
SISTEM PENJADWALAN PRAKTIKUM MENGGUNAKAN ALGORITMA GENETIKA

(Studi Kasus : Teknik Informatika S-1 ITN Malang)

Azrul Haifan¹, Yosep Agus Pranoto², Nurlaily Vendyansyah³
Teknik Informatika – ITN Malang
azrulhaifan@icloud.com

ABSTRAK

Proses pembuatan jadwal praktikum pada Laboratorium Teknik Informatika ITN Malang saat ini masih dilakukan secara manual dan masih menghasilkan cukup banyak kesalahan. Kesalahan yang ada pada jadwal praktikum saat ini adalah, adanya jadwal praktikum mahasiswa yang bentrok dengan jadwal kuliah, alokasi waktu praktikum yang menempati ruang yang sama pada waktu yang bersamaan, dan adanya mahasiswa yang ditempatkan pada ruang yang berbeda disatu waktu. Proses pembuatan jadwal dilakukan dengan cara mencocokkan secara manual data parsial yang ada, diantaranya adalah pencocokan data jadwal kuliah mahasiswa dengan data ketersediaan ruang dan data mata kuliah praktikum

Permasalahan-permasalahan yang telah dipaparkan, dalam bidang Teknologi Informatika dapat diselesaikan menggunakan teori penjadwalan. Dengan menggunakan algoritma genetika, peneliti bermaksud untuk menciptakan sistem penjadwalan praktikum yang dapat meminimalisir kesalahan dengan cara mengolah data jadwal matakuliah, matakuliah praktikum, praktikan, dan data ruangan. Algoritma genetika diharapkan mampu menghasilkan penjadwalan dengan menghindari kesalahan sebanyak mungkin serta memangkas waktu pembuatan jadwal, sehingga jadwal praktikum yang terbentuk bisa menjadi lebih baik.

Setelah dilakukan pengujian dengan data yang didapatkan dari Prodi Teknik Informatika S-1 ITN Malang periode Semester Ganjil Tahun Akademik 2019/2020 didapatkan hasil rata-rata bahwa sistem yang dibangun mampu menghasilkan penjadwalan praktikum dengan error sebesar 2.26 % dalam waktu 15.5 detik.

Kata kunci : *Penjadwalan, Praktikum, Algoritma Genetika, Optimasi.*

1. PENDAHULUAN

Proses pembuatan jadwal praktikum pada Laboratorium Teknik Informatika Institut Teknologi Nasional Malang saat ini masih dilakukan secara manual dan masih menghasilkan cukup banyak kesalahan. Proses pembuatan jadwal dilakukan dengan cara mencocokkan secara manual data parsial yang ada, diantaranya adalah pencocokan data jadwal kuliah mahasiswa dengan data ketersediaan ruang dan data mata kuliah praktikum. Data-data tersebut tersedia dalam bentuk microsoft excel sehingga pencocokan data untuk membentuk jadwal akan membutuhkan waktu yang cukup lama, dan sering menghasilkan kesalahan.

Kesalahan yang ada pada jadwal praktikum yang sering ditemukan diantaranya adalah, adanya jadwal praktikum mahasiswa yang bentrok dengan jadwal kuliah nya, alokasi waktu praktikum yang menempati ruang yang sama pada waktu yang bersamaan, dan adanya mahasiswa yang ditempatkan pada ruang yang berbeda disatu waktu. Untuk mengatasi masalah yang timbul, jadwal praktikum akan dibuat ulang tergantung dari kesalahan yang ditemukan.

Permasalahan-permasalahan yang telah dipaparkan pada paragraf sebelumnya dalam bidang Teknologi Informatika dapat diselesaikan menggunakan teori penjadwalan. Berdasarkan pernyataan Laksono dalam penelitiannya ditahun 2016, Laksono mengajukan algoritma genetika sebagai metode untuk mencari solusi dari suatu permasalahan

penjadwalan yang berkaitan dengan optimalisasi. Di dalam banyak kasus metode ini memiliki solusi yang optimal dan sangat efektif.[1] Dengan menggunakan algoritma genetika, peneliti bermaksud untuk menciptakan sistem pembuatan jadwal praktikum yang lebih minim kesalahan. Algoritma genetika diharapkan mampu menghasilkan solusi yang dapat menghindari kesalahan sebanyak mungkin serta memangkas waktu pembuatan jadwal, sehingga jadwal praktikum yang terbentuk bisa menjadi lebih baik untuk meringankan beban pihak administrasi prodi Teknik Informatika S-1 ITN Malang dalam membuat jadwal praktikum.

1.1 RUMUSAN MASALAH

Rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana cara mengubah penjadwalan praktikum pada Laboratorium Teknik Informatika ITN Malang menjadi lebih efisien ?
2. Bagaimana cara membangun sistem penjadwalan praktikum di Laboratorium Teknik Informatika ITN Malang dengan cara memanfaatkan sumber daya yang ada ?
3. Bagaimana cara merepresentasikan hasil jadwal praktikum yang terbentuk oleh sistem untuk dilampirkan sebagai jadwal praktikum akhir yang dibutuhkan oleh jurusan ?

1.2 TUJUAN

Adapun tujuan dari penelitian ini adalah sebagai berikut :

1. Menerapkan algoritma genetika sebagai metode untuk melakukan penjadwalan praktikum sehingga dapat memangkas waktu dalam pembuatan jadwal.
2. Membangun sistem penjadwalan praktikum dengan data yang didapatkan dari Prodi Teknik Informatika ITN Malang yaitu data jadwal kuliah, data matakuliah praktikum, data praktikan, dan data ruangan serta kapasitasnya.
3. Menghasilkan jadwal praktikum yang telah diolah menggunakan algoritma genetika dalam bentuk microsoft excel.

1.3 BATASAN MASALAH

Adapun batasan masalah dalam penelitian ini adalah:

1. Sistem ini hanya akan diujikan pada Lab. Teknik Informatika Institut Teknologi Nasional Malang dan tidak akan diujikan pada laboratorium lain di Institut Teknologi Nasional Malang.
2. Sistem ini hanya akan diuji dengan data yang didapatkan dari Prodi Teknik Informatika S-1 ITN Malang pada periode Semester Ganjil Tahun Akademik 2019/2020.
3. Data yang diolah pada sistem ini hanya terbatas pada data jadwal kuliah mahasiswa, data ketersediaan ruangan, data mata kuliah praktikum, dan data daftar praktikan yang ada pada Teknik Informatika Institut Teknologi Nasional Malang Periode Ganjil Tahun Akademik 2019/2020.
4. Asumsi bentrok jadwal praktikum dengan jadwal kuliah hanya berdasarkan semester dan kelas matakuliah terkait.
5. Pengujian dilakukan dengan asumsi bahwa Laboratorium Teknik Informatika S-1 ITN Malang telah berpindah gedung dari Gedung Laboratorium Teknik Industri ke Gedung Laboratorium Teknik Elektro.
6. Metode yang digunakan untuk pembuatan jadwal praktikum adalah algoritma genetika.
7. Platform yang akan dikembangkan untuk pembuatan jadwal praktikum ini adalah web.
8. Web server dan database yang digunakan pada pengembangan sistem ini adalah Apache dan My-SQL.
9. Pengembangan website pada aplikasi ini menggunakan PHP dengan Codeigniter sebagai *framework* nya.

2. TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Ditahun 2019 Sanapiah membuat penelitian tentang penyusunan jadwal asisten praktikum menggunakan algoritma genetika, pada penelitian Sanapiah, terdapat 5 tahapan yang dilakukan dalam menyelesaikan permasalahannya, yaitu representasi kromosom, *crossover*, mutasi, fungsi fitness, dan seleksi. Diakhir penelitian Sapinah mendapatkan hasil nilai fitness sama dengan 1, nilai fitness tersebut menyatakan bahwa tidak ada permasalahan atau pelanggaran aturan selama penyusunan jadwal. [2]

Dua tahun sebelum penelitian Sanapiah, di tahun 2017 Adnyana membuat penelitian mengenai Perancangan Sistem Penjadwalan Asisten Dosen Menggunakan Algoritma Genetika. Pada penelitian yang dilakukan oleh Adnyana tersebut, terdapat 4 tahapan dalam menyelesaikan permasalahannya menggunakan algoritma genetika. Yaitu pendefinisian aturan penjadwalan, representasi kromosom, pendefinisian fitness *function*, dan pendefinisian *stopping criteria*. [3]

Pada tahun 2016, tepat setahun sebelum penelitian yang dilakukan oleh Adnyana, Laksono membuat penelitian tentang Sistem Penjadwalan Kuliah Menggunakan Metode Algoritma Genetika. Pada penelitian yang dilakukan oleh Laksono di Fakultas Kedokteran Dan Kesehatan Universitas Muhammadiyah Jakarta bertujuan untuk menghasilkan sistem informasi penjadwalan yang memudahkan pihak administrasi dalam membuat jadwal, dosen, dan mahasiswa yang terlibat. Dalam penelitian yang dikerjakan Laksono, ada beberapa tahapan yang dilakukan yaitu pembangkitan populasi, seleksi, *crossover*, dan mutasi. [1]

Penelitian tentang optimasi penjadwalan mata kuliah dengan algoritma genetika juga pernah dilakukan oleh Suhartono pada tahun 2015, yang dilakukan di AMIK JTS Semarang. Dalam penelitian suhartono terdapat 6 proses algoritma genetika, yaitu proses representasi kromosom, penentuan fungsi fitness penjadwalan, *crossover* kromosom, mutasi kromosom, pembagian ruang, dan kondisi selesai atau *stopping criteria*. Penelitian yang dilakukan oleh Suhartono Dengan bantuan Algoritma Genetika penyusunan penjadwalan mata kuliah dapat dioptimalkan. Program dapat mencari solusi penjadwalan pada waktu yang dapat digunakan baik oleh mahasiswa dan ruangan yang terlibat dalam suatu mata kuliah. [4]

Menoleh ke belakang, pada tahun 2014 Erama dan Wardoyo membuat penelitian tentang penjadwalan pelajaran sekolah dengan algoritma genetika. Dalam penelitiannya Erama dan Wadoyo memodifikasi algoritma genetika dengan mengeliminasi salah satu proses didalamnya, yaitu proses *Crossover*, . Dengan menghilangkan proses *crossover*, menurut penelitian yang telah dilakukan, penjadwalan dengan algoritma genetika yang dimodifikasi memberikan hasil 3.06% lebih baik dari pada algoritma genetika murni. [5]

2.2 Dasar Teori

2.2.1 Teori Penjadwalan

Teori penjadwalan adalah sebuah studi yang mempelajari masalah-masalah distribusi optimal dan pengurutan distribusi pekerjaan dari himpunan yang terbatas. [6] Proses penjadwalan merupakan kegiatan yang dilakukan untuk menyusun kegiatan yang akan dilaksanakan dengan mempertimbangkan semua komponen yang terlibat dalam kegiatan tersebut. Pada permasalahan penjadwalan terdapat 2 (dua) kategori batasan, yaitu:

- 1) *Hard Constraint*, adalah batasan-batasan yang harus dipenuhi. Hasil penjadwalan dinyatakan layak untuk dijadikan solusi apabila tidak ada satupun *hard constraint* yang dilanggar.
- 2) *Soft Constraint*, adalah batasan-batasan yang sebaiknya tidak dilanggar, namun jika karena alasan tertentu dan tidak dapat dihindari, maka batasan tersebut dapat dilanggar.

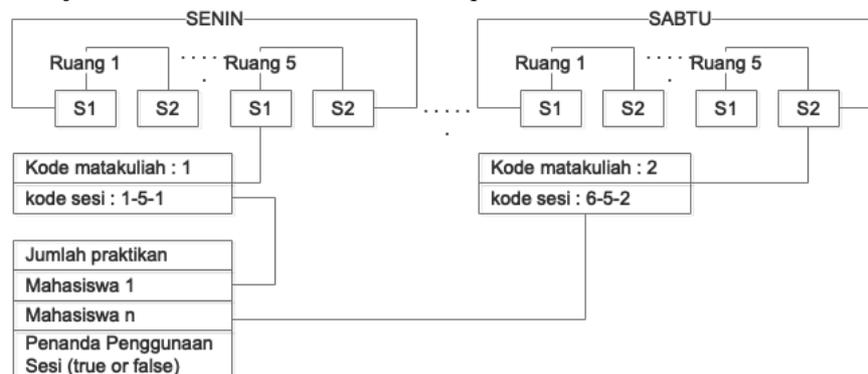
2.2.2 Algoritma Genetika

Algoritma genetika merupakan sebuah algoritma dengan pendekatan komputasional yang digunakan untuk menyelesaikan permasalahan dengan cara mencari sebuah solusi terbaik dari beberapa solusi yang ditawarkan. Cara kerja algoritma genetika terinspirasi dari evolusi biologi, algoritma genetika terlebih dahulu akan menciptakan populasi awal untuk sebuah individu dimana terdapat sebuah populasi yang berisi beberapa kromosom. Setiap kromosom tersebut mewakili sebuah solusi dari permasalahan yang akan diselesaikan. Kromosom-kromosom tersebut akan dievaluasi kualitasnya menggunakan nilai fitness, dimana nilai fitness setiap kromosom bisa jadi berbeda, tergantung dari susunan kromosomnya. Dengan adanya nilai fitness ini, kromosom-kromosom yang ada di dalam sebuah populasi dapat diseleksi agar kromosom terbaik dapat dipertahankan untuk proses di generasi berikutnya. [7]

Terdapat beberapa operator yang digunakan di dalam algoritma genetika, yaitu seleksi, perkawinan silang, dan mutasi. Proses seleksi, penyilangan, dan mutasi kemudian akan diulang, yang menghasilkan sejumlah generasi. Anggapan yang mendasari hal ini adalah, seperti di alam, individu-individu pada sebuah populasi cenderung berkembang dan meningkatkan kemampuan mereka selama individu yang cocok pada generasi ini digunakan untuk menghasilkan generasi baru.

3. METODE PENELITIAN

3.1 Desain Penjadwalan



Gambar 1 Gambaran Representasi Kromosom

3.1.4 Fungsi Fitness

Fungsi fitness merupakan fungsi evaluasi yang akan menilai layak tidaknya sebuah solusi, semakin tinggi nilai fitness, maka sebuah solusi atau kromosom akan dinyatakan semakin layak. Fungsi fitness berisi kumpulan aturan dan bobot

3.1.1 Hard Constraint

Merupakan batasan-batasan yang harus dipenuhi dan tidak boleh dilanggar, berikut adalah *hard constraint* pada penelitian ini :

- 1) Setiap praktikan hanya boleh ditempatkan dalam satu kelas dalam satu waktu.
- 2) Mata kuliah praktikum khusus hanya ditempatkan pada ruangan khusus yang telah dirancang.
- 3) Satu mahasiswa hanya dijadwalkan sekali dalam seminggu untuk satu praktikum.

3.1.2 Soft Constraint

Merupakan batasan-batasan yang sebisa mungkin untuk dihindari, namun masih boleh dilanggar. Berikut adalah *soft constraint* pada penelitian ini :

- 1) Jadwal praktikum tidak bentrok dengan jadwal kuliah mahasiswa.

3.1.3 Representasi Kromosom

Kromosom adalah kumpulan dari gen-gen yang akan membentuk sebuah solusi. Sebuah gen dirangkai sedemikian rupa sehingga dapat mewakili atau merepresentasikan solusi yang akan dibentuk. Dalam penjadwalan ini, digunakan 3 variabel yang akan membentuk susunan kromosom, yaitu banyak hari yang digunakan untuk praktikum, banyak sesi, dan banyaknya ruangan yang akan digunakan. Setiap gen dari kromosom akan mewakili kode matakuliah praktikum, kode sesi, jumlah praktikan, mahasiswa / praktikan terkait, dan penanda penggunaan ruangan.

Jika dimisalkan sebuah praktikum akan diadakan setiap hari dari hari senin hingga sabtu, mulai pukul 13.10 s/d 14.50 untuk sesi pertama, dan pukul 15.00 s/d 16.40 untuk sesi kedua, dan banyaknya ruangan yang digunakan adalah 5 ruangan. Maka besarnya kromosom dapat dihitung dengan mengkalikan semua variabel yang bersangkutan, sehingga didapat $6 * 2 * 5 = 60$. Gambaran representasi kromosom dapat dilihat pada Gambar 1.

pelanggarannya, aturan-aturan fungsi fitness yang digunakan dalam penjadwalan ini dapat dilihat pada Tabel 1.

Tabel 1 Aturan Fungsi Fitness

No	Aturan	Bobot Penalti
1	Jadwal praktikum mahasiswa tidak bentrok dengan jadwal kuliah nya	0.75
2	Setiap praktikan hanya ditempatkan di satu kelas dalam satu waktu	5
3	Mata kuliah praktikum khusus hanya ditempatkan pada ruangan khusus	1.5
4	Satu mahasiswa hanya dijadwalkan sekali dalam seminggu untuk satu praktikum	2.5

Untuk menghitung fungsi fitness, digunakan persamaan 1.

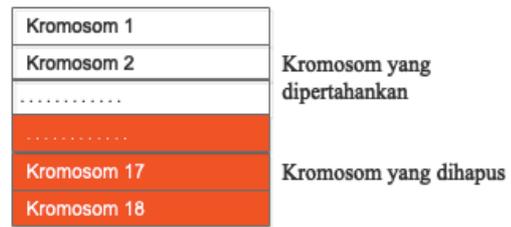
$$Fitness = \frac{1}{1 + \sum n \times w} \dots\dots\dots (1)$$

Persamaan 1 menyatakan n sebagai banyaknya pelanggaran aturan fitness, dan w menyatakan bobot penalti sesuai dengan bobot pada Tabel 1. Penambahan 1 angka pada pembagi ditujukan untuk menanggulangi pembagian tidak terdefinisikan / tidak terhingga.

3.1.5 Seleksi

Metode seleksi yang digunakan adalah *Good Fitness*, metode seleksi ini dilakukan dengan cara menghapus setengah kromosom paling jelek, kemudian karena jumlah kromosom harus sama disetiap populasi, untuk menutupi kekurangan kromosom yang telah dihapus sebelumnya akan dilakukan reproduksi kromosom baru. Jika dimisalkan, satu populasi memiliki 18 kemungkinan solusi / kromosom, maka setelah sebuah kromosom dinilai fungsi fitness nya, 9 kromosom dengan nilai paling jelek akan dihapus, dan dibentuk ulang untuk menutupi kekurangan kromosom pada populasi.

Penggambaran proses seleksi dapat dilihat pada Gambar 2.

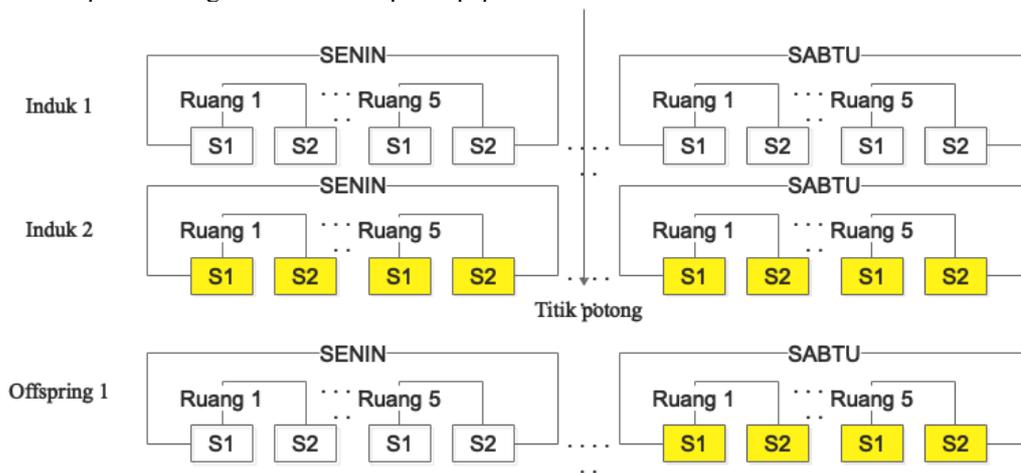


Gambar 2 Penggambaran Proses Seleksi

3.1.6 Perkawinan Silang

Perkawinan silang atau *crossover* adalah metode untuk mengawinkan atau menggabungkan 2 kromosom untuk menciptakan kromosom baru. Hal ini bertujuan untuk menghasilkan kromosom yang lebih tinggi nilai fitness nya. Dalam penjadwalan ini, metode crossover yang digunakan adalah *One-cut point crossover*. Metode *one-cut point crossover* adalah metode yang paling sederhana dimana 2 buah kromosom dipilih sebagai induk, kemudian ditentukan 1 titik potong, selanjutnya kromosom digabungkan menjadi 1 dan disebut *offspring*.

Untuk menentukan kromosom mana yang akan di kawinkan, pertama ditentukan terlebih dahulu probabilitas *crossover* nya, diasumsikan $pC = 0.25$. kemudian bangkitkan bilangan real acak antara 0-1 sejumlah dengan besarnya populasi, nantinya bilangan acak ke-x yang kurang dari pC maka kromosom ke x yang akan dikawin silangkan (dijadikan induk). Sedangkan untuk menentukan posisi perpotongan, dilakukan pengacakan bilangan mulai dari 0 hingga besarnya kromosom dikurangi 1. Penggambaran proses *crossover* dapat dilihat pada Gambar 3.



Gambar 3 Penggambaran Proses Crossover

3.1.7 Mutasi

Mutasi merupakan proses pengubahan salah satu atau beberapa gen pada sebuah kromosom. Proses ini ditujukan untuk menambahkan variasi pada kromosom-kromosom yang terbentuk sebelumnya. Pada penjadwalan ini, digunakan teknik *swapping mutation*, metode ini akan menukar gen pada sebuah kromosom dengan gen disebelahnya.

Pertama kita tentukan probabilitas mutasi terlebih dahulu, diasumsikan $pM = 0.05$, kemudian bangkitkan bilangan random antara 0-1 sebanyak besar populasi * besar kromosom. Nantinya bilangan random ke-k yang kurang dari pM akan dipilih sebagai gen yang akan ditukar. Proses penukaran gen dilakukan dengan cara menukar nilai gen index terpilih, dengan nilai gen index sesudah atau

sebelumnya. Sebelum ditukar, gen yang sudah dipilih terlebih dahulu akan dicari posisinya pada sebuah kromosom. Sehingga penukaran gen pada proses

mutasi tidak akan melibatkan kromosom lainya. Penggambaran proses mutasi dapat dilihat pada Gambar 4.

GEN	GEN 1	GEN 2	GEN 3	GEN 4	...	GEN 1077	GEN 1078	GEN 1079	GEN 1080
Nilai Random	0.1	0.3	0.08	0.09	...	0.9	0.04	0.8	0.55
Offspring N	GEN 1	GEN 2	GEN 3	GEN 4	...	GEN 1077	GEN 1079	GEN 1078	GEN 1080

Gambar 4 Penggambaran proses mutasi

3.1.8 Kriteria Berhenti

Kriteri berhenti adalah fungsi pengecekan apakah perhitungan algoritma genetika sudah saatnya dihentikan atau tidak. Dalam penjadwalan ini, digunakan 2 kriteria berhenti yaitu

1. Nilai fungsi fitness sudah bernilai 1, atau
2. Jumlah generasi atau iterasi sudah mencapai 50.

3.1.9 Perhitungan Error Penjadwalan

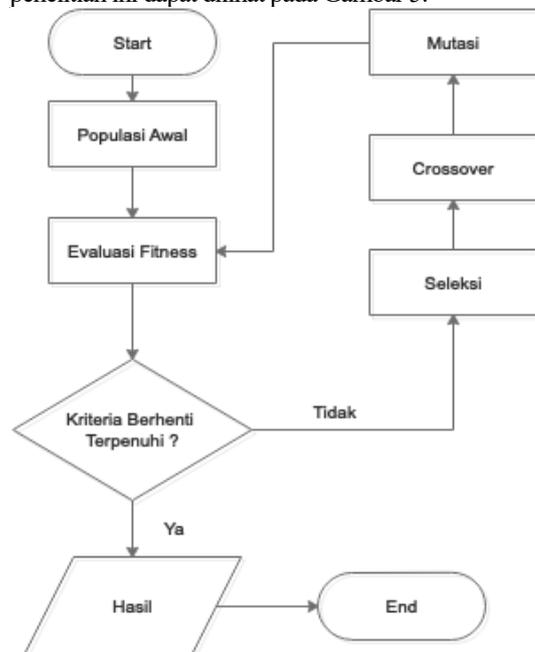
Untuk menghitung error jadwal, digunakan Persamaan 2 sebagai berikut.

$$Error = (\sum(n/m)) / 100 * 25 \dots\dots\dots (2)$$

Persamaan 2 menjelaskan bahwa n adalah nilai pelanggaran aturan, dan m adalah nilai maksimal pelanggaran tiap aturan. Sehingga jika misal aturan 1 bernilai 60, dan aturan lainya bernilai 0. Kemudian m bernilai 652.5 untuk aturan 1, maka error jadwalnya adalah 2.29%.

3.2 Flowchart Algoritma

Flowchart algoritma adalah diagram yang menunjukkan alur atau proses tahapan algoritma dari awal hingga selesai. Flowchart algoritma pada penelitian ini dapat dilihat pada Gambar 5.



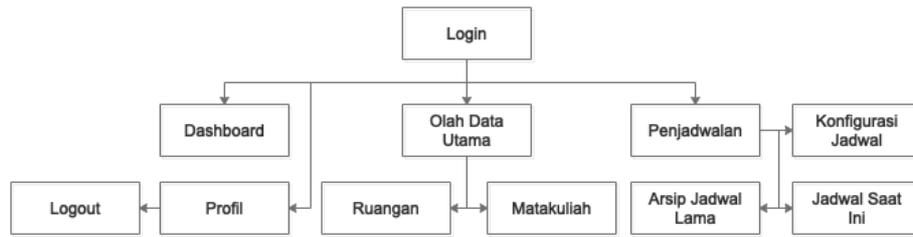
Gambar 5 Flowchart Algoritma

Gambar 5 menjelaskan alur algoritma genetika dari awal hingga akhir, yaitu :

1. Proses pembangkitan populasi awal, populasi adalah kumpulan dari solusi atau kromosom, dimana representasi kromosom dapat dilihat pada poin 3.1.3.
2. Setelah proses pembangkitan populasi awal, selanjutnya adalah proses evaluasi fungsi fitness, detail dari proses ini dapat dilihat pada poin 3.1.4.
3. Setelah melalui proses evaluasi fitness, tahapan ketiga adalah pengecekan apakah kriteria berhenti telah terpenuhi atau tidak. Jika terpenuhi, pembuatan jadwal dinyatakan selesai, dan hasil penjadwalan akan ditampilkan. Namun jika tidak, maka proses dilanjutkan ke tahapan ke-4. Kriteria berhenti dapat dilihat pada poin 3.1.8.
4. Apabila kriteria berhenti tidak terpenuhi, maka tahapan selanjutnya adalah menyeleksi kromosom-kromosom pada sebuah populasi sebelum dilakukan proses ke-5 atau proses perkawinan silang. Detail dari proses seleksi dapat dilihat pada poin 3.1.5.
5. Tahapan selanjutnya adalah perkawinan silang, proses perkawinan silang dapat dilihat lebih detail pada poin 3.1.6.
6. Setelah melalui proses perkawinan silang, kromosom akan dimutasi sesuai dengan poin 3.1.7.
7. Kembali ke langkah ke-3 yaitu pengecekan fungsi fitness, hingga kriteria berhenti terpenuhi dan proses penjadwalan dinyatakan selesai.

3.3 Struktur Menu

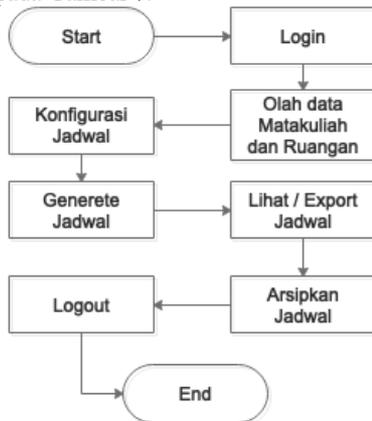
Struktur menu merupakan blok diagram yang menunjukkan urutan pengaksesan fungsi program atau aplikasi. Menu awal pada penelitian ini adalah menu halaman login. Menu login digunakan oleh seorang administrator untuk masuk ke halaman utama sistem, pada halaman utama sistem, terdapat beberapa menu lanjutan yaitu menu dashboard yang berisi rangkuman informasi sistem, diantaranya adalah informasi jumlah praktikum saat ini, jumlah ruangan, error jadwal saat ini, durasi penjadwalan terakhir dan rata-ratanya, total praktikan, dan total jadwal terbentuk. Selanjutnya juga terdapat menu olah data matakuliah dan olah data ruangan, menu konfigurasi penjadwalan, menu hasil jadwal saat ini, menu arsip jadwal, serta menu terakhir adalah menu profil yang berisi profil dari pengembang dan fungsi logout. Struktur menu dari penelitian ini dapat dilihat pada Gambar 6.



Gambar 6 Desain Struktur Menu Website

3.4 Flowchart Sistem

Flowchart Sistem menunjukkan alur kerja atau proses urutan sistem dari awal berjalanya sistem hingga berakhir, alur ini hanya menggambarkan langkah-langkah umum dari awal mula hingga akhir proses pembentukan jadwal. Flowchart sistem dapat dilihat pada Gambar 7.



Gambar 7 Flowchart Sistem

Gambar 7 menjelaskan bahwa proses pertama adalah proses *login*, setelah seorang administrasi *login* seorang administrasi akan menuju proses pengolahan data matakuliah dan ruangan, kemudian dilanjutkan ke proses konfigurasi jadwal. Setelah proses konfigurasi selesai, proses selanjutnya adalah proses *generate* jadwal, yang kemudian berlanjut ke proses lihat atau *export* data jadwal yang terbentuk. Selanjutnya jadwal yang terbentuk akan ditampilkan, kemudian seorang admin dapat memasuki proses pengarsipan jadwal. Setelah semua proses yang berhubungan dengan penjadwalan selesai, admin dapat menuju proses *logout*, dan keseluruhan sistem selesai.

4. HASIL DAN PEMBAHASAN

4.1 Hasil

1) Halaman Inisialisasi Konfigurasi Penjadwalan

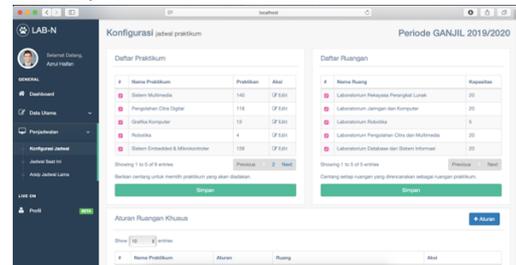
Pada halaman inisialisasi penjadwalan seorang admin diharuskan untuk mengunggah data jadwal kuliah sebagai permulaan sebelum dapat memulai proses konfigurasi penjadwalan. Tampilan halaman inisialisasi konfigurasi Penjadwalan dapat dilihat pada Gambar 8.



Gambar 8 Tampilan Halaman Inisialisasi Penjadwalan

2) Halaman Konfigurasi Jadwal

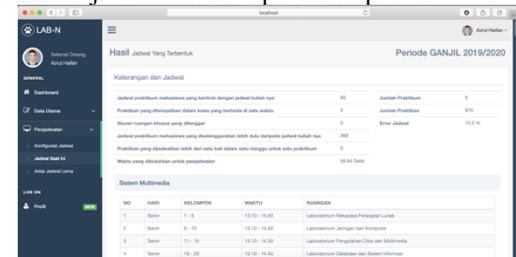
Setelah seorang admin mengunggah berkas data jadwal kuliah, seorang admin dapat melanjutkan proses konfigurasi penjadwalan dan mulai mengunggah berkas daftar praktikan untuk setiap matakuliah praktikum. Seorang admin harus mengonfigurasi ruangan yang digunakan, menambah aturan ruangan khusus (bila ada), mengatur hari dan sesi praktikum, dan mengonfigurasi bobot nilai pelanggaran fungsi fitness (apabila diperlukan). Halaman konfigurasi jadwal dapat dilihat pada Gambar 9.



Gambar 9 Tampilan Halaman Konfigurasi Penjadwalan

3) Halaman Jadwal Saat Ini

Pada halaman jadwal saat ini, akan ditampilkan jadwal yang telah terbentuk, serta beberapa informasi tambahan yaitu jumlah jadwal praktikum mahasiswa yang bentrok dengan jadwal kuliah nya, jumlah praktikan yang ditempatkan dalam kelas yang berbeda dalam satu waktu, jumlah aturan ruangan khusus yang dilanggar, jumlah jadwal praktikum mahasiswa yang diselenggarakan lebih dulu dari jadwal kuliahnya, jumlah praktikan yang dijadwalkan lebih dari satu kali dalam satu minggu, waktu yang dibutuhkan untuk membuat jadwal, jumlah praktikum, jumlah praktikan, dan *error* jadwal dalam bentuk persentase. Tampilan halaman jadwal saat ini dapat dilihat pada Gambar 10.

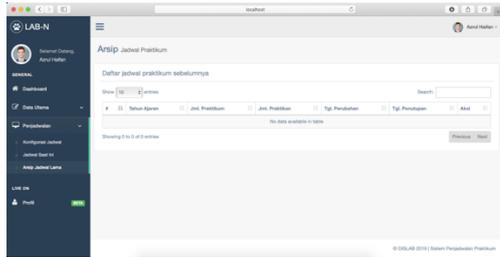


Gambar 10 Tampilan Halaman Jadwal Saat Ini

4) Halaman Arsip Jadwal

Pada halaman arsip jadwal, seorang admin dapat melihat dan mengunduh daftar jadwal praktikum lama yang telah diarsipkan. Pada menu ini, seorang admin tidak dapat lagi mengubah konfigurasi jadwal yang

terbentuk. Tampilan halaman arsip jadwal dapat dilihat pada Gambar 11.



Gambar 11 Tampilan Halaman Arsip Jadwal

4.2 Pengujian

1) Pengujian Fungsionalitas Sistem

Pada pengujian fungsionalitas sistem, akan dilakukan percobaan untuk menguji semua fungsi yang ada disistem apakah bekerja dengan normal atau tidak. Pengujian dilakukan pada 3 browser pada perangkat dengan spesifikasi CPU yang digunakan adalah Intel Core I3 4200U 2.3 Ghz, RAM 12 GB, dan Penyimpanan SATA SSD. Hasil dari pengujian fungsionalitas sistem ini dapat dilihat pada Tabel 2.

Tabel 2 Tabel Pengujian Fungsionalitas Sistem

Fungsi	Safari	Google Chrome	Firefox
Login dan Sesi	√	√	√
Informasi Dashboard	√	√	√
Tambah Data Matakuliah	√	√	√
Ubah Data Matakuliah	√	√	√
Hapus Data Matakuliah	√	√	√
Tambah Data Ruangan	√	√	√
Ubah Data Ruangan	√	√	√
Hapus Data Ruangan	√	√	√
Unggah berkas jadwal kuliah	√	√	√
Konfigurasi Matakuliah Praktikum	√	√	√
Unggah data praktikan	√	√	√
Konfigurasi Data ruangan praktikum	√	√	√
Konfigurasi Aturan ruangan khusus	√	√	√
Tambah Data sesi praktikum	√	√	√
Ubah Data sesi praktikum	√	√	√
Hapus Data sesi praktikum	√	√	√
Konfigurasi sesi praktikum	√	√	√
Konfigurasi bobot pelanggaran Fungsi Fitness	√	√	√
Fungsi Tampilkan Hasil Jadwal	√	√	√
Fungsi Export Jadwal	√	√	√
Fungsi tutup atau arsipkan jadwal	√	√	√
Fungsi Logout	√	√	√

Tabel 2 menjelaskan hasil pengujian fungsionalitas sistem yang telah dilakukan, versi browser safari yang digunakan adalah versi 13.0.3, google chrome dengan versi 78.0.3904.108, dan firefox dengan versi 70.0.1. Dimana tanda √ menandakan bahwa pengujian berjalan dengan baik sesuai dengan harapan peneliti,

sedangkan tanda X menandakan bahwa pengujian tidak sesuai dengan harapan.

2) Pengujian Nilai Probabilitas Crossover dan Mutasi

Pengujian nilai probabilitas *crossover* dan mutasi dilakukan untuk melihat hasil penjadwalan apabila diberikan nilai probabilitas *crossover* dan mutasi yang berbeda. Pengujian ini akan melihat hasil fungsi fitness jadwal terpilih, durasi penjadwalan, dan *error* jadwal yang terbentuk. Pengujian nilai probabilitas ini akan dilakukan sebanyak 5 kali dengan probabilitas *crossover* dilambangkan pC dan probabilitas mutasi dilambangkan dengan pM, dimana rincian percobaan akan dipaparkan pada Tabel 3.

Tabel 3 Tabel Pengujian nilai probabilitas *crossover* dan mutasi

No	pC	pM	Nilai	Hasil
1	0.25	0.1	Fitness 1	80
			Fitness 2	0
			Fitness 3	0
			Fitness 4	0
			Durasi Penjadwalan	15.49 Detik
			Error Jadwal	2.3 %
2	0.5	0.01	Fitness 1	80
			Fitness 2	0
			Fitness 3	0
			Fitness 4	0
			Durasi Penjadwalan	15.86 Detik
			Error Jadwal	2.3 %
3	0.75	0.01	Fitness 1	80
			Fitness 2	0
			Fitness 3	0
			Fitness 4	0
			Durasi Penjadwalan	16.28 Detik
			Error Jadwal	2.3 %
4	0.5	0.25	Fitness 1	80
			Fitness 2	0
			Fitness 3	0
			Fitness 4	0
			Durasi Penjadwalan	16.44 Detik
			Error Jadwal	2.3 %
5	0.55	0.025	Fitness 1	73
			Fitness 2	0
			Fitness 3	0
			Fitness 4	0
			Durasi Penjadwalan	15.82 Detik
			Error Jadwal	2.1 %

Setelah dilakukan pengujian dengan hasil yang dapat dilihat pada Tabel 3, didapatkan kesimpulan bahwa perubahan nilai probabilitas *crossover* dan mutasi pada penelitian ini tidak jauh berbeda durasi penjadwalan dan *error* nya.

3) Pengujian User

Pengujian *user* merupakan nilai atau respon yang diberikan oleh user terhadap aplikasi. Pada penelitian ini, kuisioner diberikan kepada pihak administrasi yang ada pada Prodi Teknik Informatika-S1 Institut Teknologi Nasional Malang. Terdapat 6 pertanyaan yang akan diajukan terhadap user, dimana hasil dari pengujian user dapat dilihat pada Tabel 4.

Tabel 4 Tabel Hasil Pengujian User

No	Pertanyaan	Penilaian		
		B	C	K
1	Bagaimana kejelasan ukuran dan warna <i>font</i> yang ada pada aplikasi ?	1	0	0
2	Bagaimana kesesuaian atau kejelasan warna tombol dan <i>background</i> pada aplikasi?	1	0	0
3	Bagaimana kejelasan alur menu yang ada pada aplikasi ?	1	0	0
4	Bagaimana menurut anda kejelasan dan kesesuaian form-form yang ada pada aplikasi saat ini ?	1	0	0
5	Bagaimana menurut anda kemudahan pembuatan jadwal praktikum pada aplikasi ini ?	1	0	0
6	Bagaimana tingkat kesesuaian hasil export dan alur penjadwalan pada aplikasi terhadap kebutuhan jadwal praktikum saat ini ?	1	0	0
TOTAL		6	0	0

Hasil dari Tabel 4 memaparkan hasil penilaian *user* terhadap aplikasi yang telah dibangun. Kolom B menunjukkan nilai Baik, C menunjukkan nilai Cukup, sedangkan K menunjukkan nilai Kurang. Penilaian diberikan oleh satu administrasi Prodi Teknik Informatika-S1 ITN Malang, dimana dapat dilihat dari Tabel 4 bahwa total penilaian Baik adalah 6, sedangkan total penilaian untuk Cukup dan Kurang adalah 0. Maka dengan ini dapat ditarik kesimpulan bahwa *user* menyatakan bahwa aplikasi berjalan dengan baik 100%.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pembahasan penelitian ini yang telah dipaparkan pada bab sebelumnya, maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Penjadwalan praktikum menggunakan algoritma genetika pada penelitian ini dapat memangkas waktu penjadwalan praktikum menjadi rata-rata 15.5 detik dibandingkan dengan penjadwalan manual yang membutuhkan waktu sehari-hari.
2. Algoritma genetika mampu menghasilkan solusi yang baik, namun tidak menjamin solusi akhir yang ditawarkan adalah solusi yang paling optimum.
3. Algoritma genetika mampu menghasilkan jadwal praktikum dengan rata-rata error sebesar 2.3% menggunakan data yang didapatkan pada periode Semester Ganjil Tahun Akademik 2019/2020.

5.2 Saran

Berdasarkan hasil penelitian yang telah diperoleh, didapatkan beberapa saran yang dapat peneliti

sampaikan guna perbaikan pada perkembangan penelitian ini dikemudian hari, yaitu :

1. Sistem penjadwalan ini dapat lebih disempurnakan apabila kelengkapan data nya ditambah. Dalam hal ini, penambahan data daftar mahasiswa setiap matakuliah, dapat menghilangkan perhitungan bentrok jadwal praktikum dengan jadwal kuliah berdasarkan asumsi semester dan kelas matakuliah terkait.
2. Sistem ini perlu dilakukan perbandingan menggunakan kasus yang sama dengan menggunakan metode lain yang sejenis seperti algoritma JST dan algoritma logika fuzzy.
3. Untuk pengembangan lebih lanjut, sistem telah dibuat sedinamis mungkin dengan harapan penelitian selanjutnya dapat menerapkan penjadwalan ini untuk keseluruhan prodi yang ada di Institut Teknologi Nasional Malang.

DAFTAR PUSTAKA

- [1] Laksono, 2016, Sistem Penjadwalan Kuliah Menggunakan Metode Algoritma Genetika (Studi Kasus: Fakultas Kedokteran Dan Kesehatan Universitas Muhammadiyah Jakarta), *Studia Informatika*, pp.177-188
- [2] Sanapiah, M.D.S., Rofi'ah, A., Jayanti, H.D., Arliana, A.G. and Wijayaningrum, V.N., 2019. Penyusunan Jadwal Asisten Praktikum Menggunakan Algoritma Genetika. *SISTEMASI*, 8(2), pp.282-287.
- [3] Adnyana, I.M.B., 2017. Perancangan Sistem Penjadwalan Asisten Dosen Menggunakan Algoritma Genetika (Studi Kasus: STIKOM Bali). *E-Proceedings KNS&I STIKOM Bali*, pp.569-574.
- [4] Suhartono, E., 2015. Optimasi Penjadwalan Mata Kuliah dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang). *INFOKAM*, 11(5).
- [5] Erama, R. and Wardoyo, R., 2014. Modifikasi Algoritma Genetika Untuk Penyelesaian Permasalahan Penjadwalan Pelajaran Sekolah. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 8(2), pp.111-120.
- [6] Tanaev, V., Gordon, W., and Shafransky, Y., 2012, *Scheduling Theory. Single-Stage Systems*, Berlin, Springer Science & Business Media.
- [7] Wilkinson, Allen, 2010, *Parallel Programming - Teknik dan Aplikasi Menggunakan Jaringan Workstation & Komputer Paralel*, Yogyakarta, Andi Publisher.