OPTIMASI PENJADWALAN KAPAL PENYEBERANGAN KETAPANG – GILIMANUK MENGGUNAKAN ALGORITMA GENETIKA

SKRIPSI

Untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun oleh: Ria Febriyana NIM: 125150207111023



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

OPTIMASI PENJADWALAN KAPAL PENYEBERANGAN KETAPANG – GILIMANUK MENGGUNAKAN ALGORITMA GENETIKA

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh : Ria Febriyana NIM: 125150207111023

Skripsi ini telah diuji dan dinyatakan lulus pada 22 Agustus 2016 Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D NIP: 19720919 199702 1 001

Mengetahui Ketua Jurusan Teknik Informatika

<u>Tri Astoto Kurniawan, S.T, M.T, Ph.D</u> NIP: 197105182003121001



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsurunsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 22 Agustus 2016



Ria Febriyana

NIM: 125150207111023



KATA PENGANTAR

Segala puji dan syukur kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul "Optimasi Penjadwalan Kapal Penyeberangan Ketapang – Gilimanuk Menggunakan Algoritma Genetika" dengan baik.

Penulis menyadari bahwa dalam penyusunan skripsi ini tidak terlepas dari dukungan, motivasi, kerjasama, masukan, maupun bimbingan dari berbagai pihak. Oleh karena itu, penulis mengucapkan terimakasih yang sebesar-besarnya kepada:

- 1. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D, selaku dosen pembimbing skripsi dan Dekan Fakultas Ilmu Komputer yang telah memberikan bimbingan, masukan dan pengarahan bagi penulis.
- 2. Bapak Tri Astoto Kurniawan, S. Si, M.T, Ph.D, selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya Malang beserta jajarannya.
- 3. Bapak Agus Wahyu Widodo, S.T, M.Cs, selaku Ketua Program Studi Teknik Informatika Universitas Brawijaya Malang beserta jajarannya.
- 4. Seluruh bapak dan ibu dosen program studi Teknik Informatika yang selama ini telah mendidik dan memberikan ilmunya kepada penulis.
- 5. Orang tua dan saudara penulis yang selalu memberikan doa dan dukungan hingga terselesainya skripsi ini.
- 6. Sahabat dan teman-teman Teknik Informatika angkatan 2012 Universitas Brawijaya dan semua teman-teman yang tidak dapat penulis sebutkan satu per satu yang selalu memberikan semangat, doa, masukan, dan dukungan agar penulis bisa menyelesaikan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan dan kesalahan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun untuk menyempurnakan skripsi ini. Akhir kata penulis berharap semoga skripsi ini dapat memberikan manfaat positif bagi semua pihak yang membaca dan menggunakannya. Aamiin.

Malang, 22 Agustus 2016

Ria Febriyana febriyanaria@gmail.com

ABSTRAK

Penyusunan penjadwalan kapal penyeberangan di Pelabuhan Ketapang -Gilimanuk sangatlah penting agar dapat memberikan pelayanan yang maksimal kepada para penumpang pengguna jasa kapal laut. Akan tetapi masih banyak kendala yang dihadapi, karena pembuatan penjadwalan masih dibuat secara manual sehingga terkadang ada nama kapal yang sama dalam sehari yang beroperasi dan tidak adanya keadilan pada pembagian porsi masing-masing kapal. Oleh karena itu, untuk mengatasi permasalahan tersebut diperlukan suatu sistem komputerisasi penjadwalan kapal guna mempercepat pengaturan jadwal pemberangkatan kapal penyeberangan Ketapang – Gilimanuk. Penerapan metode algoritma genetika dalam permasalahan penjadwalan kapal mampu menghasilkan solusi yang optimal dengan menggunakan representasi kromosom permutasi bilangan integer, metode crossover menggunakan one cut-point crossover, mutasi menggunakan reciprocal exchange mutation, dan seleksi menggunakan elitism selection. Dari pengujian parameter didapat hasil yaitu antara lain ukuran populasi sebesar 180, banyaknya generasi 200, serta kombinasi crossover rate=0,6 dan mutation rate=0,4.

Kata kunci: Penjadwalan, Algoritma Genetika.



ABSTRACT

Arrangement schedule for dispatching ships in Ketapang – Gilimanuk Harbor is very important for giving the best service to the passengers who are using ships as their transportation. However, there are still a lot of problems because of the arrangement schedule is made manually, so sometimes there are some ships with same name work in a day; another problem is there is no equality in distributing the portion for each ships. Therefore, to solve that kind of problems, dispatching schedule using computerization is needed in order to make the dispatching schedule of the ships faster in Ketapang – Gilimanuk Harbor. The use of genetic algorithm method in dispatching schedule of ships is giving the optimal solution with using representative of numeral integer chromosome permutation, crossover method using one cut-point crossover, mutation using reciprocal exchange mutation, and selection using elitism selection. From parameter test's result, there are some outcomes such as 180 population size, 200 the amount of generation, and also the combination of crossover rate=0,6 and mutation rate=0,4.

Key words: Arrangement Schedule, Genetic Algorithm.



DAFTAR ISI

OPTIMASI PENJADWALAN KAPAL PENYEBERANGAN KETAPANG MENGGUNAKAN ALGORITMA GENETIKA	i
PENGESAHAN	
PERNYATAAN ORISINALITAS	
KATA PENGANTAR	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI DAFTAR TABEL	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
BAB 1 PENDAHULUAN	
1.1 Latar belakang	1
1.2 Rumusan masalah	
1.3 Tujuan	
1.4 Manfaat	
1.5 Batasan masalah	4
1.6 Sistematika pembahasan	
BAB 2 LANDASAN KEPUSTAKAAN	
2.1 Kajian Pustaka	6
2.2 Penjadwalan Kapal	
2.2.1 Tujuan Penjadwalan	
2.3 Algoritma Genetika	9
2.3.1 Kelebihan dan Kekurangan Algoritma Genetika	10
2.4 Komponen-Komponen Utama Algoritma Genetika	10
2.4.1 Generasi Awal	11
2.4.2 Representasi Kromosom	11
2.4.3 Nilai Fitness	11
2.4.4 Crossover	
2.4.5 Mutasi	
2.4.6 Seleksi	
2.4.7 Kondisi Berhenti (Termination Condition)	14

3.3 Algoritma yang Digunakan	17
3.5 Pengujian Algoritma	18
3.5.1 Pengujian Ukuran Populasi	18
3.5.2 Pengujian Banyaknya generasi	19
3.5.3 Pengujian Kombinasi <i>Crossover Rate (cr)</i> dan <i>Mutation Rate</i>	20
BAB 4 PERANCANGAN	
4.1 Formulasi Permasalahan	22
4.2 Siklus Algoritma	24
4.3 Siklus Penyelesaian Algoritma Genetika	25
4.3.1 Representasi Kromoson dan Perhitungan Fitness	25
4.3.2 Inisialisasi Populasi Awal	
4.3.3 Reproduksi	
4.3.4 Evaluasi dan Seleksi	30
4.4 Perancangan <i>User Interface</i>	33
BAB 5 IMPLEMENTASI	35
5.1 Struktur <i>Class</i>	
5.1.1 Class Algoritma Genetika	
5.2 Potongan Source Code Utama	
5.2.1 Pembangkitan Populasi Awal	
5.2.2 Proses <i>Crossover</i>	38
5.2.3 Proses Mutasi	39
5.2.4 Perhitungan Nilai <i>Fitness</i>	39
UNIXIVERERSITATAS BY BRAYAWIIATA	
The state of the s	

3.2 Teknik Pengumpulan Data......17

5.2.5 Proses Seleksi	40
BAB 6 PENGUJIAN DAN PEMBAHASAN	41
6.1 Pengujian	41
6.1.1 Hasil Pengujian Ukuran Populasi	41
6.1.2 Hasil Pengujian Banyaknya Generasi	44
6.1.3 Hasil Pengujian Kombinasi <i>Crossover Rate (cr)</i> dan Rate(mr)	
6.1.4 Hasil Pengujian Parameter Terbaik	51
BAB 7 PENUTUP	
7.1 Kesimpulan	53
7.2 Saran	53
DAFTAR PUSTAKA	54



DAFTAR TABEL

Tabel 2.1 Penelitian yang Telah Dilakukan Sebelumnya	6
Tabel 3.1 Perancangan Pengujian Ukuran Populasi	. 19
Tabel 3.2 Perancangan Pengujian Banyaknya generasi	. 20
Tabel 3.3 Perancangan Pengujian Kombinasi Crossover Rate dan Mutation Rat	e21
Tabel 4.1 Data Nama Kapal di Dermaga LCM	. 22
Tabel 4.2 Data <i>Trip</i> Kapal di Dermaga LCM	. 23
Tabel 4.3 Contoh Jadwal Kapal Penyeberangan Ketapang – Gilimanuk	. 23
Tabel 4.4 Konversi Kromosom Menjadi Jadwal Kapal selama 7 hari	. 26
Tabel 4.5 Jenis Pelanggaran Penjadwalan Kapal	. 27
Tabel 4.6 Hasil Perhitungan Nilai Fitness	. 31
Tabel 4.7 Hasil Proses Seleksi <i>Elitism</i> Pertama	. 31
Tabel 4.8 Hasil Proses Seleksi <i>Elitism</i> Kedua	. 32
Tabel 6.1 Hasil Pengujian Ukuran Populasi	. 41
Tabel 6.2 Hasil Pengujian Banyaknya generasi	. 45
Tabel 6.3 Hasil Pengujian Kombinasi Crossover Rate dan Mutation Rate	. 48



DAFTAR GAMBAR

Gambar 2.1 Contoh Kromosom Menggunakan Representasi Permutasi 1	.1
Gambar 2.2 Contoh One Cut-Point Crossover	.3
Gambar 2.3 Contoh Reciprocal Exchange Mutation1	.3
Gambar 2.4 Contoh Insertion Mutation 1	.4
Gambar 3.1 Diagram Blok Tahapan Penelitian 1	
Gambar 4.1 Flowchart Proses Algoritma Genetika	.5
Gambar 4.2 Representasi Kromosom	
Gambar 4.3 Contoh Individu Awal2	28
Gambar 4.4 Proses <i>Crossover</i>	9
Gambar 4.5 Proses Mutasi	0
Gambar 4.6 Halaman Data Kapal Penyeberangan Ketapang – Gilimanuk 3	3
Gambar 4.7 Halaman Proses Penjadwalan Kapal Penyeberangan 3	4
Gambar 4.8 Halaman Hasil Penjadwalan Kapal Penyeberangan 3	4
Gambar 5.1 Class Diagram Algoritma Genetika	5
Gambar 6.1 Grafik Hasil Pengujian Ukuran Populasi4	4
Gambar 6.2 Grafik Hasil Pengujian Banyaknya Generasi4	7
Gambar 6.3 Grafik Hasil Pengujian Kombinasi <i>Crossover Rate</i> dan <i>Mutation Rat</i>	
Gambar 6.4 Penjadwalan Kapal yang Dihasilkan 5	

BAB 1 PENDAHULUAN

Bab ini membahas tentang latar belakang dari penyusunan skripsi, rumusan masalah, tujuan, manfaat yang akan diperoleh, batasan masalah, serta sistematika pembahasan mengenai penyusunan skripsi ini.

1.1 Latar belakang

Indonesia merupakan negara yang terdiri dari beberapa pulau yang terbentang luas dari Sabang sampai Merauke, berada diantara 2 benua dan 2 samudra, yaitu Benua Asia dan Benua Australia, Samudra Hindia dan Samudra Pasifik serta berada tepat di garis khatulistiwa. Indonesia juga termasuk salah satu negara kepulauan terbesar di dunia, yang terdiri dari 17.508 pulau dan masing-masing dari pulau, dipisahkan oleh lautan yang begitu luas. Adapun total luas daratan wilayah Indonesia 1.922.570 km² dan total luas perairannya yaitu sebesar 3.257.483 km² (Atlas Global, 2014). Dengan total luas perairan tersebut, tentunya akan menyulitkan manusia yang akan menyebrang dari pulau satu ke pulau yang lain. Apabila manusia ingin berpergian ke pulau lain tersebut, dibutuhkan suatu alat transportasi yang dapat mengantarkannya.

Seperti halnya lautan yang berada di ujung timur pulau Jawa dan pulau Bali, dipisahkan oleh Selat Bali. Apabila masyarakat yang ada di pulau Jawa ingin bepergian menuju pulau Bali tentunya akan merasa kesulitan jika tidak ada suatu alat transportasi yang dapat menghantarkan mereka menuju ke pulau Bali, begitu juga sebaliknya. Dengan kemajuan zaman yang didukung oleh kemajuan teknologi yang canggih maka dibutuhkan suatu alat transportasi yang mudah dan cepat yaitu dengan menggunakan kapal laut. Adanya angkutan laut seperti kapal tersebut tentu sangat penting untuk membuka akses dan mengembangkan perekonomian lokal (Susantono, 2013) serta membantu seseorang dalam bertransportasi. Menurut Wismulyani (2008) pengertian transportasi adalah suatu sistem yang penting bagi manusia yang digunakan sebagai sarana untuk menjalankan roda kehidupan manusia. Transportasi utama masyarakat Indonesia adalah kapal, karena negara tersebut merupakan negara kepulauan yang terdiri dari beberapa pulau.

Di ujung timur Pulau Jawa yaitu tepatnya di Kabupaten Banyuwangi memiliki tempat penyeberangan kapal yang berfungsi sebagai pelabuhan atau tempat bersandarnya kapal yang akan menghantarkan manusia atau barang dari Pelabuhan Ketapang, Kabupaten Banyuwangi menuju Pelabuhan Gilimanuk yang berada di pulau Bali, begitu juga sebaliknya. Keberadaan pelabuhan ini, tentu sangat penting dan membantu semua sektor dalam proses penjadwalan pemberangkatan dan kedatangan kapal yang ada di Pelabuhan Ketapang, Kabupaten Banyuwangi. Salah satu tugas penting dari Pelabuhan Ketapang adalah pembuatan jadwal pemberangkatan kapal yang beroperasi setiap hari.

Penjadwalan pemberangkatan kapal yang baik tentu akan menghasilkan pelayanan yang prima dan memuaskan para penumpang pengguna jasa

transportasi angkutan laut yaitu kapal. Semua kapal yang ada di pelabuhan Ketapang jumlahnya banyak dan terjadwal secara rutin oleh para pegawai di Kantor PT ASDP Cabang Ketapang, Kabupaten Banyuwangi berdasarkan utusan Otoritas Pelabuhan Penyeberangan. Akan tetapi masih ada kendala yang dihadapi dalam proses penyusunan penjadwalan pemberangkatan kapal, dikarenakan pembuatan penjadwalan masih dibuat secara manual, dengan jumlah kapal yang banyak dan dermaga yang terbatas, sehingga setiap harinya dituntut untuk membuat penjadwalan pemberangkatan kapal. Oleh karena itu, untuk mendapatkan jadwal pemberangkatan kapal penyeberangan secara adil dan rutin untuk memenuhi penjadwalan yang baik. Apabila dihitung secara manual jadwal pemberangkatan kapal tersebut sangat rumit dan tentunya membutuhkan suatu ketelitian dan waktu yang cukup lama.

Permasalahan terjadi apabila pegawai membuat jadwal dengan tidak teliti yaitu ada nama kapal yang sama dalam sehari yang beroperasi. Oleh karena itu, untuk mengatasi permasalahan tersebut diperlukan suatu penjadwalan pemberangkatan kapal menggunakan sistem komputerisasi guna mempercepat pengaturan jadwal pemberangkatan kapal agar efektif dan efisien. Penggunaan sistem komputerisasi akan mempermudah pengaturan jadwal pemberangkatan kapal yang ada di Pelabuhan Ketapang, Banyuwangi.

Perancangan penjadwalan yang baik dapat menghasilkan solusi yang mendekati optimal dalam waktu yang singkat. Penggunaan metode heuristik dalam memperoleh solusi penjadwalan yang lebih baik dapat digunakan sebagai penyelesaian permasalahan tersebut. Algoritma genetika merupakan cabang dari algoritma evolusi yang terkenal sebagai sebuah solusi permasalahan pengoptimalan yang tidak mudah untuk dipecahkan pada berbagai bidang biologi, sosiologi, ekonomi, fisika, dan lain lain (Mahmudy, 2013).

Penerapan algoritma genetika telah dibahas pada penelitian sebelumnya. Penelitian Ilmi dkk. (2015) membahas tentang permasalahan penjadwalan perawat menggunakan metode algoritma genetika. Permasalahan tersebut menggunakan representasi permutasi bilangan integer, menggunakan metode crossover yaitu one cut-point crossover, menggunakan metode mutasi yaitu reciprocal exchange mutation, dan menggunakan metode seleksi yaitu elitism selection. Dari pengujian permasalahan tersebut didapat kromosom terbaik dengan nilai fitness tertinggi dengan hasil akhirnya didapat sebuah jadwal jaga perawat selama 1 bulan di ruang ICU.

Penulis juga menemukan penelitian yang relevan yaitu pada penelitian Zulfa dkk. (2015) yang menerapkan algoritma genetika dalam permasalahan penyusunan jadwal mengajar asisten laboratorium menggunakan metode algoritma genetika. Permasalahan tersebut timbul karena adanya jadwal yang bentrok antara jadwal kegiatan asisten dengan jadwal mengajar asisten dan proses penyusunan jadwal yang membutuhkan waktu lama apabila dilakukan secara manual. Permasalahan tersebut menggunakan representasi permutasi berbasis kode asisten, menggunakan metode crossover yaitu one cut-point crossover, menggunakan metode mutasi yaitu reciprocal exchange mutation, dan

menggunakan metode seleksi yaitu *elitism selection*. Dengan mengacu pada penelitian yang telah dilakukan tersebut, algoritma genetika telah terbukti berhasil dalam mengatasi berbagai permasalahan, sehingga pada skripsi ini menggunakan algoritma genetika dalam optimasi permasalahan penjadwalan kapal.

Berdasarkan permasalahan dan metode yang telah dijelaskan tersebut, maka penulis tertarik untuk meneliti dan mengkaji lebih lanjut mengenai Optimasi Penjadwalan Kapal Penyeberangan Ketapang – Gilimanuk Menggunakan Algoritma Genetika yang akan dikembangkan dalam skripsi ini.

1.2 Rumusan masalah

Berdasarkan uraian latar belakang yang telah dijelaskan, penulis merumuskan bahwa permasalahan yang akan diselesaikan adalah sebagai berikut.

- 1. Bagaimana menerapkan algoritma genetika dalam permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk?
- Bagaimana menentukan nilai parameter yang tepat untuk mendapatkan solusi yang optimal dalam permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika?
- 3. Bagaimana hasil dari parameter terbaik dalam permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk menggunakan algoritma genetika?

1.3 Tujuan

Berdasarkan uraian latar belakang dan rumusan masalah yang telah dijelaskan, penulis merumuskan bahwa tujuan yang akan dicapai adalah sebagai berikut.

- 1. Untuk menerapkan algoritma genetika sebagai solusi dalam permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk.
- 2. Untuk menentukan nilai parameter yang tepat sehingga mendapatkan solusi yang optimal dalam permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk menggunakan algoritma genetika.
- 3. Untuk mendapatkan hasil yang optimal dari parameter terbaik dalam permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk menggunakan algoritma genetika.

1.4 Manfaat

Berdasarkan uraian latar belakang, rumusan masalah, dan tujuan yang telah dijabarkan, penulis mengharapkan manfaat sebagai berikut.

- Sistem menggunakan algoritma genetika dapat menjadikan sebuah solusi dalam permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.
- Hasil dari sistem tersebut dapat meningkatkan efisiensi dan efektifitas dalam penyusunan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

3. Hasil dari sistem tersebut dapat memberikan kontribusi kepada PT ASDP Cabang Ketapang dalam pertimbangan penyusunan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

1.5 Batasan masalah

Adapun batasan masalah yang ada pada penelitian penulis adalah sebagai berikut.

- 1. Data penjadwalan kapal penyeberangan Ketapang Gilimanuk yang digunakan dalam penelitian adalah diperoleh dari Kantor PT ASDP Indonesia Ferry Cabang Ketapang, Kabupaten Banyuwangi dan Kantor Otoritas Pelabuhan Penyeberangan Ketapang Gilimanuk.
- 2. Menggunakan data kapal dermaga *Landing Craft Machine* (LCM), dengan total keseluruhan yaitu 29 kapal.
- 3. Membuat jadwal selama 15 hari.
- 4. Penjadwalan kapal disesuaikan dengan kapal yang beroperasi dengan menggunakan pola operasi 12 kapal.
- 5. Parameter yang digunakan dari algoritma genetika adalah ukuran populasi, banyaknya generasi, crossover rate (cr), dan mutation rate (mr).

1.6 Sistematika pembahasan

Dalam penyusunan skripsi ini, sistematika pembahasan yang dilakukan terbagi menjadi tujuh bab yang secara singkat dijelaskan sebagai berikut.

BAB 1: Pendahuluan

Pada bab ini berisi mengenai latar belakang dari penyusunan skripsi, rumusan masalah, tujuan, manfaat yang akan diperoleh, batasan masalah, serta sistematika pembahasan mengenai penulisan skripsi.

BAB 2 : Landasan Kepustakaan

Pada bab ini berisi mengenai kajian pustaka yang sesuai dengan permasalahan skripsi, konsep dasar, teori yang berkaitan tentang algoritma genetika mengenai permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

BAB 3: Metodologi

Pada bab ini berisi mengenai penjelasan metodologi yang digunakan dalam penelitian, yaitu tahapan atau langkah-langkah yang dilakukan pada saat penelitian, yang dimulai dari studi pustaka, pengumpulan data, pengolahan data, implementasi sistem, pengujian sistem, dan evaluasi sistem, serta penjelasan mengenai algoritma yang digunakan, kebutuhan sistem, dan pengujian algoritma genetika.

BAB 4: Perancangan

Pada bab ini berisi mengenai penjelasan formulasi permasalahan penjadwalan kapal, siklus algoritma, siklus penyelesaian menggunakan algoritma genetika dalam permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk dan perancangan antar muka atau *user interface*.

BAB 5 : Implementasi

Pada bab ini berisi mengenai penjelasan implementasi sistem dari perancangan yang telah dibahas sebelumnya yaitu terdiri dari implementasi antarmuka atau *user interface* dan implementasi algoritma genetika dalam permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk yaitu berupa struktur *class* dan potongan *source code* utama.

BAB 6 : Pengujian dan Pembahasan

Pada bab ini berisi mengenai tabel dan grafik hasil, serta analisis pengujian menggunakan parameter algoritma genetika yang telah ditentukan sesuai dengan permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika.

BAB 7: Penutup

Pada bab akhir ini berisi mengenai kesimpulan dan saran dari hasil penelitian serta pengujian yang telah dilakukan sesuai dengan rumusan masalah yang telah ditentukan diawal mengenai permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini membahas tentang kajian pustaka yang sesuai dengan permasalahan skripsi, konsep dasar, teori yang berkaitan tentang algoritma genetika mengenai permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

2.1 Kajian Pustaka

Berdasarkan judul skripsi, penyusunan kajian pustaka digunakan untuk menemukan dan membandingkan penelitian yang sesuai guna mendukung penyusunan skripsi ini. Beberapa penelitian yang telah dilakukan, yang sesuai dengan judul dan rumusan masalah yaitu pada penelitian yang dilakukan oleh Sari dkk. (2015). Penelitian tersebut menggunakan pendekatan algoritma genetika dalam permasalahan penjadwalan mata pelajaran studi kasus SMPN 1 Gondang Mojokerto. Peneliti tersebut menggunakan representasi permutasi berbasis kode guru. Adapun metode *crossover* yang digunakan adalah *one cut-point crossover*, metode mutasinya menggunakan *reciprocal exchange mutation* serta metode seleksinya menggunakan *elitism selection*. Hasil penjadwalan mata pelajaran tersebut diambil dengan kromosom terbaik yaitu dengan nilai *fitness* tertinggi.

Kemudian pada penelitian Devi dkk. (2015) yaitu dengan judul penerapan algoritma genetika untuk penjadwalan asisten praktikum. Peneliti tersebut berhasil menerapkan algoritma genetika dalam permasalahan penjadwalan pelaksanaan praktikum untuk masing-masing kelas yang telah terdaftar sebagai praktikan. Penelitian tersebut menggunakan 52 kromosom, ukuran populasi 50, ukuran generasi 30, serta kombinasi *crossover rate* (*cr*)= 0,5 dan *mutation rate* (*mr*)= 0,5. Hasil akhir yang didapat yaitu kromosom terbaik dengan waktu optimal dan jumlah pinalti yang rendah. Berikut Tabel 2.1 yang menjelaskan tentang penelitian yang telah dilakukan sebelumnya.

Tabel 2.1 Penelitian yang Telah Dilakukan Sebelumnya

No.	Judul	Permasalahan	Metode	Hasil
1.	Optimasi Penjadwalan	Membuat penjadwalan	Algoritma genetika.	Panjang kromosom 360,
	Perawat Menggunakan Algoritma Genetika oleh	perawat secara adil.	Crossover: one cut point crossover,	Jumlah generasi 150, Jumlah populasi
	(Ilmi, et al., 2015).		Mutasi: reciprocal exchange mutation,	200, Kombinasi <i>cr</i> dan <i>mr</i> : 0,5; 0,5,
	BRAW	WAYA	Seleksi: elitism selection.	Jadwal perawat selama 1 bulan.

2.	Optimasi Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika (Studi Kasus: SMP 1 Gondang Mojokerto) oleh (Sari, et al., 2015).	Masih dibuat secara manual sehingga membutuhkan waktu yang lama dan hasil yang didapat tidak akurat.	Algoritma genetika. Crossover: one cut point crossover, Mutasi: reciprocal exchange mutation, Seleksi: elitism selection.	Panjang kromosom 552, Jumlah generasi 80, Jumlah populasi 60, Kombinasi cr dan mr: 0,4; 0,6, Kromosom terbaik dengan nilai fitness tertinggi.
3.	Optimasi Jadwal Mengajar Asisten Laboratorium Menggunakan Algoritma Genetika oleh (Zulfa, et al., 2015).	Masih dibuat secara manual dengan banyak permasalahan satunya yaitu adanya waktu yang bentrok dan proses yang lama.	Algoritma genetika. Crossover: one cut point crossover, Mutasi: reciprocal exchange mutation, Seleksi: elitism selection.	Panjang kromosom 360 Jumlah generasi 140 Jumlah populasi 70, Kombinasi cr dan mr: 0,2; 0,8 Kromosom terbaik dengan waktu optimal dan fitness tertinggi.
4.	Penerapan Algoritma Genetika untuk Penjadwalan Asisten Praktikum oleh (Devi, et al., 2015).	Masih dibuat secara manual dengan permasalahan waktu yang bentrok antara jadwal kuliah dengan jadwal mengajar asisten praktikum, serta membutuhkan waktu yang lama.	Algoritma genetika. Crossover: one cut point crossover Mutasi: reciprocal exchange mutation Seleksi: elitism selection	Panjang kromosom 52, Jumlah generasi 30, Jumlah populasi 50, Kombinasi cr dan mr: 0,5; 0,5. Kromosom terbaik dengan waktu optimal dan jumlah pinalti yang rendah.

Sumber: Diolah oleh penulis

2.2 Penjadwalan Kapal

Menurut Wismulyani (2008) kapal adalah suatu alat transportasi yang sering digunakan masyarakat Indonesia untuk dapat saling berhubungan dari satu pulau ke pulau yang lain di seluruh wilayah Indonesia karena Indonesia itu sendiri berbentuk sebagai negara kepulauan yang terdiri dari beberapa pulau. Perahu dan kapal merupakan alat transportasi yang utama di negara Indonesia. Kapal memiliki bentuk, fungsi, dan warna yang beraneka ragam.

Secara umum, pengertian penjadwalan adalah "Scheduling is the allocation of resources overtime to perform collection of risk" yang artinya penjadwalan adalah suatu pengalokasian sumber daya terbatas yang digunakan untuk mengerjakan sekumpulan pekerjaan (Ginting, 2009). Penjadwalan sangatlah penting agar suatu kegiatan dapat berjalan dengan lancar.

Salah satu hal penting dalam proses pemberangkatan kapal adalah penyusunan penjadwalan kapal. Penjadwalan ini dilakukan dengan suatu pengaturan dan penjadwalan agar pemberangkatan kapal tersebut sesuai dengan apa yang diharapkan agar dapat berjalan secara optimal. Untuk mencapai tujuan tersebut salah satu usaha yang dilakukan adalah dengan melakukan penjadwalan yang terencana dengan baik.

Dalam buku Rosnani Ginting (2009), pengertian penjadwalan (scheduling) menurut Conway adalah suatu pengurutan secara menyeluruh yang dikerjaan oleh beberapa mesin yang bekerja. Sedangkan menurut Kenneth R. Baker, pengertian penjadwalan adalah suatu proses pengalokasian sumber daya dalam jangka waktu tertentu dengan tujuan untuk memilih beberapa tugas dari sekumpulan tugas.

Permasalahan penjadwalan sebenarnya merupakan suatu permasalahan pengalokasian dengan bantuan model matematis yang dapat ditentukan dengan solusi optimal. Suatu model penjadwalan memberikan rumusan masalah yang sistematik dengan solusi yang diharapkan (Ginting, 2009).

2.2.1 Tujuan Penjadwalan

Adapun beberapa tujuan penjadwalan yang diidentifikasikan oleh Bedworth dalam buku Rosnani Ginting (2009) adalah sebagai berikut.

- Dapat meningkatkan penggunaan sumberdaya yang dapat mengurangi waktu tunggu dalam proses penjadwalan sehingga total waktu prosesnya menjadi berkurang dan kegiatan produksi meningkat.
- 2. Mengurangi sejumlah pekerjaan yang masuk dalam barisan antrian ketika ada sumber daya melakukan sejumlah pekerjaan lain.
- 3. Mengurangi keterlambatan dalam pekerjaan yang memiliki batas waktu penyelesaian agar dapat mengurangi *penalty cost* (biaya keterlambatan).
- 4. Membantu dalam pengambilan keputusan tentang perencanaan sehingga dapat menghindari biaya yang mahal apabila terjadi suatu penambahan.

2.3 Algoritma Genetika

Algoritma genetika adalah suatu algoritma pencarian heuristik yang digunakan untuk menyelesaikan masalah optimasi yang didasarkan pada seleksi alam dan genetik (Ginting, 2009). Untuk pertama kalinya, algoritma tersebut diperkenalkan oleh seorang mahasiswa dari Universitas Michigan, Amerika Serikat yang bernama John Holland pada tahun 1975. Menurutnya, suatu masalah berbentuk adaptasi (alami maupun buatan) dapat diselesaikan dengan menggunakan algoritma genetika (Desiani & Arhami, 2005).

Sifat dari algoritma genetika yaitu mencari kemungkinan dari calon solusi-solusi yang ada dengan tujuan untuk mendapatkan penyelesaian masalah secara optimal (Desiani & Arhami, 2005). Dimana algoritma tersebut mengkombinasikan secara *random* siapa individu yang terkuat yang ada pada suatu populasi dan dilakukan secara berulang-ulang dari setiap generasi ke generasi dengan menggunakan tiga operator dasar, yaitu *crossover*, mutasi, dan seleksi (Berlianty & Arifin, 2010).

Algoritma genetika dapat diterapkan pada berbagai permasalahan yaitu antara lain permasalahan optimasi penugasan mengajar bagi dosen (Mahmudy 2006), penjadwalan dan alokasi untuk ruang ujian (Mawaddah & Mahmudy 2006), permasalahan optimasi mengenai penjadwalan kuliah (Liliana & Mahmudy 2006), dan lain-lain. Pada umumnya, penggunaan metode algoritma gentika dapat memberikan solusi yang baik (Zukhri, 2014).

Genetik Algorithms (Algoritma Genetik), Evolutionary Programming, dan Evolution Strategies merupakan tiga bidang utama dalam penyelesaian permasalahan optimasi. Diantara ketiga bidang utama tersebut, algoritma genetika merupakan algoritma yang sering atau lebih banyak digunakan oleh peneliti dalam teknik optimasi (Berlianty & Arifin, 2010).

Algoritma genetika banyak diambil dari ilmu genetika (Mahmudy, 2013). Algoritma tersebut terinspirasi pada prinsip genetika dan seleksi alam yaitu pada teori evolusi Darwin. Algoritma genetika merupakan algoritma yang tidak berdasarkan matematika, akan tetapi berdasarkan pada fenomena alam yang ada pada ilmu genetika dan teori Darwin tentang "survival of the fittest" untuk mencari titik optimal (Berlianty & Arifin, 2010). Menurut teori evolusi Darwin tersebut bahwa terjadi persaingan antar individu dengan tujuan untuk memperebutkan sumberdaya dan siapa yang lemah akan mengalami kepunahan, sedangkan yang kuat itu yang akan menang (Sutojo et al., 2011). Individu tersebut secara terus menerus akan mengalami perubahan gen dengan tujuan agar dapat menyesuaikan dengan lingkungan yang ditempati (Ginting, 2011).

Menurut Sutojo, et al. (2011) adapun ciri-ciri yang dapat diselesaikan dengan permasalahan menggunakan algoritma genetika adalah sebagai berikut.

- 1. Memiliki ruang yang cukup besar, kompleks, dan sulit untuk dimengerti.
- 2. Tidak ada pengetahuan yang khusus untuk menyederhanakan ruang yang besar menjadi sempit.
- 3. Tidak ada analisis matematis apabila gagal dalam menyelesaikan permasalahan.

- 4. Solusinya tidak harus optimal, namun sudah masuk dalam kriteria yang harus dipenuhi.
- 5. Memiliki solusi yang tidak hanya satu, namun banyak solusi yang dapat diselesaikan.
- 6. Membutuhkan solusi yang "real time" atau solusi yang didapat dengan cepat agar efisien dalam mengiplementasi suatu permasalahan yang memiliki perubahan yang cepat.

2.3.1 Kelebihan dan Kekurangan Algoritma Genetika

Menurut Berlianty & Arifin (2010), beberapa hal mengenai kelebihan menggunakan algoritma genetika adalah sebagai berikut.

- 1. Algoritma genetika tidak menggunakan hasil dari manipulasi nilai parameter, melainkan dengan menggunakan manipulasi kode-kode set parameter algoritma.
- 2. Algoritma genetika tidak memiliki batasan metode yang lain maksudnya algoritma tersebut bebas untuk menggunakan kode masalah dengan berbagai cara.
- 3. Algoritma genetika tidak menggunakan satu titik saja, melainkan menggunakan populasi titik.
- 4. Algoritma genetika tidak menggunakan informasi turunan, melainkan menggunakan informasi fungsi tujuan.
- 5. Algoritma genetika tidak menggunakan aturan deterministik, melainkan menggunakan aturan probabilistik.
- 6. Algoritma genetika melakukan iterasi secara berulang ulang dan relatif banyak sehingga perlu diselesaikan dalam suatu sistem program aplikasi komputer untuk menyelesaikan suatu permasalahan tersebut.

Serta beberapa hal mengenai kekurangan menggunakan algoritma genetika adalah sebagai berikut (Berlianty & Arifin, 2010).

- Algoritma genetika menggunakan bilangan yang random atau acak, sehingga memungkinkan kromosom yang baik tidak ikut dalam proses tersebut.
- Algoritma genetika menggunakan bilangan random atau acak dalam setiap pemilihan kromosom baik digunakan dalam proses mutasi maupun crossover.
- 3. Algoritma genetika belum tentu menghasilkan solusi yang optimal karena tergantung bilangan yang *random* atau acak yang digunakan.

2.4 Komponen-Komponen Utama Algoritma Genetika

Pada algoritma genetika, terdapat beberapa komponen-komponen utama dalam penerapannya yaitu antara lain generasi awal, representasi kromosom, nilai fitness, crossover, mutasi, seleksi, dan kondisi berhenti (Termination Condition).

2.4.1 Generasi Awal

Pada generasi awal ini yaitu dengan membuat sejumlah populasi awal sesuai dengan permasalahan yang akan diselesaikan. Permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan bilangan secara acak sesuai dengan panjang interval dan porsi yang telah ditentukan.

2.4.2 Representasi Kromosom

Dalam algoritma genetika terdapat beberapa representasi kromosom yaitu antara lain representasi biner, representasi bilangan bulat, representasi bilangan riil, dan representasi permutasi. Menurut Syarif (2014), representasi permutasi digunakan pada permasalahan *Travelling Salesman Problem* (Persoalan perjalanan salesman), *Vehicle Routing Problem* (Persoalan rute kendaraan), *Shortest Path Problem* (Persoalan lintasan terpendek), *Scheduling Problem* (Persoalan penjadwalan), dan lain-lain.

Peneliti menggunakan representasi kromosom permutasi berbasis kode kapal pada permasalahan optimasi penjadwalan kapal Ketapang – Gilimanuk dengan menggunakan algoritma yang nantinya dijadikan sebagai calon solusi. Dimana pada kode tersebut mewakili sebuah kromosom dalam penjadwalan kapal yang sesuai dengan urutan angka dalam kromosomnya. Pada Gambar 2.1 merupakan contoh kromosom menggunakan representasi permutasi (Syarif, 2014) adalah sebagai berikut.

		5	2	3	6 15	4	9	7	6	8
--	--	---	---	---	-------------	---	---	---	---	---

Gambar 2.1 Contoh Kromosom Menggunakan Representasi Permutasi

2.4.3 Nilai Fitness

Tujuan dari penerapan algoritma genetika adalah untuk memaksimalkan nilai fitness (Sutojo, dkk 2011). Fungsi dari fitness tersebut adalah sebagai alat ukur untuk mengetahui tingkat kinerja suatu kromosom pada sebuah populasi dengan tujuan agar dapat bertahan hidup pada generasi selanjutnya (Desiani & Arhami, 2005). Semakin besar nilai fitness pada sebuah kromosom maka kemungkinan terpilih dan bertahan hidup pada generasi selanjutnya juga akan semakin besar. Di lingkungan nyata, individu yang memiliki nilai fitness tertinggi akan bertahan hidup, sedangkan individu yang memiliki nilai fitness terendah akan mati (Suyanto, 2014).

Setelah individu dalam populasi telah terbentuk, lalu langkah selanjutnya adalah menghitung nilai *fitness*-nya pada masing-masing individu tersebut. Nilai *fitness* dihitung dengan memberikan pinalti pada setiap aturan penjadwalan yang digunakan. Berikut merupakan aturan perhitungan fungsi *fitness* adalah sebagai berikut.

$$Fitness = \frac{1}{1 + pinalti}, \text{ dimana pinalti} = \sum Bp \sum Np$$
 (2.1)

Nilai fitness ditentukan dari nilai pinalti yang dihasilkan. Nilai pinalti tersebut merupakan jumlah pelanggaran terhadap hard costraint maupun soft constraint

yang ada pada kromosom. Apabila dalam waktu sehari ada nama kapal yang muncul lebih dari satu nama kapal yang sama, maka akan menghasilkan hard constraint dengan nilai pinalti yang dihasilkan adalah 1. Jadi fungsi fitness-nya.

$$Fitness = \frac{1}{1 + \sum Bp \sum Np}, \tag{2.2}$$

Keterangan:

∑Bp adalah bobot pelanggaran

∑Np adalah indikator pelanggaran

Semakin tinggi nilai *fitness* yang dihasilkan maka akan semakin baik solusi yang didapatkan, begitu juga sebaliknya. Semakin kecil nilai *fitness* yang dihasilkan maka akan semakin buruk solusi yang didapatkan. Nilai pinalti ini berbanding terbalik dengan nilai *fitness*, semakin kecil nilai pinaltinya, maka semakin besar nilai *fitness* yang dihasilkan.

2.4.4 Crossover

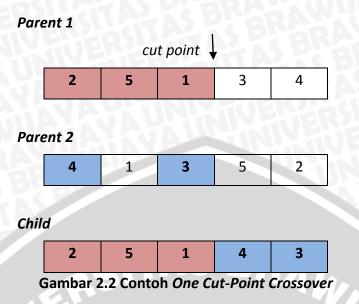
Metode *crossover* ini dilakukan dengan memilih dua kromosom induk (*parent*) secara acak dan dilakukan persilangan, kemudian menghasilkan kromosom anak (*offspring*). Tujuan dari *crossover* ini adalah untuk menambah keanekaragaman individu dalam suatu populasi (Desiani & Arhami, 2005).

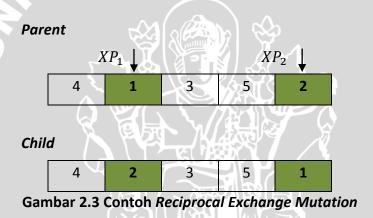
Metode *crossover* yang sering digunakan adalah *crossover* satu titik (one cutpoint crossover) (Kusumadewi, 2003). Metode *crossover* tersebut dilakukan dengan memilih satu titik potong secara acak dan memindahkan silang dari hasil titik potong tersebut (Ginting, 2011). Gambar 2.2 merupakan contoh one cut-point crossover (Mahmudy, 2013).

2.4.5 Mutasi

Mutasi adalah suatu proses pengubahan nilai dari satu atau lebih gen dalam suatu kromosom (Desiani & Arhami, 2005). Tujuan dari mutasi ini adalah untuk menghasilkan kromosom baru agar mendapatkan nilai *fitness* yang lebih baik dan nantinya akan menghasilkan solusi yang optimal. Proses mutasi ini bersifat *random* sehingga tidak menjamin proses mutasi akan selalu mendapatkan solusi yang optimal dan nilai *fitness* yang lebih baik, akan tetapi dalam penerapannya mendapat kontraversi karena sifatnya yang *random* tersebut dapat mengganggu kromosom baik sehingga tetap digunakan dengan probabilitas rendah.

Cara mutasi yaitu dengan memilih satu titik *random* yang nanti akan mengubah nilai gen pada titik tersebut, memilih satu induk secara *random* dari suatu populasi dengan mengubah nilai gen yang terpilih tersebut. Metode mutasi yang digunakan tersebut adalah *random mutation* (Mahmudy, 2013). Mutasi tersebut dilakukan pada seluruh gen yang ada pada suatu individu (Ginting, 2011). Berikut Gambar 2.3 yang merupakan contoh mutasi paling sederhana yaitu *reciprocal exchange mutation* yaitu secara *random* memilih dua posisi (*exchange point/XP*) dan menukarkan nilai pada posisi yang telah ditentukan (Mahmudy, 2013).





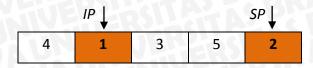
Metode mutasi lain yaitu *insertion mutation* yaitu metode yang dilakukan secara *random* dengan memilih satu posisi (*selected point/SP*) lalu mengambil dan menyisipkan angka tersebut ke posisi yang lain (*insertion point/IP*). Berikut Gambar 2.4 yang merupakan contoh *insertion mutation* (Mahmudy, 2013).

2.4.6 Seleksi

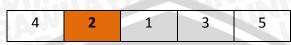
Seleksi adalah suatu proses pemilihan kromosom terbaik dalam suatu populasi sehingga mendapat peluang yang besar untuk dapat bertahan hidup, sedangkan yang kurang baik mendapat peluang yang kecil dan akan hilang (Desiani & Arhami, 2005). Tujuan dari seleksi ini adalah memberikan kesempatan yang besar kepada individu yang paling fit untuk melakukan reproduksi (Kusumadewi, 2003).

13





Child



Gambar 2.4 Contoh Insertion Mutation

Metode seleksi yang sering digunakan pada algoritma genetika adalah roulette wheel, binary tournament, dan elitism (Mahmudy, 2013). Metode elitism merupakan metode dengan memilih individu terbaik dari populasi dengan mengumpulkan beberapa individu dari populasi (parent) dan offspring dalam suatu wadah. Nantinya individu terbaik akan terpilih dan akan masuk pada generasi selanjutnya.

Adapun kelemahan menggunakan metode seleksi *elitism* yaitu tidak memberikan kesempatan pada nilai *fitness* yang kecil untuk bereproduksi. Namun menurut Syarif (2014) metode *elitism* sangatlah efektif dalam berbagai permasalahan karena dengan metode tersebut dapat terhindar dari kehilangan kromosom yang terbaik.

2.4.7 Kondisi Berhenti (Termination Condition)

Setelah melalui proses-proses maka akan menghasilkan suatu kromosom baru yaitu offspring. Offspring tersebut dihitung pada nilai fitness-nya apakah sudah optimal atau belum, apabila belum maka akan diseleksi terus sampai terpenuhi kriteria kondisi untuk berhenti. Adapun kriteria kondisi berhenti (Termination Condition) yang sering digunakan menurut Syarif (2014) adalah sebagai berikut.

- Iteriasi akan berhenti sampai pada generasi tertentu (maximum generation),
- 2. Iterasi akan berhenti setelah dilakukan percobaan pada beberapa generasi secara berurutan tidak mendapatkan nilai *fitness* yang berubah,
- 3. Iterasi akan berhenti setelah dilakukan percobaan pada beberapa generasi kemudian pada generasi selanjutnya tidak mendapatkan nilai *fitness* yang lebih baik.

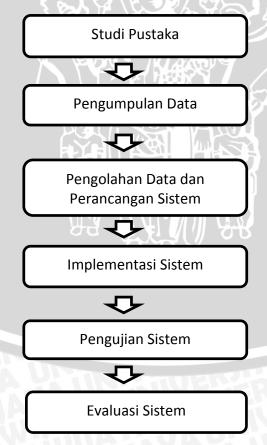
BAB 3 METODOLOGI

Bab ini membahas tentang metodologi yang digunakan dalam penelitian, yaitu tahapan atau langkah-langkah yang dilakukan pada saat penelitian, yang dimulai dari studi pustaka, pengumpulan data, pengolahan data, implementasi sistem, pengujian sistem, dan evaluasi sistem, serta penjelasan mengenai algoritma yang digunakan, kebutuhan sistem, dan pengujian algoritma genetika.

Skripsi ini dapat dikategorikan sebagai penelitian implementatif dengan pendekatan perancangan (design), yang nantinya dari penelitian tersebut akan menghasilkan sebuah sistem yaitu berupa perangkat lunak (software) yang dapat digunakan dalam penyusunan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

3.1 Tahapan Penelitian

Dalam penelitian pembuatan sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan beberapa metode penelitian yang meliputi urutan langkah-langkah yang harus dilakukan dalam menyelesaikan suatu permasalahan. Adapun diagram blok tahapan metode penelitian dalam skripsi ini, ditunjukkan pada Gambar 3.1 adalah sebagai berikut.



Gambar 3.1 Diagram Blok Tahapan Penelitian

3.1.1 Studi Pustaka

Dalam tahapan penelitian ini pertama yaitu studi pustaka yang dilakukan dengan mencari, mendapatkan, dan mengumpulkan dasar-dasar teori dan sumber acuan mengenai metode algoritma genetika dan penjadwalan kapal penyeberangan Ketapang — Gilimanuk yang nantinya dapat dipelajari mengenai informasi, pengetahuan, dan teori yang terkait tentang pembuatan sistem agar dari hasil penelitian, dapat menyelesaikan penyusunan penjadwalan kapal. Peneliti melakukan pencarian referensi di perpustakaan, ruang baca, toko buku, jurnal, internet, penjelasan dari dosen pembimbing dan teman-teman mahasiswa, dan lain-lain yang dapat membantu dalam penyelesaian permasalahan skripsi ini.

3.1.2 Pengumpulan Data

Pengumpulan data merupakan tahapan yang penting dalam proses penelitian karena dengan mendapatkan data yang tepat maka penelitian dapat berlangsung sesuai dengan perumusan masalah yang telah ditentukan dari awal. Proses pengumpulan data yang dilakukan yaitu dengan melakukan observasi dan wawancara. Tahap untuk memperoleh data dengan melakukan observasi yaitu datang secara langsung di Kantor PT ASDP Indonesia Ferry Cabang Ketapang, Kabupaten Banyuwangi dan Kantor Otoritas Pelabuhan Penyeberangan Ketapang – Gilimanuk. Kemudian melakukan wawancara ke beberapa pegawai yang ada di kantor tesebut tentang penyusunan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

3.1.3 Pengolahan Data dan Perancangan Sistem

Tahap studi pustaka dan pengumpulan data telah dilakukan, maka tahap selanjutnya adalah mengolah data yang telah didapat dan melakukan perancangan sistem. Tahap perancangan sistem ini bertujuan untuk menganalisis kebutuhan apa saja yang dibutuhkan dalam pembuatan sistem optimasi penjadwalan kapal menggunakan algoritma genetika untuk mempermudah penulis dalam melakukan tahap selanjutnya dan agar menghasilkan sistem yang optimal.

3.1.4 Implementasi Sistem

Tahap implementasi sistem ini yaitu dengan menerapkan algoritma genetika dalam permasalahan pengoptimalan penjadwalan kapal penyeberangan Ketapang – Gilimanuk berdasarkan perancangan sistem yang telah dirancang. Dalam implementasi sistem ini, menggunakan bahasa pemrograman Java, pembuatan antarmuka, dan nama kapal yang diperoleh dari Kantor PT ASDP Indonesia Ferry Cabang Ketapang dan Kantor Otoritas Pelabuhan Penyeberangan Ketapang – Gilimanuk.

3.1.5 Pengujian Sistem

Pada tahap pengujian sistem ini penulis menguji sistem sesuai dengan parameter algoritma genetika, yaitu antara lain pengujian ukuran populasi,

banyaknya generasi, serta kombinasi *crossover rate (cr)* dan *mutation rate (mr)*. Pengujian dilakukan agar dapat membuktikan bahwa hasil sistem berupa perangkat lunak (*software*) tersebut telah mampu bekerja dengan baik sesuai dengan kebutuhan dan mengetahui seberapa baik solusi yang didapatkan dalam penerapan algoritma genetika dalam menyelesaikan permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk.

3.1.6 Evaluasi Sistem

Langkah terakhir yaitu tahap evaluasi sistem yaitu dengan mengevaluasi hasil pengujian menggunakan algoritma algoritma genetika dalam penyusunan penjadwalan kapal penyeberangan Ketapang — Gilimanuk. Evaluasi sistem yang dilakukan yaitu melakukan pengujian kembali mengenai hasil dari parameter terbaik yang telah dilakukan sebelumnya. Tujuan dari evaluasi sistem tersebut adalah untuk mengetahui solusi terbaik dalam penyusunan penjadwalan kapal penyeberangan dan mengukur kualitas solusi yang didapat dari penerapan algoritma genetika, serta mengetahui apa saja kekurangan sementara yang ada dalam sistem tersebut. Oleh karena itu, nantinya akan diketahui apakah sistem tersebut telah sesuai dengan tujuan awal sistem.

3.2 Teknik Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah data jadwal operasi kapal yang ada di Pelabuhan Ketapang, Banyuwangi di dermaga Landing Craft Machine (LCM). Data tersebut diperoleh dari Kantor PT ASDP Indonesia Ferry Cabang Ketapang, Kabupaten Banyuwangi dan Kantor Otoritas Pelabuhan Penyeberangan Ketapang – Gilimanuk. Dari data tersebut nantinya akan dihasilkan keluaran berupa penjadwalan kapal penyeberangan Ketapang – Gilimanuk yang beroperasi selama 15 hari.

3.3 Algoritma yang Digunakan

Penelitian mengenai sistem optimasi penjadwalan menggunakan algoritma genetika seperti yang telah ada dalam penelitian sebelumnya pada kajian pustaka, telah terbukti efektif dan berhasil dalam permasalahan penyusunan penjadwalan. Implementasi permasalahan menggunakan bahasa pemrograman Java karena merupakan bahasa *multithreading* yaitu dapat membuat program yang dapat melakukan beberapa tugas pada waktu yang bersamaan (Ramadhani, 2015) dan telah banyak digunakan dalam berbagai permasalahan menggunakan metode algoritma genetika menggunakan bahasa pemrograman Java yaitu pada permasalahan mengenai penjadwalan perawat yang telah dilakukan oleh Devi dkk. (2015), permasalahan penjadwalan karyawan *training* pada divisi *food* and *beverage* yang dilakukan oleh Girinata, Mahmudy (2016) dan lain-lain. Sistem optimasi penjadwalan kapal ini merupakan aplikasi berbasis aplikasi *desktop*, dan tentunya, Java juga dapat dijalankan pada sistem operasi atau *platform* apapun (Hariyanto, 2011).

3.4 Kebutuhan Sistem

Spesifikasi kebutuhan perangkat dalam pembuatan sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika agar dapat berjalan dengan baik dibutuhkan suatu sistem yaitu sebagai berikut.

- a. Spesifikasi Kebutuhan Perangkat Keras (Hardware) yaitu:
 - Prosesor Intel(R) Core(TM) i3-2330M CPU @2.20GHz 2.20 GHz.
 - Memori 2.00 GB.
 - Hardisk dengan kapasitas 500 GB.
- b. Spesifikasi Kebutuhan Perangkat Lunak (*Software*) yaitu:
 - Sistem operasi yang digunakan adalah menggunakan sistem Operasi Microsoft Windows 7.
 - Bahasa Pemrograman Java.
 - Aplikasi Netbeans.
- c. Spesifikasi Kebutuhan data yaitu:
 - Data penjadwalan kapal penyeberangan Ketapang Gilimanuk yang digunakan dalam penelitian adalah data yang diperoleh dari Kantor PT ASDP Indonesia Ferry Cabang Ketapang, Kabupaten Banyuwangi dan Kantor Otoritas Pelabuhan Penyeberangan Ketapang – Gilimanuk.
 - Menggunakan data kapal dermaga Landing Craft Machine (LCM), dengan total keseluruhan yaitu 29 kapal.

3.5 Pengujian Algoritma

Penerapan menggunakan metode algoritma genetika untuk proses optimasi penyusunan penjadwalan dilakukan pengujian parameter algoritma genetika yaitu sebagai berikut.

3.5.1 Pengujian Ukuran Populasi

Pada sub bab ini, pengujian algoritma dilakukan untuk mengetahui ukuran populasi yang paling optimal dalam penyelesaian permasalahan penjadwalan pemberangkatan kapal Ketapang — Gilimanuk menggunakan algoritma genetika. Pengujian populasi ini menggunakan ukuran populasi yaitu 20, 40, 60, 80, 100, 120, 140, 160, 180, dan 200 untuk rata-rata nilai *fitness*-nya dilakukan 10 kali percobaan dari hasil yang dapat digunakan untuk analisa. Adapun banyaknya generasi, dan kombinasi *crossover rate (cr) dan mutasi rate (mr)* didapat dari penelitian yang telah dilakukan. Berikut pada Tabel 3.1 perancangan pengujian ukuran populasi pada yaitu sebagai berikut.

Tabel 3.1 Perancangan Pengujian Ukuran Populasi

	UNIX	T		*		Nilai <i>f</i>	itnes	S			RI	Rata –
No.	Ukuran Populasi			Perc	obaar	ı ukur	ran po	opula	si ke-	46		rata Nilai
45		1	2	3	4	5	6	7	8	9	10	Fitness
1.	20						H			TO		453
2.	40											HITE
3.	60									1		
4.	80											
5.	100				FA	G						
6.	120		2					41	4	A		
7.	140									K		
8.	160										1	7,
9.	180			8	$\mathbf{A}(\mathbf{a})$							
10	200		7	4 }	5 ()			! /	1			

3.5.2 Pengujian Banyaknya generasi

Pada sub bab ini, pengujian algoritma dilakukan untuk mengetahui banyaknya generasi yang paling optimal dalam penyusunan penjadwalan pemberangkatan kapal Ketapang-Gilimanuk menggunakan algoritma genetika. Perancangan pengujian dilakukan dengan banyaknya generasi yaitu 20, 40, 60, 80, 100, 120, 140, 160, 180, 200 dan untuk rata-rata nilai fitness-nya dilakukan 10 kali percobaan. Adapun ukuran populasi didapatkan dari pengujian sebelumnya. Berikut Tabel 3.2 perancangan pengujian banyaknya generasi yaitu sebagai berikut.

Tabel 3.2 Perancangan Pengujian Banyaknya generasi

	UNIX	TIV	13	Ha	PS	Nilai j	fitnes	S			RL	Rata-
No.	Banyaknya generasi		P	ercok	aan l	oanya	knya	gene	rasi k	e-		rata Nilai
Add.		1	2	3	4	5	6	7	8	9	10	Fitness
1.	20		1				H	Z	A	di		453
2.	40								Y			111:
3.	60									9		
4.	80											
5.	100				· A	9	B	b				
6.	120		3					1	4	1		
7.	140									1		
8.	160										4	
9.	180											
10.	200		~	५ य				% ~	1			

3.5.3 Pengujian Kombinasi Crossover Rate (cr) dan Mutation Rate (mr)

Pada poin ini, pengujian algoritma dilakukan untuk mengetahui kombinasi dari crossover rate (cr) dan mution rate (mr) yang terbaik yang dapat menghasilkan solusi yang optimal. Berikut pada Tabel 3.3 perancangan pengujian kombinasi crossover rate (cr) dan mutation rate (mr) sebanyak 10 kali percobaan yaitu sebagai berikut.

Tabel 3.3 Perancangan Pengujian Kombinasi Crossover Rate dan Mutation Rate

Komk	oinasi	M	Nilai fitness									Rata-
Komik	Jillasi			Percol	baan l	kombi	nasi c	r dan	mr ke			rata Nilai
cr	mr	1	2	3	4	5	6	7	8	9	10	Fitness
1	0	41		4					X	TT	14	453
0,9	0,1								Ų.			HTTE
0,8	0,2		2									MIN
0,7	0,3											
0,6	0,4				7	ΛG		3 F				
0,5	0,5		Q	5	Ш				4	la		
0,4	0,6									Y		
0,3	0,7											7,
0,2	0,8					S						
0,1	0,9		-	M	K	3						
0	1								3	2		



BAB 4 PERANCANGAN

Bab ini membahas tentang formulasi permasalahan penjadwalan kapal, siklus algoritma, siklus penyelesaian menggunakan algoritma genetika dalam permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk dan perancangan antar muka atau *user interface*.

4.1 Formulasi Permasalahan

Pada sub ini akan membahas tentang permasalahan penjadwalan kapal penyeberangan yang akan diselesaikan. Permasalahan dalam penelitian ini adalah untuk mengoptimasi penjadwalan kapal penyeberangan Ketapang — Gilimanuk menggunakan algoritma genetika. Masalah tersebut timbul karena pembuatan penjadwalan kapal masih dibuat secara manual, sehingga membutuhkan waktu yang cukup lama, kurang adil dalam pembagian porsi masing-masing kapal, serta membutuhkan suatu ketelitian agar tidak terjadi kesalahan. Oleh karena itu, untuk mengatasi permasalah tersebut diperlukan adanya sistem optimasi penjadwalan kapal penyeberangan Ketapang — Gilimanuk menggunakan algoritma genetika. Data yang digunakan dalam permasalahan penjadwalan kapal penyeberangan Ketapang — Gilimanuk ini adalah data yang diperoleh dari PT ASDP Indonesia Ferry Cabang Ketapang dan Kantor Otoritas Pelabuhan Penyeberangan Ketapang — Gilimanuk. Penjadwalan kapal disesuaikan dengan kapal yang beroperasi. Pada Tabel 4.3 ini merupakan data kapal yang akan diselesaikan adalah sebagai berikut.

Tabel 4.1 Data Nama Kapal di Dermaga LCM

No.	Kode Kapal	Nama Kapal
1.	LAD	KMP LABITRA ADINDA
2.	JMB	LCT JAMBO
3.	SMS	LCT SMS SWAKARYA
4.	TELT	KMP TRISAKTI ELFINA
5.	PST	LCT PUTRI SRITANJUNG
6.	HER	LCT HERLIN IV
7.	TNP	LCT TUNU PRATAMA
8.	ARJ	LCT ARJUNA
9.	RAF	KMP RAFELIA
10.	DMF	KMP DHARMA FERRY
11.	PIN	LCT PANCAR INDAH

Sumber: Kantor PT ASDP Indonesia Ferry Cabang Ketapang (2015)

Jumlah kapal dimisalkan dalam penelitian perhitungan manual ini adalah 11 kapal yaitu yang terdapat pada Tabel 4.1 tersebut, dengan jumlah *trip* 8 kali dan

kapal yang beroperasi sebanyak 8 (2 kapal *off* dan 1 kapal *docking* yakni kapal LCT. Pancar Indah/PIN). Data waktu *trip* ditunjukkan pada Tabel 4.2 adalah sebagai berikut.

Tabel 4.2 Data Trip Kapal di Dermaga LCM

No.	Trip	Waktu				
1.	1	07.30				
2.	2	08.00				
3.	3	08.30				
4.	4	09.00				
5.	5	09.30				
6.	6	10.00				
7.	7	10.30				
8.	8	11.00				

Permasalahan penyusunan penjadwalan tersebut pembagiannya tidak adil karena banyaknya kapal dan jumlah dermaga yang terbatas. Pada Tabel 4.3 merupakan contoh penjadwalan kapal Ketapang – Gilimanuk yang dibuat secara manual.

Tabel 4.3 Contoh Jadwal Kapal Penyeberangan Ketapang – Gilimanuk

Hari		OFF							
1.	LAD	JMB	HER	SMS	TEL	TNP	DMF	PST	RAF, ARJ
2.	RAF	ARJ	TNP	SMS	PST	LAD	JMB	PST	TEL, HER
3.	TEL	HER	PST	JMB	DMF	RAF	ARJ	SMS	LAD, TNP
4.	LAD	TNP	JMB	SMS	PST	TEL	RAF	DMF	HER, ARJ
5.	HER	ARJ	DMF	RAF	HER	SMS	TEL	PST	LAD, JMB
6.	LAD	JMB	ARJ	RAF	PIN	PST	TEL	SMS	DMF, HER
7.	DMF	HER	JMB	LAD	ARJ	SMS	TEL	RAF	PST, TNP

Pada Tabel 4.3 tersebut merupakan contoh penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan pola operasi 8 kapal dari 11 kapal yang ada (2 kapal off dan 1 kapal docking yakni kapal LCT Pancar Indah/PIN).

Permasalahan muncul apabila penjadwalan dibuat secara manual yaitu apabila petugas tidak teliti, maka akan ada nama kapal yang sama muncul lebih dari satu kali dalam sehari seperti dalam contoh yakni kapal LCT Putri Sritanjung (PST) dan LCT Herlin IV (HER) yang muncul lebih dari sekali dalam sehari.

Kemudian permasalahan tidak adanya keadilan pada jadwal kapal tersebut. Ada beberapa kapal yang mendapat operasi dengan porsi operasi yang lebih banyak, sedangkan yang lain tidak memiliki porsi operasi yang sama. Contohnya kapal LCT SMS Swakarya (SMS) dan LCT Putri Sritanjung (PST) yang mendapat porsi operasi 7 kali, sedangkan kapal LCT Tunu Pratama (TNP) mendapat porsi operasi 3 kali. Tentu tidak adanya keadilan dalam pembagian jadwal kapal tersebut. Lalu apabila petugas tidak teliti memungkinkan memasukkan nama kapal yang *docking* pada jadwal kapal penyeberangan.

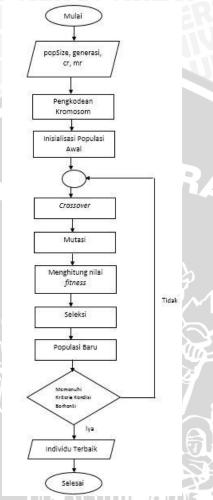
Berdasarkan permasalahan tersebut, penulis tertarik untuk meneliti dan mengkaji lebih lanjut mengenai sistem optimasi penjadwalan kapal Penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika yang akan dikembangkan dalam skripsi ini. Permasalahan penjadwalan yaitu dengan membuat jadwal pemberangkatan kapal penyeberangan secara adil dan efisien untuk memenuhi penjadwalan yang baik. Karena dengan menggunakan sistem komputerisasi akan mempermudah setiap pengaturan jadwal pemberangkatan kapal yang ada di Ketapang – Gilimanuk.

4.2 Siklus Algoritma

Adapun siklus algoritma dalam penyelesaian permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika adalah sebagai berikut.

- 1. Memasukkan parameter algoritma genetika yaitu ukuran populasi (popSize), banyaknya generasi (generasi), crossover rate (cr), dan mutation rate (mr).
- 2. Melakukan pengkodean kromosom dengan menggunakan representasi kromosom bilangan *integer* yang sesuai dengan porsi yang telah ditentukan.
- 3. Membangkitkan populasi awal sesuai dengan jumlah populasi yang telah ditentukan.
- 4. Melakukan proses reproduksi, yaitu yang pertama dengan cara *crossover* dengan memilih dua *parent* secara acak lalu menentukan titik potongnya dan menghasilkan *child*. Metode *crossover* yang digunakan adalah *one cut-point crossover*.
- 5. Melakukan proses reproduksi, yaitu yang kedua dengan cara mutasi dengan cara memilih dua gen pada kromosom yang telah terpilih secara acak untuk dilakukan proses mutasi, kemudian ditukarkan nilai gennya tersebut dan menghasilkan child. Metode mutasi yang digunakan adalah reciprocal exchange mutation.
- 6. Melakukan perhitungan nilai *fitness* pada masing-masing individu.
- 7. Melakukan evaluasi dengan cara seleksi dengan memilih suatu individu sebanyak jumlah populasi gabungan dari individu dan *child* hasil proses *crossover* dan mutasi untuk dapat berlanjut pada iterasi selanjutnya berdasarkan nilai *fitness* tertinggi sampai terendah sesuai dengan jumlah populasi.

8. Apabila telah memenuhi kriteria kondisi berhenti, maka proses iterasi akan berhenti dan dihasilkan solusi terbaik yaitu suatu individu yang memiliki nilai *fitness* tertinggi. Namun, apabila kriteria kondisi berhenti belum terpenuhi, maka proses iterasi akan terus berlanjut.



Gambar 4.1 Flowchart Proses Algoritma Genetika

4.3 Siklus Penyelesaian Algoritma Genetika

Adapun siklus penyelesaian permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika dapat dijelaskan dalam beberapa sub bab adalah sebagai berikut.

4.3.1 Representasi Kromoson dan Perhitungan Fitness

Dalam permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan representasi kromosom permutasi. Penyeberangan Ketapang – Gilimanuk dalam satu hari dibutuhkan 8 kapal tiap harinya sehingga kebutuhan penjadwalan kapal selama satu minggu yakni 7 hari maka kromosom penyusunan menggunakan algoritma genetika adalah sebanyak 8 x 7 = 56 gen. Gen dalam kromosom direprentasikan dengan angka *integer* yang menunjukkan nomor kode kapal. Terdapat 10 kapal yang ada sehingga susunan angka *integer* gennya yaitu angka 1 sampai dengan 10.

Dalam waktu 1 minggu atau 7 hari, setiap kapal mendapat jadwal operasi sebanyak 56 : 10 = 5,6. Sehingga jadwal operasi kapal dapat dilakukan 5 atau 6 kali dalam seminggu.

Setiap 1 kolom gen dalam kromosom menunjukkan *trip*, sehingga 8 kolom menunjukkan kebutuhan kapal tiap harinya yang dapat beroperasi di Pelabuhan Ketapang – Gilimanuk dalam seminggu atau 7 hari. Berikut representasi kromosom yang digambarkan pada Gambar 4.2.

			Har	i ke 1			
1	2	6	3	4	7	8	1
2	3	7	3	5	5	10	4
3	2	8	7	1	6	9	5
4	3	9	8	5	1	6	10
5	4	10	X19	6	4	7	8
6	5	1	10		8	19	2
2	6	2	1	10	9	35	4
				— M	MU		1

Hari ke 7

Gambar 4.2 Representasi Kromosom

Berikut ini merupakan konversi kromoson menjadi jadwal kapal selama 7 hari yaitu pada Tabel 4.4 adalah sebagai berikut.

Tabel 4.4 Konversi Kromosom Menjadi Jadwal Kapal selama 7 hari

HARI	Representasi Kromosom									
ПАКІ			A9	KA	APAL	A.R.				
1.	1	2	6	3	4	7	8	1		
2.	2	3	7	3	5	5	10	4		
3.	3	2	8	7	1	6	9	5		
4.	4	3	9	8	5	1	6	10		
5.	5	4	10	9	6	4	7	8		
6.	6	5	1	10	7	8	9	2		
7.	2	6	2	1	10	9	3	4		

Nilai *Fitness* adalah ukuran suatu nilai yang dibentuk dari individu. Nilai tersebut dihitung berdasarkan jumlah pelanggaran penyusunan penjadwalan. Berikut merupakan rumus nilai *fitness* yang dibentuk.

$$Fitness = \frac{1}{1 + (\sum P1 * 20 + \sum P2 * 35 + \sum P3 * 45)}$$
(4.1)

Keterangan:

P1 = Pelanggaran 1

P2 = Pelanggaran 2

P3 = Pelanggaran 3

Untuk dapat menghasilkan penjadwalan yang baik, terdapat beberapa constraint atau batasan dalam penyusunan penjadwalan kapal penyeberangan Ketapang — Gilimanuk. Pengertian constraint adalah suatu batasan yang tidak boleh terjadi agar dapat menghasilkan suatu penjadwalan yang baik. Apabila penjadwalan tersebut terjadi suatu pelanggaran dengan apa yang telah ditetapkan maka akan diberikan suatu nilai pinalti untuk setiap pelanggaran. Nilai konstanta pada rumus fitness merupakan nilai yang berdasarkan pada jenis pelanggaran.

Jenis pelanggaran ada 2 jenis, yaitu hard constraint dan soft constraint. Hard constraint adalah suatu batasan atau aturan keras yang harus dapat dipenuhi dalam penyusunan penjadwalan, sedangkan soft constraint adalah batasan atau aturan yang masih bisa "ditawar", akan tetapi sedapat mungkin dipenuhi agar dapat menghasilkan solusi penjadwalan yang baik. Apabila jumlah pelanggaran yang dihasilkan semakin kecil, maka penjadwalan yang dihasilkan akan semakin baik sehingga solusi penjadwalan yang baik dapat terpenuhi. Berikut Tabel 4.5 merupakan batasan atau constraint dalam permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk adalah sebagai berikut.

Tabel 4.5 Jenis Pelanggaran Penjadwalan Kapal

No.	Pelanggaran	Pelanggaran Pelanggaran		Nilai Pelanggaran
1.	Kapal beroperasi tidak sesuai dengan porsi yang telah ditentukan (P1)	soft constraint	20	1
2.	Dalam sehari, kapal tidak boleh muncul lebih dari satu nama kapal yang sama (P2)	hard constraint	35	1
3.	Apabila kapal docking, maka kapal tidak dapat beroperasi (P3)	hard constraint	45	1

4.3.2 Inisialisasi Populasi Awal

Inisialisasi populasi awal merupakan penyelesaian pencarian untuk mendapatkan solusi penjadwalan yang optimal. Algoritma genetika ini dilakukan inisialisasi dengan membuat suatu himpunan solusi yang nantinya akan dievalusi yang apabila solusi tersebut bukan merupakan solusi yang optimal maka dilakukan proses reproduksi (Berlianty, Arifin, 2010).

Dalam permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk, peneliti menggunakan representasi *integer* yang dibangkitkan secara acak sesuai dengan porsi yang telah ditentukan. Berikut merupakan contoh individu awal secara acak yang ditunjukkan pada Gambar 4.3 adalah sebagai berikut.

1	2	6	3	4	37	8	1
2	3	7	3	5	5	10	4
3	2	8	7	1	6	9	5
4	3	9	8	5	1	6	10
5	4	10	9	6	4	7	8
6	5	1	10	7	8	9	2
2	6	2)	1	10/	9	3	4

Gambar 4.3 Contoh Individu Awal

4.3.3 Reproduksi

Terdapat proses *crossover* dan mutasi pada tahap reproduksi ini. *Crossover* adalah proses penggabungan dua kromosom induk yang sebelumnya telah ditentuan titik potongnya menjadi kromosom baru (*offspring*). Metode *crossover* yang digunakan dalam permasalahan penjadwalan kapal ini adalah *one cut-point crossover*.

Pada contoh perhitungan manual ini, peneliti menggunakan *crossover* dengan inisialisasi *crossover rate* (*cr*) 0,2. Langkah pertama yaitu dengan memilih 2 induk secara acak untuk melalui proses *crossover* tersebut. Individu yang terpilih untuk melakukan proses *crossover* adalah individu 1 dan 2. Titik *crossover* dipilih secara acak dan menghasilkan *one cut-point crossover* pada hari ke 4. Berikut Gambar 4.4 yang merupakan hasil proses *crossover*.

Parent 1

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

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

Parent 2

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

Child 1

1	2	6	3	4 /	575	85	1
2	3	7	3	5	5 (10	4
3	2	8	7	/14	6	9 7	5
4	3	9	8	5	1	6	10
1	10	9	8	6	4 (5	7
3	4	6	9	2	13	77	1
10	9	10	5	4	3	7	6

Child 2

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

Gambar 4.4 Proses Crossover

Setelah melakukan proses *crossover*, langkah selanjutnya yaitu dengan melakukan proses mutasi. Mutasi adalah proses perubahan nilai gen yang dilakukan secara acak. Dalam penelitian ini, proses mutasi yang dilakukan yaitu dengan menggunakan metode *reciprocal exchange mutation* yaitu dengan memilih dua posisi (*exchange point/XP*) kromosom pada hari tertentu secara acak kemudian menukar nilai pada posisi tersebut. Individu yang terpilih untuk melakukan proses *crossover* adalah individu 3. Berikut Gambar 4.5 yang merupakan contoh perhitungan manual hasil proses mutasi yang dilakukan dengan inisialisai *mutation rate* (*mr*) 0,3.

Parent 3 sebelum proses mutasi

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

Parent 3 setelah proses mutasi

7	1	4	8 (9	10	3	2
1	5	6	8 -	3	5	9	1
1	4	6	9	2	8	2	10
6	5	4	3	2	J12	7	7
3	5	7	8	2	4	5	10
3	6	6	9	10/	4	1	7
9	10	8	2	3	4	5	6

Gambar 4.5 Proses Mutasi

4.3.4 Evaluasi dan Seleksi

Pada tahap proses evaluasi ini dilakukan perhitungan *fitness* pada tiap individu dalam suatu populasi. Metode seleksi yang peneliti gunakan adalah dengan menggunakan metode *elitism* yaitu dengan memilih individu yang memiliki nilai *fitness* tertinggi. Perhitungan nilai *fitness* dilakukan pada tiap individu dalam suatu populasi. Didapatkan nilai *fitness* sebanyak 15 yaitu 10 dari individu awal, 2 dari hasil proses *crossover*, dan 3 dari hasil proses mutasi. Adapun hasil perhitungan nilai *fitness* yang didapat, ditunjukkan pada Tabel 4.9 yaitu sebagai berikut.

Tabel 4.6 Hasil Perhitungan Nilai Fitness

Individu	Nilai <i>Fitness</i>
P1	0,005682
P2	0,007092
P3	0,004065
P4	0,009434
P5	0,004739
P6	0,007092
P7	0,003559
P8	0,004739
P9	0,005682
P10	0,003165
C1	0,004237
C2,	0,004975
- M1	0,004739
M2	0,005682
M3	0,003559

Proses seleksi *elitism* dilakukan melalui proses *sorting* dengan nilai *fitness* individu tertinggi sebanyak jumlah populasi. Individu baru yang terpilih ditunjukkan pada Tabel 4.10 yaitu sebagai berikut.

Tabel 4.7 Hasil Proses Seleksi Elitism Pertama

Individu	Nilai Fitness	Parent Asal
P1	0,009434	P4
P2	0,007092	P2
Р3	0,007092	P6
P4	0,005682	P1
P5	0,005682	P9
P6	0,005682	M2
P7	0,004975	C2
P8	0,004739	P5

P9	0,004739	P8
P10	0,004739	M1
C1	0,004237	C1
C2	0,004065	P3
M1	0,003559	P7
M2	0,003559	M3
M3	0,003165	P10

Iterasi tersebut dilakukan sampai iterasi kedua yang akan menghasilkan solusi penjadwalan kapal dengan nilai *fitness* yang ditunjukkan pada Tabel 4.11 yaitu sebagai berikut.

Tabel 4.8 Hasil Proses Seleksi Elitism Kedua

Individu	Nilai Fitness	Parent Asal
P1		M1
P2	0,02778	P1
P3 (11)	0,01408	P2
P4 (8)	0,01408	M2
P5	0,01408	M3
P6 (0,00943	20 P3
P7	0,00709	P4
P8	0,00709	P5
P9	0,00662	C1
P10	0,00585	// P6

Kemudian akan didapat solusi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika yaitu pada Gambar 4.6 sebagai berikut.

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

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

Gambar 4.6 Jadwal Penyeberangan Kapal Penyeberangan Ketapang – Gilimanuk

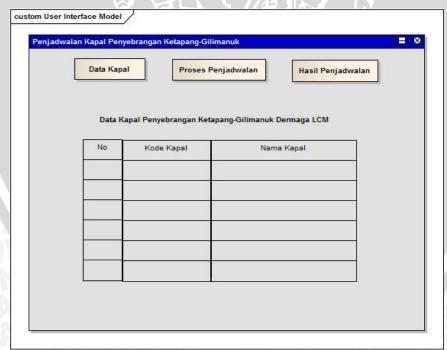
Solusi jadwal pada Gambar 4.6 tersebut merupakan jadwal kapal pada hari pertama sampai hari ke tujuh dengan tidak adanya pelanggaran 1, 2, atau 3, hard constraint maupun soft constraint. Jadwal kapal yang beroperasi antara lain kapal yang sesuai dengan kode kapal tersebut.

4.4 Perancangan User Interface

Perancangan *user interface* untuk sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk Menggunakan Algoritma Genetika ini terdiri dari tiga halaman yaitu perancangan halaman data kapal, perancangan halaman proses penjadwalan kapal, dan perancangan halaman hasil penjadwalan kapal menggunakan algoritma genetika.

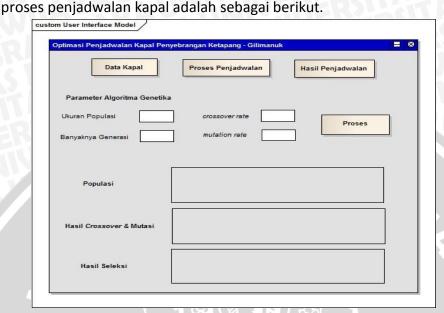
1. Perancangan User Interface Halaman Data Kapal

Halaman data kapal tersebut menampilkan data kapal yang penyeberangan Pelabuhan Ketapang – Gilimanuk di dermaga LCM. Pada Gambar 4.8 ditunjukkan gambar tampilan rancangan *user interface* halaman data kapal adalah sebagai berikut.



Gambar 4.6 Halaman Data Kapal Penyeberangan Ketapang – Gilimanuk

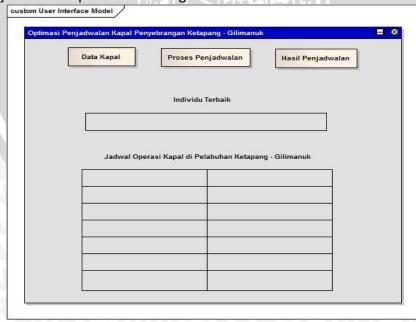
2. Perancangan *User Interface* Halaman Proses Penjadwalan Kapal Halaman proses penjadwalan kapal tersebut menampilkan proses algoritma genetika dalam permasalahan penyusunan jadwal kapal. Pada Gambar 4.9 ditunjukkan gambar tampilan rancangan *user interface* halaman proses penjadwalan kapal adalah sebagai harilwat



Gambar 4.7 Halaman Proses Penjadwalan Kapal Penyeberangan

3. Halaman Perancangan User Interface Halaman Hasil Penjadwalan Kapal

Halaman hasil penjadwalan kapal tersebut menampilkan hasil penyusunan penjadwalan kapal menggunakan algoritma genetika. Pada Gambar 4.10 ditunjukkan gambar tampilan rancangan *user interface* halaman hasil penjadwalan kapal adalah sebagai berikut.



Gambar 4.8 Halaman Hasil Penjadwalan Kapal Penyeberangan

BAB 5 IMPLEMENTASI

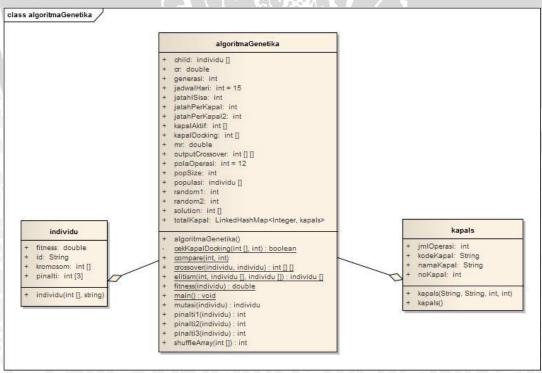
Bab ini membahas tentang tahap implementasi sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika yang berdasarkan pada perancangan yang telah dibahas pada bab sebelumnya. Untuk menjelaskan tahap ini, akan dibagi menjadi dua sub-bab yaitu struktur class dan potongan source code utama.

5.1 Struktur Class

Struktur *class* ini terdiri dari beberapa method dengan menggunakan bahasa pemrograman Java. Tahap ini merupakan tahap perancangan dalam proses pembuatan suatu sistem yang digunakan untuk mengimplementasi perancangan tersebut menjadi sebuah program penjadwalan kapal penyeberangan menggunakan algoritma genetika.

5.1.1 Class Algoritma Genetika

Perancangan *class* algoritma genetika mengenai permasalahan optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk ini terdiri dari pembangkitan populasi awal, proses *crossover* menggunakan *one-cut point crossover*, proses mutasi menggunakan *reciprocal exchange mutation*, perhitungan nilai *fitness*, dan proses seleksi menggunakan *elitism selection*. Gambar 5.1 merupakan *class* diagram algoritma genetika pada implementasi mengenai permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika adalah sebagai berikut.



Gambar 5.1 Class Diagram Algoritma Genetika

5.2 Potongan Source Code Utama

5.2.1 Pembangkitan Populasi Awal

Pembangkitan populasi awal yaitu dengan pembentukan individu awal dengan menggunakan representasi permutasi berbasis *integer* yang dibangkitkan secara acak. Adapun potongan *source code* utama dalam pembangkitan populasi awal terdapat pada *Source Code* 5.1 adalah sebagai berikut.

```
public algoritmaGenetika()
2
3
                totalKapal = new LinkedHashMap<>(29);
                totalKapal.put(1, new kapals (1, 0, "DRC", "DARMA
4
5
    RUCITRA"));
6
                totalKapal.put(2, new kapals (2, 0, "PST", "PUTRI
    SRITANJUNG"));
8
                totalKapal.put(3, new kapals (3, 0, "PIN", "PANCAR
    INDAH"));
9
10
                totalKapal.put(4, new kapals (4, 0, "REG", "RAJA
11
    ENGGANO"));
12
                totalKapal.put(5, new kapals (5, 0, "LAD", "LABITRA
    ADINDA"));
13
14
                totalKapal.put(6, new kapals (6, 0, "ARJ", "ARJUNA"));
                totalKapal.put(7, new kapals (7, 0, "GIL2", "GILIMANUK
15
    2"));
16
17
                totalKapal.put(8, new kapals (8, 0, "GB2", "GERBANG
18
    SAMUDRA"));
19
                totalKapal.put(9, new kapals (9, 0, "GIL1", "GILIMANUK
20
    1"));
21
                totalKapal.put(10, new kapals (10, 0, "TNP", "TUNU
22
    PRATAMA"));
                totalKapal.put(11, new kapals (11, 0, "TJ9", "TRIMA
23
24
    JAYA 9"));
25
                totalKapal.put(12, new kapals (12, 0, "PRT",
26
    "PRATITA"));
27
                totalKapal.put(13, new kapals (13, 0, "TB2", "TRISILA
28
    BAKTI 2"));
29
                totalKapal.put(14, new kapals (14, 0, "LBT",
     "LIBERTY"));
30
31
                totalKapal.put(15, new kapals (15, 0, "NGF", "NIAGA
32
    FERRY"));
33
                totalKapal.put(16, new kapals (16, 0, "MRP", "MARINA
34
    PRATAMA"));
35
                totalKapal.put(17, new kapals (17, 0, "JMB", "JAMBO"));
36
                totalKapal.put(18, new kapals (18, 0, "RJW",
37
    "RAJAWALI"));
38
                totalKapal.put(19, new kapals (19, 0, "SMS", "SMS
    SWAKARYA"));
39
40
                totalKapal.put(20, new kapals (20, 0, "TEL", "TRISAKTI
    ELFINA"));
41
                totalKapal.put(21, new kapals (21, 0, "RAF",
42
43
    "RAFELIA"));
44
                totalKapal.put(22, new kapals (22, 0, "DMF", "DHARMA
45
    FERRY"));
46
                totalKapal.put(23, new kapals (23, 0, "HER",
47
    "HERLIN"));
48
                totalKapal.put(24, new kapals (24, 0, "SBK", "SUMBER
49
    BERKAT"));
                totalKapal.put(25, new kapals (25, 0, "TAD", "TRISAKTI
50
    ADINDA"));
51
                totalKapal.put(26, new kapals (26, 0, "TB1", "TRISILA
52
53
    BAKTI 1"));
```

```
totalKapal.put(27, new kapals (27, 0, "NSD", "NUSA
55
     DUA"));
56
                 totalKapal.put(28, new kapals (28, 0, "SDM", "SERIAL
57
     DOMAR"));
58
                 totalKapal.put(29, new kapals (29, 0, "CLG",
     "CEMERLANG"));
59
60
                 //untuk mengecek kapal yang aktif
61
                 kapalAktif = new int[totalKapal.size() -
62
     kapalDocking.length];
63
                 int p = 0;
64
                 for (Map.Entry<Integer, kapals> i :
65
     totalKapal.entrySet())
66
67
     if(!algoritmaGenetika.cekKapalDocking(i.getValue().noKapal,
     kapalDocking))
68
69
70
                          kapalAktif[p++] = i.getValue().noKapal;
                          for (int j = 0; j < kapalAktif.length; j++) {</pre>
71
72
                              if (!(kapalAktif[j]==0)) {
73
74
75
76
                 jatahPerKapal = (int) Math.floor(jadwalHari *
77
78
     polaOperasi * 1f / kapalAktif.length);
79
                 jatahPerKapal2 = (int) Math.ceil(jadwalHari *
80
     polaOperasi * 1f / kapalAktif.length);
     jatahSisa = jadwalHari * polaOperasi -
(kapalAktif.length * jatahPerKapal);
81
82
83
                 //perhitungan jatah porsi per kapal yang aktif
84
                 for (int i = 0; i < kapalAktif.length; i++) {</pre>
85
                 totalKapal.get(kapalAktif[i]).jmlOperasi=jatahPerKapal;
86
87
                 for(int i = 0; i < jatahSisa; i++)</pre>
88
89
                     random1 = (int) (Math.random()*kapalAktif.length);
90
91
     (totalKapal.get(kapalAktif[random1]).jmlOperasi==jatahPerKapal2) {
91
                     } else {
92
                       totalKapal.get(kapalAktif[random1]).jmlOperasi++;
93
94
95
96
                 for(int i = 0; i < kapalAktif.length; ++i)</pre>
97
98
99
                 //individu awal sebelum di shuffle yg telah sesuai dgn
100
     jatah kapal
101
                 solution = new int[jadwalHari * polaOperasi];
102
                 p = 0;
103
                 for(int i = 0 ; i < kapalAktif.length; ++i)</pre>
104
105
                     for (int j = 0; j <
106
     totalKapal.get(kapalAktif[i]).jmlOperasi; ++j)
107
108
                          solution[p++] =
109
     totalKapal.get(kapalAktif[i]).noKapal;
200
201
202
             shuffleArray(solution);
```

Source Code 5.1 Pembangkitan Populasi Awal

Kemudian untuk proses pembentukan populasi awal yang dibangkitkan sebesar ukuran *popSize* terdapat pada *Source Code* 5.2 adalah sebagai berikut.

```
int[] solusi = Arrays.copyOf(solution,

int[] temp;

int[] temp;

for (int i = 0; i < popSize; i++) {
    temp=shuffleArray(solusi);
    populasi[i]= new individu ("Individu "+(i+1),

Arrays.copyOf(temp, temp.length));

</pre>
```

Source Code 5.2 Melakukan Inisialisasi Populasi Awal Sebesar popSize

5.2.2 Proses Crossover

Pada proses *crossover* ini menggunakan metode *one-cut point* yaitu dengan membangkitkan titik potong secara acak dengan memilih dua *parent* secara acak kemudian dilakukan proses persilangan. Adapun potongan *source code* utama pada proses *crossover* terdapat pada *Source Code* 5.3 dan proses pertukaran gen pada antar individu terdapat pada *Source Code* 5.4 adalah sebagai berikut.

```
public static int [][] crossover (individu individu1, individu
     individu2) {
3
              Random random = ThreadLocalRandom.current();
4
              int [][]hasil=new int [2][];
5
              //one cut-point crossover secara random pada parent 1
6
              int [] [] parent1 = new int[2][];
              int oneCut = 180;
8
              int parent1Cut1;
9
              parent1Cut1 = 1 + (random.nextInt(178)); // titik potong
10
     dari gen 1 sampai 179
11
              int parent1IndexCut1 = parent1Cut1;
              parent1 [0] = new int [parent1IndexCut1];
for (int i=0; i < parent1IndexCut1 ; i++) {</pre>
12
13
14
                   parent1 [0][i] = individu1.kromosom [i];
15
16
              int parent1Cut2= oneCut - parent1Cut1;
17
              int parent1IndexCut2 = parent1Cut2;
              parent1 [1] = new int [parent1IndexCut2];
for (int i=0; i< parent1IndexCut2; i++) {</pre>
18
19
20
                   parent1 [1][i] = individu1.kromosom
21
     [i+parent1IndexCut1];
22
              //one cut-point crossover pada parent 2
23
24
              int [] [] parent2 = new int[2][];
25
              int oneCut2 = 180;
26
              int parent2Cut1 = parent1Cut1;
              int parent2IndexCut1 = parent2Cut1;
27
28
              parent2 [0] = new int [parent2IndexCut1];
29
              for (int i=0; i < parent2IndexCut1 ; i++) {</pre>
30
                   parent2 [0][i] = individu2.kromosom [i];
31
32
              int parent2Cut2 = oneCut2 - parent2Cut1;
33
              int parent2IndexCut2 = parent2Cut2;
              parent2 [1] = new int [parent2IndexCut2];
for (int i=0; i<parent2IndexCut2; i++) {</pre>
34
35
                   parent2 [1][i] = individu2.kromosom
36
37
     [i+parent2IndexCut1];
38
```

Source Code 5.3 Proses Crossover dengan Metode One-Cut Point Crossover

```
//menggabungkan parent \overline{1} sebelah kiri dengan parent 2 sebelah
2
3
      int crossover [] = new int [parent1[0].length+parent2[1].length];
4
             for (int i = 0; i < parent1[0].length+parent2[1].length;</pre>
5
    i++) {
6
                 if (i<parent1[0].length) {</pre>
                      crossover[i]=parent1[0][i];
8
                 } else {
9
                      crossover[i]=parent2[1][i-parent1[0].length];
10
11
12
     //menggabungkan parent 2 sebelah kiri dengan parent 1 sebelah
13
             int crossover2 []= new int
14
15
     [parent2[0].length+parent1[1].length];
16
             for (int i = 0; i < parent2[0].length+parent1[1].length;</pre>
17
    i++) {
18
                 if (i<parent2[0].length) {
                      crossover2[i]=parent2[0][i];
19
20
                 } else {
21
                      crossover2[i]=parent1[1][i-parent2[0].length];
22
23
24
             hasil[0]=crossover;
25
             hasil[1]=crossover2;
26
             return hasil;
27
```

Source Code 5.4 Proses Pertukaran Gen Antar Individu

5.2.3 Proses Mutasi

Proses mutasi menggunakan metode reciprocal exchange mutation yaitu dengan memilih dua gen pada suatu kromosom yang telah terpilih, lalu ditukarkan nilai gennya. Adapun potongan source code utama pada proses mutasi menggunakan metode reciprocal exchange mutation terdapat pada Source Code 5.5 adalah sebagai berikut.

```
public individu mutasi(individu parent) {
2
            //memilih dua nilai gen secara acak
3
            random1 = (int) (Math.random()*parent.kromosom.length);
4
            random2 = (int) (Math.random()*parent.kromosom.length);
5
            int [] array = Arrays.copyOf(parent.kromosom,
6
    parent.kromosom.length);
            individu mutasi = new individu ("mutasi", array);
            int a = parent.kromosom[random2];
8
9
            int b = parent.kromosom[random1];
10
            //menukar nilai gen
            mutasi.kromosom[random1] = a;
11
12
            mutasi.kromosom[random2] = b;
13
            return mutasi;
```

Source Code 5.5 Mutasi dengan Metode Reciprocal Exchange Mutation

5.2.4 Perhitungan Nilai Fitness

Pada perhitungan nilai *fitness* ini menggunakan rumus yang telah dibahas pada Bab 4 mengenai jenis pelanggaran dan nilainya. Adapun potongan *source code* utama pada proses perhitungan nilai *fitness* terdapat pada *Source Code* 5.6 adalah sebagai berikut.

```
public static double fitness (individu individu1) {
2
            double nilaiFitness;
3
            //perhitungan rumus fitness yang digunakan
4
            nilaiFitness = 1.0/(1.0+((individu1.pinalti[0]*20.0) +
5
    (individu1.pinalti[1]*35.0) + (individu1.pinalti[2]*45.0)));
6
            return nilaiFitness;
```

Source Code 5.6 Proses Perhitungan Nilai Fitness

5.2.5 Proses Seleksi

Proses seleksi menggunakan metode elitism yaitu dengan mengurutkan individu yang memiliki nilai fitness tertinggi. Adapun potongan source code utama pada proses seleksi terdapat pada Source Code 5.7 adalah sebagai berikut.

```
public static individu[] elitism(individu[] parent, individu[]
    child, int popSize) {
             individu max;
3
             int pointer = 0;
             //membuat array baru hasil gabungan individu dan child
5
    hasil proses crossover & mutasi
6
            individu[] gabunganIndividuChild =
8
    Stream.concat(Arrays.stream(parent),
    Arrays.stream(child)).toArray(individu[]::new);
9
10
             //gabungan individu dan child sebelum diurutkan nilai
11
    fitness-nya dari yg tertinggi sampai terendah
12
             for (int i = 0; i < gabunganIndividuChild.length; i++) {</pre>
13
             //gabungan individu dan child setelah diurutkan nilai
14
15
    fitness-nya dari yg tertinggi sampai terendah
            Arrays.sort(gabunganIndividuChild, individuFitnessDesc);
16
17
             for (int i = 0; i < gabunganIndividuChild.length; i++) {</pre>
18
19
             //mengembalikan hasil sorting sejumlah popSize
20
             return Arrays.copyOfRange(gabunganIndividuChild, 0,
21
    popSize);
22
```

Source Code 5.7 Proses Seleksi dengan Metode Elitism

BAB 6 PENGUJIAN DAN PEMBAHASAN

Bab ini membahas tentang pengujian pada sistem optimasi penjadwalan kapal penyeberangan Ketapang — Gilimanuk dan pembahasan mengenai hasil dari pengujian parameter yang digunakan dalam algoritma genetika yaitu terdiri dari pengujian ukuran populasi, pengujian banyaknya generasi, dan pengujian kombinasi crossover rate (cr) dan mutation rate (mr).

6.1 Pengujian

Pengujian ini dilakukan untuk mendapatkan solusi yang optimal dalam permasalahan penjadwalan penyeberangan Ketapang – Gilimanuk menggunakan Algoritma Genetika.

6.1.1 Hasil Pengujian Ukuran Populasi

Dalam pengukuran pengujian ukuran populasi yang tepat bertujuan untuk menghasilkan solusi yang optimal pada sistem penjadwalan kapal penyeberangan Ketapang – Gilimanuk ditinjau dari rata-rata nilai *fitness* terbaik yang dihasilkan. Ukuran populasi yang akan diujikan yaitu 20, 40, 60, 80, 100, 120, 140, 160, 180 dan 200 sebanyak 10 kali percobaan dengan menggunakan banyaknya generasi 50, serta kombinasi *crossover rate* (*cr*)= 0,6 dan *mutation rate* (*mr*)= 0,4. Tabel 6.1 merupakan hasil pengujian ukuran populasi yaitu sebagai berikut.

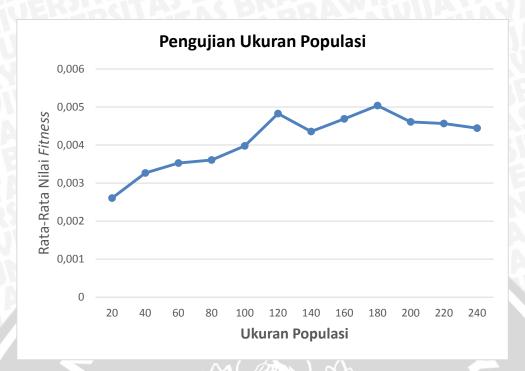
Tabel 6.1 Hasil Pengujian Ukuran Populasi

Ukuran					Nilai <i>I</i>	itness			6		Rata-rata	
			Perc	obaaı	า Uku	ran Po	pulas	i ke-	9		Nilai	
Populasi	1	2	3	4	5	6	7	8	9	10	Fitness	
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0		
	02	02	02	02	03	02	02	02	03	02		
24 1	34	62	43	05	16	10	52	84	26	69		
	74	46	30	76	45	08	52	90	79	54		
20	17	71	90	13	56	40	52	02	73	17	0,00260658	
451	84	91	02	16	96	33	52	84	85	78	4	
	03	60	43	87	20	61	52	90	62	97		
MINA	75	10	30	24	25	34	52	02	09	57		
	58	49	90	28	31	45	52	84	15	41		
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	MORA	
411	04	03	02	02	02	03	03	02	04	02		
MALAT	42	26	97	62	65	32	26	46	97	65	0,00326411	
40	47	79	61	46	95	22	79	30	51	95	8	
	78	73	90	71	74	59	73	54	24	74	3	
	76	85	47	91	46	13	85	18	37	46		
	10	62	61	60	80	62	62	71	81	80		

							A Z G				
	61 94	09 15	90 47	10 49	85 10	12 62	09 15	92 11	09 45	85 10	Attil
			VAS		A P A		THE			1 - 4 -	
	0,0	0,0	0,0	0,0	0,0 04	0,0	0,0	0,0	0,0	0,0 04	PLOB
MALAN	55	03 98	19	21	52	97	52	49	62	14	(A) Pa
	87	40	29	54	48	61	48	65	46	93	
60	18	63	82	34	86	90	86	03	71	77	0,00352477
00	86	74	45	08	87	47	87	49	91	59	2
TARRE	12	50	61	36	78	61	78	65	60	33	
SCHIL	09	19	40	01	28	90	28	03	10	60	
TIE.	96	92	35	28	05	47	05	49	49	99	
HTIV	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	02	03	03	05	02	03	03	03	03	03	
	80	49	75	52	93	83	55	69	06	37	
	89	65	93	48	25	14	87	00	74	83	0.00000400
80	88	03	98	61	51	17	18	36	84	78	0,00360483
	76	49	49	87	31	62	86	90	66	37	4
	40	65	62	84	96	45	12	03	25	83	
	44	03	40	53	48	21	09	69	76	78	
	94	49	60	03	09	07	96	00	68	37	
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	05	04	03	03	04	03	04	03	04	04	
	10	06	49	21	23	26	23	06	32	73	
	20	50	65	54	72	79	72	74	90	93	0,00397573
100	40	40	03	34	88	73	88	84	04	36	9
	81	65	49	08	13	85	13	66	32	49	3
	63	04	65	36	55	62	55	25	90	28	
	26	06	03	01	93	09	93	76	04	90	
31.1	53	5	49	28	22	15	22	68	32	99	
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	10	04	03	04	03	05	03	03	04	04	
45)	98	52	26	32	69	52	21	32	42	97	
120	90	48 86	79 73	90 04	00 36	48 61	54 34	22 59	47 78	51 24	0,00482633
120	98	87	85	32	90	87	08	13	76	37	7
	90	78	62	90	03	84	36	62	10	81	
TUA	10	28	09	04	69	53	01	12	61	09	SOA
1	9	05	15	32	00	03	28	62	94	45	SPER
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	03	02	04	06	05	03	06	02	03	05	
140	69	69	23	84	10	06	02	97	37	52	0,00435451
TAS	00	54	72	93	20	74	40	61	83	48	1
LATE	36	17	88	15	40	84	96	90	78	61	

	90	78	13	06	81	66	38	47	37	87	
	03	97	55	84	63	25	55	61	83	84	
UAU	69	57	93	93	26	76	42	90	78	53	SOAM
	00	41	22	15	53	68	16	47	37	03	REBR
ANTIV	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	La AS
1244	06	02	04	05	04	04	04	06	03	03	
	84	84	73	37	97	52	32	41	37	43	4131.677
	93	90	93	63	51	48	90	02	83	64	MATTI
160	15	02	36	44	24	86	04	56	78	26	0,00468680
25114	06	84	49	08	37	87	32	41	37	11	7
413.6	84	90	28	60	81	78	90	02	83	68	AUD A
MATT	93	02	90	21	09	28	04	56	78	38	
	15	84	99	50	45	05	32	41	37	48	1
12/	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	05	04	03	04	06	05	07	02	04	05	
	37	97	83	85	21	10	09	89	32	68	
	63	51	14	43	11	20	21	01	90	18	
180	44	24	17	68	80	40	98	73	04	18	0,00503436
	08	37	62	93	12	81	58	41	32	18	7
	60	81	45	20	42	63	15	04	90	18	
	21	09	21	38	23	26	60	04	04	18	
	50	45	07	83	60	53	28	62	32	18	
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	04	04	05	03	05	04	06	04	03	04	
	42	14	84	55	23	06	02	73	69	32	
	47	93	79	87	56	50	40	93	00	90	0.00460630
200	78	77	53	18	02	40	96	36	36	04	0,00460639
	76	59	21	86	09	65	38	49	90	32	5
	10	33	63	12	42	04	55	28	03	90	
4	61	60	74	09	40	06	42	90	69	04	
24	94	99	26	96	83	50	16	99	00	32	

Berdasarkan hasil pengujian ukuran populasi mengenai sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika pada Tabel 6.1, didapatkan bahwa semakin besar ukuran populasi, maka nilai *fitness* yang dihasilkan juga cenderung mengalami peningkatan.



Gambar 6.1 Grafik Hasil Pengujian Ukuran Populasi

Terlihat bahwa hasil pengujian pada Gambar 6.1, ukuran populasi 20 sampai dengan 120 mengalami kenaikan rata-rata nilai *fitness*. Akan tetapi, pada ukuran populasi 140 mengalami penurunan dan mengalami kenaikan kembali pada ukuran populasi 160 sampai dengan 180, namun mengalami penurunan kembali pada ukuran populasi 200 sampai dengan 240. Pengujian dengan ukuran populasi yang terlalu besar, membutuhkan waktu komputasi yang semakin lama dan belum tentu mendapatkan nilai *fitness* yang lebih tinggi. Peningkatan ukuran populasi juga tidak menjamin adanya kenaikan nilai *fitness* yang diperoleh disebabkan area pencarian semakin besar. Semakin kecil ukuran populasi juga tidak menjamin adanya penurunan nilai *fitness* yang diperoleh karena konsep acak pada algoritma genetika. Pada pengujian ukuran populasi dengan nilai *fitness* yang paling optimal adalah pada ukuran 180 populasi.

6.1.2 Hasil Pengujian Banyaknya Generasi

Dalam pengukuran pengujian banyaknya generasi yang tepat bertujuan untuk menghasilkan solusi yang optimal pada penjadwalan kapal penyeberangan Ketapang – Gilimanuk ditinjau dari rata-rata nilai *fitness* terbaik yang dihasilkan. Banyaknya generasi yang akan diujikan yaitu 20, 40, 60, 80, 100, 120, 140, 160, 180, dan 200 sebanyak 10 kali percobaan dengan menggunakan hasil ukuran populasi terbaik pada hasil pengukuran pengujian sebelumnya yaitu populasi, serta kombinasi *crossover rate* (*cr*)=0,6 dan *mutation rate* (*mr*)= 0,4. Tabel 6.2 merupakan hasil pengujian banyaknya generasi yaitu sebagai berikut.

Tabel 6.2 Hasil Pengujian Banyaknya generasi

Ukuran				4	Nilai <i>F</i>	itness	26				Rata-rata	
Generasi			Percol	baan I	Banya	knya į	gener	asi ke		19	Nilai	
Generasi	1	2	3	4	5	6	7	8	9	10	Fitness	
PER	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	MATE	
	01	01	01	02	01	01	01	01	01	01		
	90	62	58	59	73	78	83	83	67	86	SAUR	
HEDS	11	33	47	06	61	25	15	15	78	56	0,0018425	
20	40	76	86	73	11	31	01	01	52	71	5	
	68	62	05	57	11	19	83	83	34	64		
	44	33	38	51	11	42	15	15	89	17		
	10	76	82	29	11	95	01	01	93	91		
	64	62	72	53	11	90	83	83	28	04		
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	V	
	03	03	02	02	04	03	02	04	03	03		
	83	43	59	14	14	32	12	42	26	55		
40	14	64	06	59	93	22	31	47	79	87		
40	17	26	73	22	77	59	42	78	73	18	9	
	62	11	57	74	59	13	25	76	85	86		
	45	68	51	67	33	62	05	10	62	12		
	21	38	29	81	60	12	30	61	09	09		
	07	48	53	11	99	62	78	94	15	96		
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0		
	04	06	03	05	05	10	07	10	06	08		
	14	21	43	37	84	41	09	98	84	62		
1	93	11	64	63	79	66	21	90	93	06		
60	77	80	26	44	53	66	98	10	15	89	0,0068989	
S	59	12	11	08	21	66	58	98	06	65	6	
	33	42	68	60	63	66	15	90	84	51		
	60	23	38	21	74	66	60	10	93	72		
	99	60	48	50	26	60	28	90	15	41		
	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	/A	
	27	17	10	13	17	12	47	13	06	08		
JAV	77	85	41	15	85	34	61	15	84	62	of Sign	
A	77	71	66	78	71	56	90	78	93	06	0.0175650	
80	77	42	66	94	42	79	47	94	15	89	0,0175659	
	77	85	66	73	85	01	61	73	06	65	5	
50	77	71	66	68	71	23	90	68	84	51	31246	
5 PIT	77	42	66	42	42	45	47	42	93	72	TIVE	
TAD	70	80	60	10	80	60	60	10	15	41		

	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	47	27	16	17	14		17	47	16	12	
WAU	61	77	39	85	08	24	85	61	39	34	SOAM
	90	77	34	71	45	39	71	90	34	56	Drag
100	47	77	42	42	07	02	42	47	42	79	0,02423374
	61	77	62	85	04	43	85	61	62	01	7
	90	77	29	71	22	90	71	90	29	23	
BRS	47	77	50	42	53	24	42	47	50	45	111124
LASP	60	70	80	80	50	39	80	60	80	60	EUTHA
1411						0.0					
	0,0 24	0,0 47	1,0 00	1,0 00	0,0 24	0,0 27	0,0 47	0,0	0,0 17	0,0 47	
TV EAR											
	39	61	00	00	39	77	61	15	85	61	
120	02	90	00	00	02	77	90	78	71	90	0,22504304
120	43	47	00	00	43	77	47	94	42	47	5
	90	61	00	00	90	77	61	73	85	61	, ,
	24	90	00	00	24	77	90	68	71	90	
	39	47	00	00	39	77	47	42	42	47	
	00	60	00	00	00	70	60	10	80	60	
	1,0	1,0	0,0	0,0	1,0	0,0	0,0	1,0	0,0	0,0	
	00	00	14	24	00	47	47	00	47	12	
	00	00	08	39	00	61	61	00	61	34	
	00	00	45	02	00	90	90	00	90	56	
140	00	00	07	43	00	47	47	00	47	79	0,41936775
	00	00	04	90	00	61	61	00	61	01	7
	00	00	22	24	00	90	90	00	90	23	
	00	00	53	39	00	47	47	00	47	45	
	00	00	50	00	00	60	60	00	60	60	
	1,0	0,0	1,0	1,0	0,0	1,0	1,0	1,0	0,0	1,0	
	00	27	00	00	47	00	00	00	24	00	
	00	77	00	00	61	00	00	00	39	00	
AS N	00	77	00	00	90	00	00	00	02	00	
160	00	77	00	00	47	00	00	00	43	00	0,70997870
100	00	77	00	00	61	00	00	00	90	00	7
	00	77	00	00	90	00	00	00	24	00	
ALTO F											
	00	77	00	00	47	00	00	00	39	00	
	00	70	00	00	60	00	00	00	00	00	
	1,0	0,0	0,0	1,0	1,0	0,0	1,0	0,0	1,0	1,0	RBRA
4111	00	24	27	00	00	24	00	47	00	00	12 KC B
100	00	39	77	00	00	39	00	61	00	00	0,61241773
180	00	02	77	00	00	02	00	90	00	00	1
AR	00	43	77	00	00	43	00	47	00	00	EHERS
	00	90	77	00	00	90	00	61	00	00	
HT BY	00	24	77	00	00	24	00	90	00	00	TINIT

	00	39	77	00	00	39	00	47	00	00	VILLETTA
Little	00	00	70	00	00	00	00	60	00	00	
MAT	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	BRES
	00	00	00	00	00	00	00	00	00	00	AS BR
Attil	00	00	00	00	00	00	00	00	00	00	A AS
	00	00	00	00	00	00	00	00	00	00	
200	00	00	00	00	00	00	00	00	00	00	1,00000000
LKGB	00	00	00	00	00	00	00	00	00	00	000000000
	00	00	00	00	00	00	00	00	00	00	
	00	00	00	00	00	00	00	00	00	00	JAUL
HT ELL	00	00	00	00	00	00	00	00	00	00	

Berdasarkan hasil pengujian banyaknya generasi mengenai sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika pada Tabel 6.2, didapatkan bahwa semakin besar banyaknya generasi, maka nilai fitness yang dihasilkan juga cenderung mengalami peningkatan.



Gambar 6.2 Grafik Hasil Pengujian Banyaknya Generasi

Terlihat bahwa hasil pengujian pada Gambar 6.2, banyaknya generasi 20 sampai dengan 160 mengalami peningkatan. Namun, pada populasi 180 mengalami penurunan, dan mengalami peningkatan kembali pada banyaknya generasi 200 dengan rata-rata nilai *fitness* tertinggi. Pada pengujian banyaknya generasi dengan nilai *fitness* yang paling optimal adalah pada ukuran 200 generasi.

6.1.3 Hasil Pengujian Kombinasi *Crossover Rate (cr)* dan *Mutation Rate(mr)*

Dalam pengukuran pengujian kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) yang tepat bertujuan untuk menghasilkan solusi optimal pada penjadwalan kapal penyeberangan Ketapang – Gilimanuk ditinjau dari rata-rata nilai *fitness* terbaik yang dihasilkan. Dengan menggunakan ukuran populasi yaitu sebanyak 180 dan banyaknya generasi yaitu sebanyak 200 sesuai dari hasil pengukuran pengujian sebelumnya. Kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) yang digunakan yaitu angka 0 sampai dengan 1. Tabel 6.3 merupakan hasil pengujian kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) yaitu sebagai berikut.

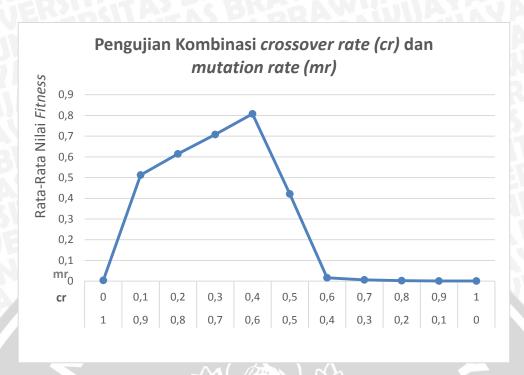
Tabel 6.3 Hasil Pengujian Kombinasi Crossover Rate dan Mutation Rate

Koml	binasi			5			itness		14			Rata- rata
Cr	mr	1	Per 2	3	in Kon	nbina: 5	6	an <i>mr</i> 7	yang 8	ке- 9	10	Nilai Fitness
		0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
		02	01	06	05	07	05	03	03	05	02	
		62	67	21	84	63	37	21	06	23	89	
		46	78	11	79	35	63	54	74	< 56	01	
1	0	71	52	80	53	87	44	34	84	02	73	0,00437
		91	34	12	21	78	08	08	66	09	41	8028
		60	89	42	63	62	60	36	25	42	04	
		10	93	23	74	59	21	01	76	40	04	
		49	28	60	26	54	50	28	68	83	62	
		0,0	1,0	0,0	1,0	1,0	1,0	0,0	1,0	0,0	0,0	
		24	00	12	00	00	00	27	00	12	47	
		39	00	34	00	00	00	77	00	34	61	
		02	00	56	00	00	00	77	00	56	90	0.54244
0,9	0,1	43	00	79	00	00	00	77	00	79	47	0,51244
		90	00	01	00	00	00	77	-00	01	61	7843
		24	00	23	00	00	00	77	00	23	90	
40		39	00	45	00	00	00	77	00	45	47	
Hit		00	00	60	00	00	00	70	00	60	60	
		0,0	1,0	0,0	0,0	1,0	1,0	1,0	1,0	1,0	0,0	141
	41	47	00	24	47	00	00	00	00	00	24	
		61	00	39	61	00	00	00	00	00	39	BRA
411		90	00	02	90	00	00	00	00	00	02	0.64.40
0,8	0,2	47	00	43	47	00	00	00	00	00	43	0,61440
		61	00	90	61	00	00	00	00	00	90	1858
		90	00	24	90	00	00	00	00	00	24	410
	E B	47	00	39	47	00	00	00	00	00	39	MATT
		60	00	00	60	00	00	00	00	00	00	

	MUL	0,0	1,0	0,0	1,0	0,0	1,0	1,0	1,0	1,0	1,0	LATIN
0,7		24	00	27	00	27	00	00	00	00	00	VV44
		39	00	77	00	77	00	00	00	00	00	
	0,3	02	00	77	00	77	00	00	00	00	00	
		43	00	77	00	77	00	00	00	00	00	0,70799
	ALLA	90	00	77	00	77	00	00	00	00	00	458
		24	00	77	00	77	00	00	00	00	00	4401
191		39	00	77	00	77	00	00	00	00	00	13,24
FAS		00	00	70	00	70	00	00	00	00	00	TIVI
		1,0	1,0	0,0	1,0	1,0	1,0	1,0	0,0	1,0	1,0	NUN
4		00	00	24	00	00	00	00	47	00	00	ATTA
	47-	00	00	39	00	00	00	00	61	00	00	19-4-16
	0,4	00	00	02	00	00	00	00	90	00	00	
0,6		00	00	43	00	00	00	00	47	00	00	0,80720
0,0	0, 1	00	00	90	00	00	00	00	61	00	00	0929
V		00	00	24	00	00	00	00	90	00	00	
		00	00	39	00	00	00	00	47	00	00	
		00	00	00	00	00	00	00	60	00	00	4
		00	00	00	00	00		00	00	00	00	
		0,0	1,0	1,0	0,0	0,0	0,0	1,0	0,0	1,0	0,0	
	0,5	17	00	00	47	47	24	00	24	-00	47	
		85	00	00	61	61	39	00	39	00	61	
		71	00	00	90	90	02	00	02	00	90	0.42004
0,5		42	00	00	47	47	43	00	43	00	47	0,42094
		85	00	00	61	61	90	00	90	00	61	9477
		71	00	00	90	90	24	00	24	00	90	
		42	00	00	47	47	39	00	39	00	47	
		80	00	00	60	60	00	00	00	00	60	
		0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
		08	47	12	17	16	12	16	12	09	12	
		62	61	34	85	39	34	39	34	90	34	
		06	90	56	71	34	56	34	56	09	56	
0,4	0,6	89	47	79	42	42	79	42	79	90	79	0,01661
450		65	61	01	85	62	01	62	01	09	01	6747
		51	90	23	71	29	23	29	23	90	23	
		72	47	45	42	50	45	50	45	09	45	
		41	60	60	80	80	60	80	60	90	60	
		0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	TORA
	0,7	06	05	08	05	06	07	06	06	06	04	P. R
		41	68	62	23	62	09	41	21	41	32	0,00630
0,3		02	18	06	56	25	21	02	11	02	90	2378
		56	18	89	02	16	98	56	80	56	04	2376
		41	18	65	09	55	58	41	12	41	32	MILE
	2 6	02	18	51	42	62	15	02	42	02	90	
		52	10	J±	74	02	13	102	74	02	50	

		56	18	72	40	91	60	56	23	56	04	Left V
		41	18	41	83	39	28	41	60	41	32	
0,2 0,	0,8	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,00240 584
		02	02	02	02	02	02	02	02	02	02	
		69	19	69	16	01	37	24	24	65	77	
		54	29	54	91	61	52	21	21	95	00	
		17	82	17	97	29	96	52	52	74	83	
		78	45	78	39	03	91	46	46	46	10	
		97	61	97	69	22	21	63	63	80	24	
		57	40	57	63	58	14	67	67	85	93	
		41	35	41	12	06	01	71	71	10	07	
TTH	V	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
	0,9	01	01	01	01	01	01	01	01	01	01	
		39	26	67	90	32	51	16	46	38	44	
		66	42	78	11	27	28	82	84	69	71	0.004.4-
0,1		48	22	52	40	51	59	24	28	62	78	0,00145
-		04	50	34	68	32	30	29	78	55	00	4627
		46	31	89	44	27	40	90	12	20	28	
		92	60	93	10	51	84	65	04	11	94	
		73	55	28	64	32	72	42	11	09	35	
		0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	
0	1	01	01	01	01	01	01	01	01	01	01	
		18	14	78	42	35	09	29	24	29	14	
		90	15	25	65	86	76	70	06	70	15	0.00430
		60	52	31	33	95	94	16	94	16	52	0,00129
		64	51	19	52	65	84	86	78	86	51	7235
		20	14	42	35	21	08	12	90	12	14	
		92	15	95	37	73	34	19	81	19	15	
		74	52	90	80	91	24	19	88	19	52	

Berdasarkan hasil pengujian ukuran populasi mengenai sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika pada Tabel 6.3, didapatkan bahwa grafik kombinasi *crossover rate (cr)* dan *mutation rate (mr)* adalah sebagai berikut.



Gambar 6.3 Grafik Hasil Pengujian Kombinasi Crossover Rate dan Mutation
Rate

Terlihat rata-rata nilai *fitness* yang didapat sangat bermacam-macam karena memang tidak ada ketetapan yang digunakan untuk parameter kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) dalam menghasilkan solusi yang optimal sesuai dengan permasalahan yang ingin diselesaikan. Penentuan kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) ini merupakan hal yang sulit (Mahmudy, Marian & Luong 2013d). Apabila nilai *crossover rate* (*cr*) terlalu tinggi dan nilai *mutation rate* (*mr*) terlalu rendah maka tidak dapat mengeksplorasi area lain dalam suatu pencarian, sedangkan apabila nilai *crossover rate* (*cr*) terlalu rendah dan nilai *mutation rate* (*mr*) terlalu tinggi maka secara efektif tidak dapat mengeksploitasi area lain dalam suatu pencarian (Mahmudy, 2013). Pada Gambar 6.3 grafik kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) memiliki nilai rata-rata nilai *fitness* hampir sama pada 0,4:0,6,0,3:0,7,0,2:0,8,0,1:0,9, dan paling rendah kombinasi *cr* dan *mr* pada 0:1. Pada pengujian kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) dengan nilai *fitness* yang paling optimal adalah pada *crossover rate* (*cr*)=0,6 dan *mutation rate* (*mr*)=0,4.

6.1.4 Hasil Pengujian Parameter Terbaik

Pengujian parameter terbaik bertujuan untuk mengetahui seberapa optimal solusi yang didapatkan dalam permasalahan penjadwalan kapal penyeberangan Ketapang – Gilimanuk. Parameter terbaik didapat dari pengujian sebelumnya yaitu dengan menggunakan ukuran populasi sebanyak 180, banyaknya generasi sebanyak 200, dan kombinasi crossover rate (cr)=0,6 dan mutation rate (mr)=0,4 yang memiliki rata-rata nilai fitness tertinggi. Maka didapat individu terbaik dengan nilai fitness yaitu sebesar 1, dengan tidak terdapat pelanggaran 1, 2, ataupun 3, hard constrain maupun soft constraint, yaitu masing-masing kapal

memiliki jatah porsi adil yang telah ditentukan, tidak ada nama kapal yang sama dalam sehari, serta kapal docking tidak beroperasi. Individu terbaik tersebut akan menjadi sebuah solusi dalam permasalahan penjadwalan kapal penyeberangan Ketapang - Gilimanuk. Dari hasil individu terbaik tersebut Sehingga dapat disimpulkan bahwa dengan menggunakan metode algoritma genetika dapat menyelesaikan permasalahan penjadwalan kapal penyeberangan Ketapang -Gilimanuk dengan optimal. Berikut Gambar 6.4, merupakan tempilan penjadwalan kapal yang dihasilkan.



Gambar 6.4 Penjadwalan Kapal yang Dihasilkan

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan implementasi dan hasil pengujian mengenai sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika dapat diambil beberapa kesimpulan yakni sebagai berikut.

- 1. Sistem optimasi penjadwalan kapal penyeberangan Ketapang Gilimanuk dapat diimplementasikan dengan menggunakan algoritma genetika yaitu dengan menggunakan representasi kromosom permutasi berbasis kode kapal yang diacak secara acak sesuai dengan porsi yang telah ditentukan, memiliki panjang kromosom dengan panjang interval [1.......180]. Metode yang digunakan dengan menerapkan algoritma genetika yaitu dengan menggunakan metode crossover yaitu one-cut point crossover, metode mutasi menggunakan reciprocal exchange mutation, dan metode seleksi menggunakan elitism selection.
- 2. Untuk menentukan parameter yang tepat yaitu dilakukan pengujian mengenai pengujian ukuran populasi, pengujian banyaknya generasi, serta pengujian kombinasi crossover rate (cr) dan mutation rate (mr). Dari hasil pengujian tersebut didapat rata-rata nilai fitness tertinggi yaitu pada ukuran populasi 180, banyaknya generasi 200, serta kombinasi crossover rate (cr)= 0,6 dan mutation rate (mr)= 0,4. Dari hasil tersebut, menentukan parameter algoritma genetika sangat berpengaruh terhadap solusi yang didapat yaitu dengan memilih individu terbaik yang memiliki nilai fitness tertinggi untuk dijadikan sebuah solusi dalam permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk. Semakin besar nilai fitness yang dihasilkan dalam suatu kromosom, maka semakin baik pula solusi yang didapatkan. Sedangkan semakin kecil nilai fitness yang dihasilkan dalam suatu kromosom, maka semakin buruk solusi yang didapatkan.
- 3. Dari hasil pengujian parameter terbaik, didapat individu terbaik memiliki nilai fitness 1 dengan tidak terdapat nilai pelanggaran 1, 2, ataupun 3, hard constraint ataupun soft constraint yang telah ditentukan. Sehingga dapat disimpulkan bahwa algoritma genetika dapat menyelesaikan permasalahan penjadwalan kapal penyeberangan Ketapang Gilimanuk dengan optimal.

7.2 Saran

Sistem optimasi penjadwalan kapal penyeberangan Ketapang – Gilimanuk menggunakan algoritma genetika dapat dikembangkan dengan menggunakan metode mutasi, *crossover*, dan seleksi yang lain untuk mendapatkan solusi yang bervariasi dan lebih baik. Serta pemilihan parameter yang tepat sehingga akan didapatkan hasil yang lebih optimal dalam waktu yang lebih singkat.

DAFTAR PUSTAKA

Atlas Global. Solo: CV Buana Raya.

Berlianty, Intan & Arifin, Miftahol. 2010. Teknik-teknik Optimasi Heuristik. Yogyakarta: Graha Ilmu.

Desiani, Anita & Arhami, Muhammad. 2005. Konsep Kecerdasan Buatan. Yogyakarta: ANDI.

Devi, O. C., Mahmudy, W. F., & Setiawan B. D., 2015. Penerapan Algoritma Genetika untuk Penjadwalan Asisten Praktikum. S1. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, vol. 5, no. 11.

Ginting, Rosnani. 2009. Penjadwalan Mesin. Yogyakarta: Graha Ilmu.

Girinata, I. M. C., Mahmudy, W. F., 2016. Optimasi Penjadwalan Karyawan Training Divisi Food and Beverage Menggunakan Algoritma Genetika Studi Kasus Ibis Benoa Bali. S1. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, vol. 5, no. 13.

Hariyanto, Bambang. 2011. *Esensi- Esensi Bahasa Pemrograman Java.* Bandung: INFORMATIKA.

Ilmi, R. R., Mahmudy, W. F., & Ratnawati, D. E., 2015. *Optimasi Penjadwalan Perawat Menggunakan Algoritma Genetika*. S1. *DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya*, vol. 5, no. 13.

Kusumadewi, Sri. 2003. Artificial Intelligence (Teknik dan Aplikasinya). Yogyakarta: Graha Ilmu.

Mahmudy, Wayan Firdaus. 2013. *Modul Algoritma Evolusi*. Program Teknologi Informasi dan Ilmu Komputer. Malang: Universitas Brawijaya.

Ramadhani, Citra. 2015. *Dasar Algoritma dan Struktur Data dengan Bahasa Java.* Yogyakarta: ANDI.

Sari, D. D. P., Mahmudy, W. F., & Ratnawati D. E., 2015. Optimasi Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika (Studi Kasus: SMPN 1 Gondang Mojokerto). S1. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, vol. 5, no. 13.

Syarif, Admi. 2014. *Algoritma Genetika Teori dan Aplikasi Edisi 2*. Yogyakarta: Graha Ilmu.

Susantono, Bambang. 2013. *Transportasi & Investasi*. Jakarta: PT Kompas Media Nusantara.

Sutojo, T., Mulyanto E., & Suhartono V., 2011. *Kecerdasan Buatan*. Yogyakarta: ANDI.

Suyanto. 2014. Artificial Intelligence. Bandung: INFORMATIKA.

Wismulyani, Endar. 2008. *Transportasi di Indonesia dari Masa ke Masa*. Klaten: Penerbit Cempaka Putih.

Zukhri, Zainuddin. 2014. Algoritma Genetika Metode Komputasi Evolusioner untuk Menyelesaikan Masalah Optimasi. Yogyakarta: ANDI.

Zulfa, I., Mahmudy, W. F., & Setiawan, B. D., 2015. *Optimasi Jadwal Mengajar Asisten Laboratorium Menggunakan Algoritma Genetika*. S1. *DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya*, vol. 5, no. 11.

http://www.indonesia.go.id/in/sekilas-indonesia/geografi-indonesia diakses tanggal 25 Oktober 2015

http://stackoverflow.com/questions/1519736/random-shuffling-of-an-array potongan source code Fisher-Yates shuffle diakses tanggal 14 Mei 2016

http://www.programming-algorithms.net/article/43676/Fisher-Yates-shuffle potongan source code Fisher-Yates shuffle diakses tanggal 14 Mei 2016

