunakan 2-tournament selection. Proses penyilangan menggunakan metode uniform crossover dengan probabilitas 0.60 dan terjadi 50 buah penyilangan dengan harapan ada 30 kromosom anak ($0,60 \times 50$) selama setiap generasi. Probabilitas mutasi sebesar 0.4. [4]

2.1 Algoritma Genetik

Teknik pencarian dilakukan atas sejumlah solusi yang mungkin yang dikenal (populasi). Individu yang terdapat dalam satu populasi disebut dengan istilah kromosom. Kromosom berupa suatu solusi yang masih berbentuk simbol. Populasi awal dibangun secara acak, sedangkan populasi berikutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan istilah generasi. Pada setiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang disebut dengan fungsi *fitness*. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut. Generasi berikutnya dikenal dengan istilah anak (*offspring*) terbentuk dari gabungan dua kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyilangan (*crossover*). Selain operator penyilangan, suatu kromosom dapat juga dimodifikasi dengan menggunakan operator mutasi. Populasi generasi yang baru dibentuk dengan cara menyeleksi nilai *fitness* dari kromosom induk (*parent*) dan nilai *fitness* dari kromosom anak (*offspring*), serta menolak kromosom-kromosom yang lainnya sehingga ukuran populasi (jumlah kromosom dalam suatu populasi) konstan. Setelah melalui beberapa generasi, maka algoritma ini akan konvergen ke kromosom terbaik. [5]

2.2 Algoritma Tabu Search

Tabu Search merupakan suatu metode optimasi yang menggunakan *short-term memory* untuk menjaga agar proses pencarian tidak terjebak pada nilai optimum local. Metode ini menggunakan *Tabu List* untuk menyimpan sekumpulan solusi yang baru saja di evaluasi. Selama proses optimasi, pada setiap iterasi, solusi yang akan di evaluasi akan dicocokkan terlebih dahulu dengan isi *Tabu List* untuk melihat apakah solusi tersebut sudah ada pada *Tabu List*. Apabila solusi tersebut sudah ada pada *Tabu List*, maka solusi tersebut tidak akan dievaluasi lagi pada iterasi berikutnya. Apabila sudah tidak ada lagi solusi yang tidak menjadi anggota *Tabu List*, maka nilai terbaik yang baru saja diperoleh merupakan solusi yang sebenarnya. [6]

2.3 Pencarian Lokal (*Local Search*)

Pencarian lokal merupakan pertukaran atau penempatan kembali informasi yang di bawa oleh gen-gen dalam satu kromosom anak dengan harapan dapat meningkatkan kualitas kromosom tersebut. Pencarian lokal dapat dilakukan dengan menukar dua gen atau permutasi beberapa gen tanpa mengurangi kualitas dari kromosom sebelumnya. Dalam kasus tertentu, pencarian lokal hanya dilakukan kepada gen yang terkena pinalti, artinya hanya informasi yang kurang tepat yang dipertukarkan antar gen dalam satu kromosom yang sama.

2.4 Algoritma Genetik Tabu Search

Algoritma *Genetik Tabu Search* mengkombinasikan dua pendekatan dengan tujuan untuk mengurangi kekurangan yang ada pada kedua algoritma. Pada permulaannya menciptakan satu solusi secara acak, dan untuk beberapa iterasi, hal ini mengoptimalkannya dengan menggunakan metode dasar *Tabu Search*, kemudian algoritma kombinasi ini menerapkan prinsip algoritma *genetik* untuk mencapai suatu solusi, dan menciptakan suatu generasi solusi baru. [7]

2.5 Algoritma Memetika

Algoritma *Memetika* (AM) adalah perluasan dari Algoritma *Genetik*. Kelebihan dari AG adalah pada cara kerjanya yang paralel. AG bekerja dalam ruang pencarian yang menggunakan banyak individu sekaligus, sehingga kemungkinan AG untuk terjebak pada ekstrim lokal lebih kecil dibandingkan metode lain. Kekurangan dari AG adalah dalam hal waktu komputasi karena harus melakukan evaluasi *fitness* pada semua solusi di setiap iterasinya, sehingga AG bisa lebih lambat dibandingkan dengan metode lain. [8]

Namun telah diketahui bahwa AG tidak cukup baik untuk mencari solusi yang sangat dekat dengan solusi optimal. Kekurangannya ini dapat diakomodasi dengan cara menambahkan metode *Local Search* pada AG. Hasil penggabungan ini dikenal dengan sebutan Algoritma *Memetika* (AM). AM adalah suatu metode pencarian *heuristic* yang memiliki karakteristik yang sama dengan AG dikombinasikan dengan metode *Local Search* yang secara bersama-sama dapat meningkatkan kualitas pencarian solusi (Moscato, 1989).

Pada AM, *Local Search* bertujuan untuk melakukan perbaikan lokal yang dapat diterapkan sebelum dan atau sesudah proses seleksi, *crossover*, dan mutasi. *Local Search* juga dapat berguna untuk mengontrol besarnya ruang pencarian solusi. AM dapat memberikan hasil yang lebih baik daripada AG, namun memerlukan waktu komputasi yang lebih lama.

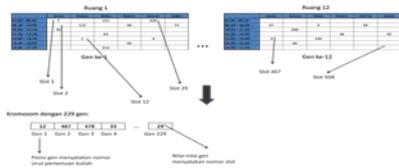
3. Metode Penelitian

Bahan penelitian menggunakan data dari Fakultas Pertanian Universitas Mataram berupa Nama Mata Kuliah, Kode Mata Kuliah, Kode Kelas, Jumlah Mahasiswa, Jumlah Kelompok Mahasiswa, Jumlah SKS Mata Kuliah, dan Kapasitas Ruang Kelas.

Fakultas Pertanian Universitas Mataram terdiri dari 12 ruangan yang terdiri dari:

- 2 ruang dengan kapasitas 140 mahasiswa,
- 2 ruang dengan kapasitas 120 mahasiswa,

- c. 2 ruang dengan kapasitas 90 mahasiswa,
- d. 2 ruang dengan kapasitas 75 mahasiswa,
- e. 4 ruang dengan kapasitas 70 mahasiswa.



Gambar 1 Konversi kromosom dengan 12 gen ke dalam kromosom menggunakan 229 gen



Gambar. 2 Konversi kromosom dengan 229 gen dalam bentuk slot waktu dan ruang

Satu hal yang harus diingat adalah “satu kromosom harus merepresentasikan satu solusi atau jadwal”. Suatu jadwal direpresentasikan ke dalam kromosom dengan panjang sama dengan jumlah pertemuan kuliah. Posisi gen menyatakan nomor urutan pertemuan kuliah yang bernilai *integer* positif, sedangkan nilai-nilai gen menyatakan nomor urutan slot ruang dan waktu (antara 1 sampai 504). Pada Gambar 1, kromosom 3 dimensi (dengan 12 gen) bisa direpresentasikan kembali menjadi kromosom 2 dimensi (dengan 229 gen). Pada kromosom dengan 229 gen, tidak ada satupun gen yang bernilai kosong. Hal ini jauh lebih efisien dibandingkan dengan kromosom dengan 12 gen. Gambar 2 mengilustrasikan konversi kromosom dengan 229 gen dalam bentuk slot waktu dan ruang. Batasan ini di ambil dari permasalahan penjadwalan kuliah yang berstandar internasional yang dikenal dengan sebutan *University Course Timetabling Problem* (UCTP). Berikut adalah batasan keras dan batasan lunak berdasarkan UCTP.

1. *Hard constraint* (batasan keras)
 - a. Dua atau lebih matakuliah tidak dapat diberikan pada suatu waktu dengan ruangan yang sama.
 - b. Setiap matakuliah harus dilakukan pada ruangan yang memenuhi fasilitas dan kapasitas untuk matakuliah tersebut.
 - c. Tidak ada mahasiswa yang mendapatkan dua atau lebih matakuliah pada waktu yang sama.
2. *Soft constraint* (batasan lunak)
 - a. Mahasiswa tidak memperoleh matakuliah pada akhir slot waktu terakhir setiap harinya.
 - b. Mahasiswa tidak mendapat matakuliah lebih dari empat jam secara berturutan.
 - c. Mahasiswa tidak memperoleh hanya satu matakuliah dalam satu hari.

Langkah berikutnya yaitu memilih operator evolusi dan teknik untuk melakukan setting parameter, yaitu:

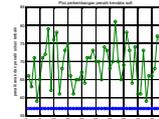
- a. Algoritma *Genetik Tabu Search*

Model populasinya menggunakan *generational replacement* dengan elitisme, seleksi orang tua berupa metode algoritma *Tabu Search*, rekombinasinya menggunakan *uniform crossover*, proses mutasinya menggunakan *swap mutation*.

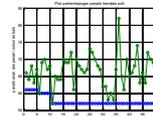
- b. Algoritma *Memetika*

Model populasinya menggunakan *generational replacement* dengan elitisme, seleksi orang tua berupa *tournament selection*, rekombinasinya menggunakan *uniform crossover*, proses mutasinya menggunakan *swap mutation*, dan ada penambahan proses yaitu *local search* (pencarian lokal). Data atau variabel yang diujikan antara lain banyaknya kategori *hard constraint* yang dilanggar dan *soft constraint* yang belum terpenuhi dari solusi terbaik yang merupakan hasil penyelesaian penjadwalan kuliah, serta waktu eksekusi yang dibutuhkan dalam mencari solusi atau pemecahan masalah penjadwalan kuliah yang optimal sebanyak slot waktu tertentu.

4. Hasil dan Pembahasan



Gambar 3 Grafik pinalti *Soft Constraint* untuk slot 36 dari Algoritma *Memetika* pada Semester Ganjil 2009/2010



Gambar 4 Grafik pinalti *Soft Constraint* untuk slot 36 dari Algoritma *Genetik Tabu Search* pada Semester Ganjil 2009/2010

Hasil keluaran aplikasi penjadwalan kuliah untuk tiga semester pada Fakultas Pertanian Universitas Mataram dapat dilihat pada Tabel 4.1 hingga Tabel 4.3. Nilai dari *Hard Constraint* dan *Soft Constraint* merupakan nilai pinalti dari solusi yang dihasilkan. Nilai pinalti menandakan atas banyaknya pelanggaran terhadap suatu batasan (*constraint*) sehingga semakin kecil nilai pinalti, maka kualitas solusi semakin meningkat.

Jika dilihat secara keseluruhan antara Tabel 4.1 hingga Tabel 4.3 dapat dibuat suatu analisa berdasarkan pola yang terbentuk antara hasil keluaran Algoritma *Memetika* dan Algoritma *Genetik Tabu Search*. Pada dasarnya hasil keluaran antara kedua Algoritma ini hampir mirip, namun untuk variabel pengujian waktu eksekusi pada Algoritma *Genetik Tabu Search* dapat 2 hingga 3 kali lipat lebih lama dari waktu eksekusi Algoritma *Memetika*. Untuk slot paling rendah yaitu 36 memiliki hasil keluaran jadwal matakuliah yang menggunakan Algoritma *Genetik Tabu Search* lebih unggul dibandingkan dengan Algoritma *Memetika*, karena Algoritma *Memetika* masih terjadi pelanggaran untuk *Hard-2* dan *Hard-3*, sedangkan Algoritma *Genetik Tabu Search* bisa menanggulangi syarat dari *Hard-2*. Untuk slot waktu yang lebih besar, aplikasi penjadwalan kuliah yang menggunakan metode Algoritma *Memetika* lebih unggul daripada Algoritma *Genetik Tabu Search* meskipun syarat untuk dan *Hard-3* masih dilanggar oleh kedua Algoritma. Untuk syarat dari *Soft-1*, *Soft-2*, dan *Soft-3* dari kedua Algoritma memiliki keluaran jumlah penalti yang tidak berbeda jauh dan memiliki pola bahwa slot berbanding terbalik dengan jumlah penalti *Soft-1*, *Soft-2*, dan *Soft-3*. Hal ini juga berkaitan dengan waktu eksekusi yang dibutuhkan aplikasi penjadwalan

kuliah dari kedua Algoritma, yaitu antara slot dan waktu eksekusi saling berbanding terbalik.

Tabel 4.1 Hasil aplikasi penjadwalan kuliah Algoritma *Memetika* dan Algoritma *Genetik Tabu Search* pada semester ganjil tahun ajaran 2009/2010 dengan menggunakan iterasi 50

Slot	Algoritma Genetik Tabu Search									
	Hard Constraint			Soft Constraint			Total Hard Constraint	Total Soft Constraint	Waktu Eksekusi (detik)	Solusi Optimal Pada Iterasi ke-
	Hard-1	Hard-2	Hard-3	Soft-1	Soft-2	Soft-3				
36	0	4	10	37	20	0	14	57	7427	0
42	0	0	0	34	22	0	0	56	680,91	5
48	0	0	0	29	18	0	0	47	843,26	1
54	0	0	0	22	22	0	0	44	343,51	1
60	0	0	0	16	26	0	0	42	567,59	20
66	0	0	0	13	23	0	0	36	242,8	11

Lanjutan Tabel 4.1

Slot	Algoritma Genetik Tabu Search									
	Hard Constraint			Soft Constraint			Total Hard Constraint	Total Soft Constraint	Waktu Eksekusi (detik)	Solusi Optimal Pada Iterasi ke-
	Hard-1	Hard-2	Hard-3	Soft-1	Soft-2	Soft-3				
36	0	0	12	37	20	0	12	57	10449	11
42	0	0	0	30	22	0	0	52	2424,4	12
48	0	0	0	26	23	0	0	49	1935,5	39
54	0	0	0	23	14	0	0	37	1600,4	33
60	0	0	0	23	18	0	0	41	1526,6	42
66	0	0	0	21	23	0	0	44	594,04	12

Pada Gambar 3 dan Gambar 4 terdapat lingkaran berwarna biru menandakan adanya proses *elitisme* yang berjalan dengan baik dari tiap iterasi. *Elitisme* adalah proses pengkopian individu yang bernilai *fitness* tertinggi atau memiliki jumlah penalti paling kecil dari setiap evolusi agar tidak hilang pada generasi selanjutnya karena adanya hasil evolusi yang berbeda-beda dari setiap kali dilakukan iterasi. Untuk lingkaran yang berwarna hijau menandakan bahwa pada iterasi tersebut memiliki solusi terbaik dari 10 solusi yang ada dengan jumlah penalti paling kecil.

Keunggulan sistem ini pada aplikasi penjadwalan kuliah yang menggunakan dua metode *heuristic* yaitu Algoritma *Genetik Tabu Search* dalam menentukan dan menghasilkan keluaran penjadwalan kuliah yang optimal ini dibuat dengan tujuan mempermudah pembuatan penjadwalan kuliah secara otomatis dan mencegah adanya kesalahan dari faktor manusia seperti yang diterapkan dalam pembuatan jadwal secara manual. Selain itu, sistem ini bisa sebagai acuan dalam menentukan algoritma yang lebih unggul dan optimal untuk digunakan dalam penerapannya ke dalam bentuk penjadwalan kuliah. Kelemahan sistem ini masih menggunakan bahasa pemrograman MATLAB tanpa

menggunakan bantuan database sebagai masukannya. Data masukannya dibuat secara manual dalam bentuk file Microsoft Excel.

Tabel 4.2 Hasil aplikasi penjadwalan kuliah Algoritma *Memetika* dan Algoritma *Genetik Tabu Search* pada semester ganjil tahun ajaran 2011/2012 dengan menggunakan iterasi 50

Slot	Algoritma Memetika									
	Hard Constraint			Soft Constraint			Total Hard Constraint	Total Soft Constraint	Waktu Eksekusi (detik)	Solusi Optimal Pada Iterasi ke-
	Hard-1	Hard-2	Hard-3	Soft-1	Soft-2	Soft-3				
36	0	1	38	38	16	0	39	54	8532,6	7
42	0	0	13	34	17	0	13	51	9202,6	18
48	0	2	2	27	20	0	4	47	8353,1	0
54	0	3	0	25	15	0	3	40	1008,7	0
60	0	0	0	25	21	0	0	46	1136,6	11
66	0	0	0	23	20	0	0	43	618,58	3

Lanjutan Tabel 4.2

Slot	Algoritma Genetik Tabu Search									
	Hard Constraint			Soft Constraint			Total Hard Constraint	Total Soft Constraint	Waktu Eksekusi (detik)	Solusi Optimal Pada Iterasi ke-
	Hard-1	Hard-2	Hard-3	Soft-1	Soft-2	Soft-3				
36	0	2	38	38	14	0	40	52	8654,6	29
42	0	0	14	32	17	0	14	49	8878,2	26
48	0	0	2	33	28	0	2	61	9235,8	0
54	0	0	0	23	20	0	0	43	3478,9	34
60	0	0	0	25	22	0	0	47	1497,8	0
66	0	0	0	14	22	0	0	36	761,14	0

Tabel 4.4 Hasil aplikasi penjadwalan kuliah Algoritma *Memetika* dan Algoritma *Genetik Tabu Search* pada semester genap tahun ajaran 2009/2010 dengan menggunakan iterasi 50

Slot	Algoritma Memetika									
	Hard Constraint			Soft Constraint			Total Hard Constraint	Total Soft Constraint	Waktu Eksekusi (detik)	Solusi Optimal Pada Iterasi ke-
	Hard-1	Hard-2	Hard-3	Soft-1	Soft-2	Soft-3				
36	0	0	20	39	14	0	20	53	7237,2	0
42	0	0	8	27	19	0	8	46	9086,9	49
48	0	0	0	28	18	0	0	46	1348,4	14
54	0	0	0	26	18	0	0	44	653	5
60	0	0	0	25	14	0	0	39	622,07	10
66	0	0	0	14	21	0	0	35	135	2

Lanjutan Tabel 4.3

Slot	Algoritma Genetik Tabu Search						
	Hard Constraint			Soft Constraint			Solusi

	H a r d - 1	H a r d - 2	Har d-3	Soft -1	Soft -2	S o f t - 3	Constr aint	Soft Con stra int	si (detik)	Opt imal Pad a lter asi ke-
36	0	0	22	36	16	0	22	52	7755	12
42	0	0	8	29	18	0	8	47	8398,2	38
48	0	0	0	26	20	0	0	46	1050,3	50
54	0	0	0	23	21	0	0	44	299,27	7
60	0	0	0	21	18	0	0	39	911,12	26
66	0	0	0	18	19	0	0	37	785,78	35

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan maka dapat di buat beberapa kesimpulan sebagai berikut:

1. Penggunaan Algoritma *Memetika* dan Algoritma *Genetik Tabu Search* khususnya dalam permasalahan penjadwalan kuliah dapat dilakukan secara optimal dengan meminimumkan terpenuhinya jumlah batasan keras dan batasan lunak yang diujikan.
2. Untuk slot paling rendah yaitu 36 memiliki hasil keluaran jadwal matakuliah yang menggunakan Algoritma *Genetik Tabu Search* lebih unggul dibandingkan dengan Algoritma *Memetika*, sedangkan untuk slot waktu yang lebih besar, aplikasi penjadwalan kuliah yang menggunakan metode Algoritma *Memetika* lebih unggul daripada Algoritma *Genetik Tabu Search*.
3. Untuk syarat dari *Soft-1*, *Soft-2*, dan *Soft-3* dari kedua Algoritma memiliki keluaran jumlah penalti yang tidak berbeda jauh dan memiliki pola bahwa slot berbanding terbalik dengan jumlah penalti *Soft-1*, *Soft-2*, dan *Soft-3*.
4. Pola waktu eksekusi yang dibutuhkan aplikasi penjadwalan kuliah dari kedua Algoritma, yaitu antara slot dan waktu eksekusi saling berbanding terbalik.

5.2 Saran

Beberapa saran yang dapat disampaikan untuk pengembangan lebih lanjut dari hasil penelitian ini adalah sebagai berikut:

1. Aplikasi yang di buat ini masih menggunakan data masukan Microsoft Excel yang masih dibuat secara manual sehingga dibutuhkan sistem database yang lebih baik.
2. Perlu adanya penyempurnaan khususnya pada aplikasi yang menggunakan Algoritma *Genetik Tabu Search* karena waktu eksekusi yang dibutuhkan untuk mendapatkan solusi yang optimal dibutuhkan waktu yang lama.

Daftar Pustaka

- [1] Mutakhirroh, Ing., 16 Juni 2007, *Pemanfaatan metode heuristik dalam pencarian jalur terpendek dengan algoritma semut dan algoritma genetika*, Seminar Nasional Aplikasi Teknologi Informasi 2007 (SNATI 2007), Yogyakarta.

- [2] Aria, Muhammad., 2008, *Aplikasi algoritma genetik untuk optimasi penjadwalan mata kuliah*, Majalah Ilmiah Unikom, Vol.6, hlm. 17-25.
- [3] Thamilselvan, R., Balasubramanie,P., 2009, *Integrating Genetic Algorithm, Tabu Search Approach for Job Shop Scheduling*, International Journal of Computer Science and Information Security (IJCSIS), Vol. 2, No. 1.
- [4] Altiparmak, Fulya., Dengiz, Berna., 2003, *Optimal design of reliable computer networks: a comparison of metaheuristics*, Journal of Heuristics, 9: 471–487.
- [5] Kusumadewi, Sri., 2003, *Artificial intelligence (teknik dan aplikasinya)*, Graha Ilmu, Yogyakarta.
- [6] Kusumadewi, Sri., Purnomo, Hari., 2005, *Penyelesaian masalah optimasi dengan teknik-teknik heuristik*, Graha Ilmu, Yogyakarta.
- [7] Obtiko, M., 1998, *Introduction to genetic algorithm*.
- [8] Yamada, T., Reeves, C., 1997, *Permutation flowshop, Scheduling by genetic local search*, IEEE conference publication, 232-238, IEEE.
- [9] Darmawan, Irfan., 2011, *hibridisasi Genetic-Tabu Search Algorithm untuk penjadwalan job terhadap beberapa resource di dalam komputasi grid*, Seminar Nasional Teknologi Informasi 2011 (SNATI 2011), ISSN: 1907-5022.

Biodata Penulis

Moh. Ali Albar, memperoleh gelar Diploma (A.Md), Program Studi Diploma Teknik Elektro Fakultas Teknik UGM, lulus tahun 2006. Tahun 2008 memperoleh gelar Sarjana Teknik (S.T) dari Program Studi Teknik Elektro Fakultas Teknik UGM. Tahun 2009 melanjutkan program Pascasarjana dari Program Studi Teknik Elektro Fakultas Teknik UGM. Saat ini masih aktif sebagai mahasiswa Pascasarjana pada Program Studi Teknik Elektro Fakultas Teknik UGM.