

OPTIMASI FUZZY TSUKAMOTO DENGAN ALGORITMA GENETIKA

STUDI KASUS PERAMALAN PERMINTAAN BARANG SEMEN

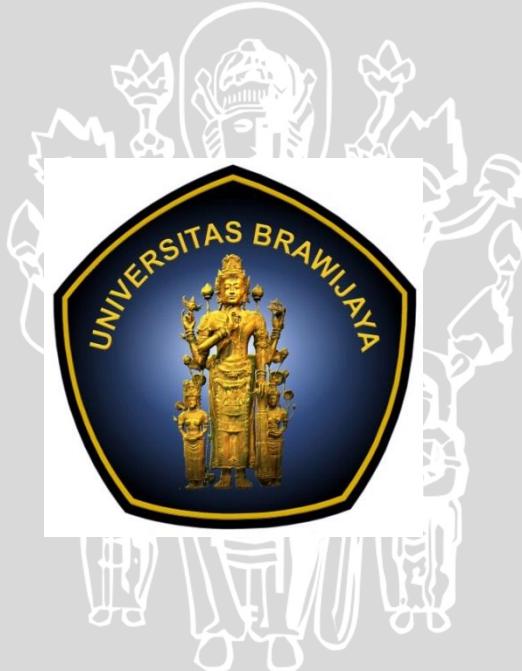
SKRIPSI

Untuk Memenuhi Sebagian Persyaratan
Memperoleh Gelar Sarjana Komputer

Disusun oleh:

Nama: Rifki Setya Armanda

NIM: 115090607111002



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

PENGESAHAN

OPTIMASI FUZZY TSUKAMOTO DENGAN ALGORITMA GENETIKA
STUDI KASUS PERAMALAN PERMINTAAN BARANG SEMEN

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Rifki Setya Armanda
NIM. 11509050607111002

Skripsi ini telah diuji dan dinyatakan lulus pada

10 November 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

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP. 19710518 200312 1 001

PERNYATAAN ORISINALITAS

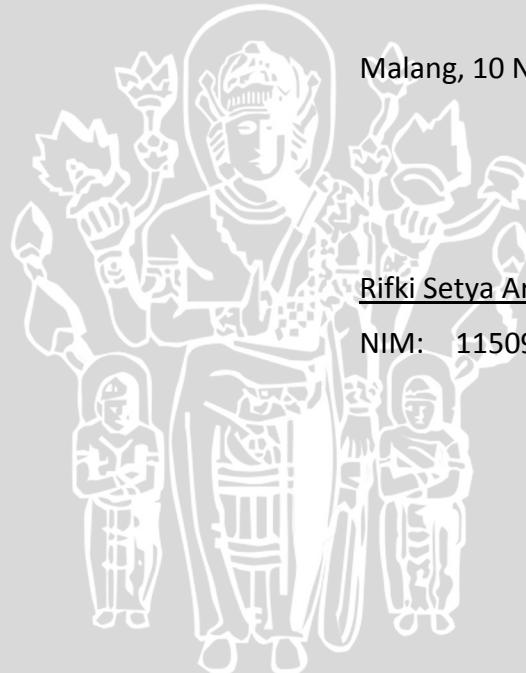
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan 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 unsur-unsur 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, 10 November 2016

Rifki Setya Armando

NIM: 115090607111002



KATA PENGANTAR

Puji syukur kehadirat Allah SWT, karena atas rahmat dan karunia yang telah diberikan penulis dapat menyelesaikan menyelesaikan naskah Tugas Akhir yang berjudul “OPTIMASI FUZZY TSUKAMOTO DENGAN ALGORITMA GENETIKA STUDI KASUS PERAMALAN PERMINTAAN BARANG SEMEN”.

1. Bapak Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D. selaku dosen pembimbing dalam pembuatan naskah Tugas Akhir.
2. Bapak Mukromin, selaku Manajer SDM Koperasi Warga Semen Gresik (KWSG) yang telah memberikan izin sehingga penulis dapat melaksanakan kegiatan pengambilan data Tugas Akhir di Koperasi Warga Semen Gresik (KWSG).
3. Bapak Teguh Suhartono, selaku kepala bagian pengembangan sistem dan informasi di Koperasi Warga Semen Gresik (KWSG) Gresik.
4. Kedua orang tua serta keluarga penulis yang telah memberikan semangat, doa serta dukungannya.
5. Rekan-rekan penulis di Universitas Brawijaya atas dorongan semangat dan bantuan ilmu yang diberikan.
6. Semua pihak yang telah membantu menyelesaikan naskah Tugas Akhir ini yang tidak dapat penulis sebut satu persatu.

Semoga segala bantuan yang telah diberikan kepada penulis mendapat balasan yang setimpal dari Tuhan Yang Maha Esa. Penulis menyadari bahwa masih banyak kekurangan dalam penyusunan naskah Tugas Akhir ini baik dalam teknik penyajian materi maupun pembahasan. Untuk itu saran dan kritik yang sifatnya membangun sangat penulis harapkan. Semoga Naskah Tugas Akhir ini dapat bermanfaat dan dapat digunakan sebagaimana Semestinya.

Malang, 10 November 2016

Penulis

rifikisetyaarmanda@gmail.com

ABSTRAK

Peramalan merupakan seni atau suatu ilmu yang digunakan dalam memperkirakan kejadian untuk dimasa depan. Hal tersebut dilakukan dengan melibatkan suatu pengambilan data sebelumnya atau data pada masa lalu yang kemudian menempatkannya pada masa yang akan datang dengan suatu bentuk model yang sistematis. Perencanaan kapasitas produksi yang fleksibel adalah dengan melakukan perencanaan kapasitas produksi yang sesuai dengan besarnya kebutuhan permintaan. Apabila kapasitas produksi yang direncanakan terlalu besar sehingga melebihi kebutuhan permintaan yang sebenarnya, maka perusahaan tersebut akan mengalami tingkatan kerugian yang cukup besar. Dengan adanya suatu peramalan dalam permintaan barang, maka suatu perusahaan dapat mencapai tujuan serta mengambil keputusan dalam memproduksi barang agar dapat memenuhi keinginan permintaan pasar.

Pada skripsi ini, metode yang digunakan dalam peramalan permintaan barang adalah Algoritma Genetika dan Fuzzy Inferensi System Tsukamoto. Tahapan dalam perhitungan metode Algoritma Genetika dan Fuzzy Tsukamoto adalah dengan mengambil data permintaan pada setiap minggunya secara random, yang kemudian nilai tersebut akan dijadikan sebagai batasan-batasan pada setiap kriteria yang digunakan. Representasi kromosom yang digunakan adalah *real code genetic algorithm* dimana representasi tersebut dibangkitkan secara random dengan interval tertentu. Dari pengujian yang dilakukan, sistem mampu memberikan nilai terbaik pada jumlah populasi 80, jumlah generasi 120, kombinasi cr 0.3 dan mr 0.7. Pada metode Algoritma Genetika dan Fuzzy Tsukamoto dapat memberikan hasil nilai error yang lebih rendah dibandingkan dengan menggunakan metode Fuzzy Tsukamoto.

Kata Kunci: *Algoritma Genetika, Peramalan Permintaan Barang, Fuzzy Inferensi System Tsukamoto*

ABSTRACT

Forecasting is an art or a science that used to estimating events in the future. That matter can be done by involving a previous data retrieval or data on the past and then put it in the future with a systematic model form. Planning of flexible production capacity is by doing an appropriate production capacity planning based on the amount of demand needs. When the planned production capacity is too large thus exceeding the needs of actual demand, then the company will run into a level of significant losses. With the existence of a forecasting in the demand for goods, so a company can achieve the goals and take decisions in the production of goods so that can fulfill market demand.

In this essay, the methods used in forecasting demand for goods is a Genetic Algorithm and Fuzzy Inference System Tsukamoto. Stages in the computation method of the Genetic Algorithm and Fuzzy Tsukamoto is by taking the data requests randomly every week, afterwards the value will be used as boundaries on each of the criteria used. The representation of chromosome used is real code genetic algorithm where the representation generated randomly with a certain interval. Based on the tests performed, the system is able to provide the best value in populations is 80, the amount of generation is 120, and a combination of cr and mr are 0.3 and 0.7. Genetic Algorithm and Fuzzy Tsukamoto methods can give results of error value is lower than just using Fuzzy Tsukamoto.

Keywords: *Genetic Algorithms, Goods Demand Forecasting, Fuzzy Inference System Tsukamoto*



DAFTAR ISI

PENGESAHANii
PERNYATAAN ORISINALITASiii
KATA PENGANTAR.....	.iv
ABSTRAK.....	.v
ABSTRACTvi
DAFTAR ISIvii
DAFTAR TABEL.....	.x
DAFTAR GAMBAR.....	.xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah	2
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Peramalan	7
2.2.1 Jenis-Jenis Peramalan	7
2.3 Permintaan	8
2.4 Algoritma Genetika	8
2.4.1 Langkah Penyelesaian Algoritma Genetika	9
2.4.2 Penerapan Algoritma Genetika.....	10
2.4.3 Membangun Generasi Awal.....	12
2.4.4 Reprentasi Chromosom	12
2.4.5 Reproduksi	13
2.4.5.1 Proses Crossover	13
2.4.5.2 Mutasi	13
2.4.6 Fitnes Cost.....	14
2.4.7 Pengurutan (sorting)	14



2.4.8 Seleksi.....	14
2.4.9 Tahap Pengulangan.....	15
2.4.10 Diagram Alir Algoritma Genetika	15
2.5 Fuzzy Inferensi System Tsukamoto.....	16
BAB 3 METODOLOGI	20
3.1 Tahapan Penelitian	20
3.1.1 Studi Literatur	21
3.1.2 Pengambilan Data	21
3.1.3 Mengolah Data.....	21
3.1.4 Perancangan Sistem.....	22
3.1.5 Implementasi	22
3.1.6 Pengujian dan Analisis.....	22
3.1.7 Pengambilan Kesimpulan.....	23
3.2 Teknik Pengumpulan Data.....	23
3.3 Algoritma yang Digunakan.....	23
3.4 Kebutuhan Sistem	23
3.5 Pengujian Algoritma Genetika	23
3.5.1 Pengujian Kombinasi <i>Crossover Rate</i> dan <i>Mutation Rate</i>	24
3.5.2 Pengujian Ukuran Populasi	25
3.5.3 Pengujian Ukuran Generasi.....	25
3.5.4 Pengujian Jumlah Data.....	26
BAB 4 Perancangan	27
4.1 Formulasi Permasalahan.....	27
4.1.1 Contoh Permasalahan	28
4.2 Siklus Algoritma Genetika	34
4.3 Siklus Penyelesaian Masalah Menggunakan Algoritma Genetika	36
4.3.1 Repräsentasi Kromosom dan Perhitungan Fitness.....	36
4.3.2 Inisialisasi Populasi Awal	40
4.3.3 Reproduksi	41
4.3.3.1 Perhitungan Crossover.....	41
4.3.3.2 Perhitungan Mutasi.....	42
4.3.4 Evaluasi dan Seleksi.....	42



4.4 Perancangan User Interface	44
4.4.1 Rancangan <i>User Interface</i> Halaman Input Data Permintaan dan Aturan Fuzzy.....	44
4.4.2 Rancangan <i>User Interface</i> Halaman Perhitungan dan Hasil	45
4.4.3 Rancangan <i>User Interface</i> Halaman Perbandingan Hasil.....	46
BAB 5 Implementasi	48
5.1 Implementasi Program	48
5.1.1 Proses Ambil Data dan Aturan Fuzzy	48
5.1.2 Proses Inisialisasi Kromosom	49
5.1.3 Proses Generate Random Parent Awal.....	49
5.1.4 Proses Crossover	49
5.1.5 Proses Mutation	50
5.1.6 Proses Evaluasi Seleksi	51
5.1.7 Proses Perhitungan Error dan Fitness.....	52
5.1.8 Proses Perhitungan Fuzzyifikasi	52
5.1.9 Proses Inferensi.....	54
5.1.10 Proses Perhitungan Deffuzyifikasi.....	54
5.1.11 Proses Perhitungan α -predikat $x z$	54
5.1.12 Proses Perhitungan Tsukamoto	55
5.2 Implementasi User Interface	55
5.2.1 Halaman Input Data dan Aturan Fuzzy	55
5.2.2 Halaman Input Data dan Aturan Fuzzy	56
5.2.3 Halaman Perbandingan Hasil	56
BAB 6 Pengujian dan Analisis	58
6.1 Hasil dan Analisa Uji Coba Ukuran Populasi	58
6.2 Hasil dan Analisa Uji Coba Banyaknya Generasi.....	59
6.3 Hasil dan Analisa Uji Coba Kombinasi Cr dan Mr.....	61
6.4 Hasil dan Analisa Uji Coba Menggunakan Parameter Terbaik	62
BAB 7 Penutup	65
7.1 Kesimpulan.....	65
7.2 Saran	65
DAFTAR PUSTAKA.....	67



DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	6
Tabel 2.1 Kajian Pustaka (Lanjutan)	7
Tabel 2.2 Contoh Pengkodean Biner.....	10
Tabel 2.3 Contoh Pengkodean Permutasi.....	12
Tabel 2.4 Contoh Exchange Mutation.....	14
Tabel 2.5 Aturan Fuzzy Tsukamoto	18
Tabel 2.5 Aturan Fuzzy Tsukamoto (Lanjutan).....	19
Tabel 3.1 Perancangan Uji Coba Kombinasi Crossover Rate dan Mutation Rate	24
Tabel 3.2 Perancangan Uji Coba Ukuran Populasi	25
Tabel 3.3 Perancangan Uji Coba Ukuran Generasi	25
Tabel 3.3 Perancangan Uji Coba Ukuran Generasi (Lanjutan).....	26
Tabel 3.4 Perancangan Uji Coba Jumlah Data	26
Tabel 4.1 Data Permintaan Barang	27
Tabel 4.1 Data Permintaan Barang (Lanjutan).....	28
Tabel 4.2 Contoh Permasalahan	28
Tabel 4.3 Fuzzyifikasi nilai kriteria	30
Tabel 4.4 Basis Aturan.....	30
Tabel 4.4 Basis Aturan (Lanjutan)	31
Tabel 4.5 Konversi Fuzzifikasi pada Basis Aturan.....	31
Tabel 4.6 Hasil Prediksi Permintaan.....	32
Tabel 4.7 Data Asli Permintaan.....	32
Tabel 4.8 Error Fuzzy Tsukamoto	33
Tabel 4.9 Error Fuzzy Tsukamoto	39
Tabel 4.10 Hasil Perhitungan Fitness	40
Tabel 4.11 Inisialisasi Populasi Awal Secara Random	40
Tabel 4.12 Inisialisasi Populasi Awal Secara Berurutan	41
Tabel 4.13 Crossover Extends Intermediate	41
Tabel 4.14 Crossover Extends Intermediate (Lanjutan).....	42
Tabel 4.15 Mutation Exchange Point	42
Tabel 4.16 Evaluasi.....	42

Tabel 4.16 Evaluasi (Lanjutan)	43
Tabel 4.18 Seleksi Elitsm	43
Tabel 5.1 Source Code Ambil Data.....	48
Tabel 5.2 Source Code Inisialisasi Kromosom.....	49
Tabel 5.3 Source Code Generate Random Parent Awal	49
Tabel 5.4 Source Code Crossover.....	50
Tabel 5.5 Source Code Mutasi	51
Tabel 5.6 Source Code Evaluasi Seleksi.....	51
Tabel 5.7 Source Code Error dan Fitness	52
Tabel 5.8 Source Code Fuzzyifikasi	52
Tabel 5.9 Source Code Inferensi	54
Tabel 5.10 Source Code Deffuzyifikasi	54
Tabel 5.11 Source Code Perhitungan α -predikat $x z$	55
Tabel 5.12 Source Code Perhitungan Tsukamoto.....	55
Tabel 6.1 Hasil Pengujian Ukuran Populasi.....	58
Tabel 6.2 Hasil Pengujian Jumlah Generasi.....	60
Tabel 6.3 Hasil Pengujian Kombinasi Cr dan Mr	61
Tabel 6.4 Hasil Pengujian Menggunakan Parameter Terbaik	63
Tabel 6.5 Contoh Hasil Perhitungan Menggunakan Kromosom Terbaik.....	63



DAFTAR GAMBAR

Gambar 2.1 Pengkodean Pohon	11
Gambar 2.2 Diagram Alir Algoritma Genetika	15
Gambar 2.3 Mapping input dan output	16
Gambar 3.1 Diagram Waterfall	20
Gambar 4.1 Diagram Alir Proses Algortima Genetika.....	35
Gambar 4.2 Contoh Reprentasi Kromosom [Pi] Random	36
Gambar 4.3 Contoh Reprentasi Kromosom [Pi] Berurutan	37
Gambar 4.4 Grafik Kriteria T3	37
Gambar 4.5 Grafik Kriteria T2	38
Gambar 4.6 Grafik Kriteria T1	38
Gambar 4.7 Grafik Kriteria T	39
Gambar 4.8 Rancangan <i>User Interface</i> Halaman Input Data Permintaan dan Aturan Fuzzy.....	44
Gambar 4.9 Rancangan <i>User Interface</i> Halaman Perhitungan Dan Hasil.....	45
Gambar 4.10 Rancangan <i>User Interface</i> Halaman Perbandingan Hasil.....	46
Gambar 5.1 Implementasi Halaman Input Data dan Aturan Fuzzy	56
Gambar 5.2 Implementasi Halaman Perhitungan dan Hasil.....	56
Gambar 5.3 Implementasi Halaman Perbandingan Hasil	57
Gambar 6.1 Grafik Hasil Uji Coba Ukuran Populasi	59
Gambar 6.2 Grafik Hasil Uji Coba Jumlah Generasi	60
Gambar 6.3 Grafik Hasil Uji Coba Kombinasi Cr dan Mr.....	62

BAB 1 PENDAHULUAN

1.1 Latar belakang

Pada perkembangan jaman yang terjadi pada era globalisasi saat ini membawa dampak tersendiri dalam dunia bisnis, khususnya pada dunia perindustrian. Perkembangan yang terjadi dalam dunia industri mengakibatkan terjadinya persaingan yang sangat ketat. Maka dari itu, setiap perusahaan dituntut untuk dapat mempersiapkan kekuatan dalam menghadapi suatu persaingan pada setiap perusahaan maupun keinginan pada konsumen.

Pada industri yang bergerak pada bidang industri memiliki tingkatan persaingan yang sangat ketat. Persaingan tersebut merupakan tuntutan perhatian dari setiap perusahaan dalam meningkatkan kemampuan bersaing dengan perusahaan lainnya, dengan tujuan agar produk yang dipasarkan dapat menguasai angka pasar. Sehingga pada setiap perusahaan mampu memenuhi barang produksi berdasarkan permintaan pasar, serta setiap perusahaan harus dapat lebih efektif dan efisien dalam memenuhi permintaan pasar.

Dalam menyikapi permasalahan tersebut setiap perusahaan mampu mengetahui permintaan barang produksi yang akan datang. Peramalan suatu permintaan barang merupakan hal yang diperhitungkan untuk dimasa selanjutnya atau masa yang akan datang berdasarkan data-data permintaan yang dimiliki sebelumnya. Dengan adanya suatu peramalan dalam permintaan barang, maka suatu perusahaan dapat mencapai tujuan serta mengambil keputusan dalam memproduksi barang agar dapat memenuhi keinginan permintaan pasar.

Menurut Nasution dan Prasetyawan (2008), peramalan permintaan diharapkan akan terealisir untuk jangka waktu tertentu pada masa yang akan datang, peramalan permintaan tersebut akan menjadi masukan yang sangat penting dalam mengambil keputusan perencanaan dan pengendalian suatu perusahaan. Namun dalam kegiatan peramalan memerlukan penerapan metode yang optimal dalam menghasilkan suatu peramalan permintaan. Hal tersebut bertujuan agar dapat mengetahui tingkat permintaan yang akan datang dan meminimumkan kesalahan pada perhitungan peramalan. Apabila nilai dalam peramalan yang kurang tepat, maka akan menyebabkan ketidaksesuaian kuantitas dan kualitas produk dengan permintaan pasar.

Pada penelitian ini ditujukan untuk mengetahui peramalan permintaan barang produksi semen. Sistem yang akan dibangun dalam penelitian ini mampu menghasilkan nilai akurasi yang tepat dalam meramalkan suatu permintaan barang dengan menggunakan metode Algoritma Genetika dan Fuzzy Tsukamoto. sebab metode Algoritma Genetika dan Fuzzy Tsukamoto merupakan metode yang cocok dalam memberikan hasil yang optimal dalam permasalahan yang cukup kompleks.

Penggunaan pada kriteria untuk metode Fuzzy Tsukamoto berdasarkan data-data permintaan dalam kurun waktu 25 minggu. Untuk kriteria dalam

penelitian ini adalah 3 Minggu sebelumnya (T3), 2 Minggu sebelumnya (T2), dan 1 Minggu sebelumnya (T1). Akan tetapi dalam penggunaan metode Fuzzy Tsukamoto nilai akurasi yang didapatkan kurang optimal, sebab metode Fuzzy Tsukamoto hanya bergantung pada aturan yang berbentuk IF-THEN yang direpresentasikan dengan himpunan fuzzy. Agar dapat menghasilkan nilai yang sesuai maka dalam fungsi keanggotaan tersebut dioptimasi ke dalam metode Algoritma Genetika.

Beberapa penelitian sebelumnya telah membahas mengenai permasalahan optimasi pada bahan baku. Misalkan pada Mahmudah, Irhamah, dan Mumpuni (2011) yang telah menerapkan Algoritma Genetika dalam kasus optimasi persediaan Multi-Item Fuzzy EOQ. Pada penelitian yang sejenis dilakukan oleh Indroprasto, dan Suryani dengan permasalahan pengendalian persediaan produk dengan metode EOQ menggunakan Algoritma Genetika yang bertujuan agar dapat mengifisiensikan biaya persediaan. Selain itu penelitian yang dilakukan oleh Palupi, dkk (2011) juga membahas persediaan multi barang dalam proses produksi bahan baku pada perusahaan PT UWBM Sidoarjo.

Beberapa penelitian yang berkaitan tentang peramalan atau prediksi dengan menggunakan Algoritma Genetika telah dilakukan oleh Rahmi (2015) yang menerapkan Algoritma Genetika dalam memprediksi harga saham, pada penelitian tersebut dapat menunjukkan bahwa Algoritma Genetika mampu memberikan hasil nilai prediksi yang mendekati nilai atau data aslinya. Kemudian pada penelitian berikutnya dilakukan oleh Permatasari, dkk (2015) yang membahas tentang prediksi konsumsi Kwh beban listrik menggunakan Algoritma Genetika, pada penelitian tersebut bahwa Algoritma Genetika mampu memberikan hasil prediksi yang lebih baik.

Berdasarkan permasalahan yang telah dijelaskan, maka penulis menentukan judul pada skripsi ini adalah “Optimasi Fuzzy Tsukamoto dengan Algoritma Genetika Studi Kasus Peramalan Permintaan Barang Semen”. Pada penelitian ini, obyek yang di tuju adalah jumlah permintaan barang produksi semen di gudang pada perusahaan Koperasi Semen Gresik, mengingat bahwa persaingan pada bidang industri semakin ketat.

1.2 Rumusan masalah

Berdasarkan paparan latar belakang tersebut, maka rumusan masalah adalah sebagai berikut :

1. Apakah penerapan metode Algoritma Genetika dan Fuzzy Tsukamoto dapat memberikan nilai error yang lebih rendah dalam hal peramalan permintaan barang?
2. Bagaimana menetukan parameter yang tepat dalam menggunakan metode Algoritma Genetika dan Fuzzy Tsukamoto dalam peramalan permintaan barang?
3. Apakah hasil dari metode Algoritma Genetika mampu menghasilkan nilai optimal dibandingkan dengan menggunakan Fuzzy Tsukamoto?



1.3 Tujuan

Tujuan dalam penilitian ini adalah sebagai berikut :

1. Mengetahui permintaan untuk minggu selanjutnya berdasarkan data permintaan pada minggu-minggu sebelumnya
2. Menetukan pengadaan suatu barang produksi berdasarkan peramalan jumlah permintaan.

1.4 Manfaat

Manfaat dalam Penilitian ini dapat diperoleh sebagai berikut :

1. Memberikan suatu informasi yang akurat bagi perusahaan Koperasi Semen Gresik sebagai bahan dalam mempertimbangkan persediaan barang produksi semen berdasarkan prediksi permintaan.
2. Mendapatkan suatu solusi yang optimal dari perhitungan dengan metode Algoritma Genetika dan Fuzzy Tsukamoto.
3. Dapat menentukan parameter algoritma yang tepat dalam masalah optimasi.
4. Mendapatkan representasi kromosom yang sesuai dan efisien dalam hal menentukan batasan-batasan nilai kriteria yang dibutuhkan.

1.5 Batasan masalah

Batasan masalah pada penilitian ini adalah sebagai berikut :

1. Permintaan : Permintaan barang yang dimaksud adalah permintaan barang produksi Semen dari suplier pada gudang Koperasi Semen Gresik yang berada di daerah Jawa Barat.
2. Barang : Barang produksi yang dimaksud adalah bahan baku Semen yang melakukan pemasaran pada provinsi Jawa Barat.
3. Bahasa Pemrograman : Bahasa pemrograman yang akan digunakan dalam aplikasi ini adalah Java.
4. Data : Data yang akan digunakan merupakan jumlah permintaan konsumen terhadap produksi barang semen pada Koperasi Semen Gresik disetiap minggunya (dalam waktu 25 minggu pertama) pada tahun 2015.

1.6 Sistematika pembahasan

Agar dapat mencapai tujuan yang diharapkan, maka sistematika pembahasan yang disusun dalam tugas akhir ini adalah sebagai berikut :

I. BAB 1 Pendahuluan

Pada bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta metodologi penulisan.

II. BAB 2 Landasan Kepustakaan

Pada bab ini berisi tentang tinjauan pustaka dan dasar teori yang diambil dari sumber pustaka dan referensi yang berhubungan dengan optimasi pesediaan barang dan Algoritma Genetika serta teori untuk menyelesaikan suatu permasalahan ini.

III. BAB 3 Metodelogi

Pada bab ini berisi tentang pembahasan metode yang digunakan dalam penulisan yang terdiri dari tahapan penelitian mulai dari proses pengambilan data dengan cara observasi dan wawancara. Metode analisis data yang digunakan dalam penilitian ini dengan Algoritma Genetika. Algoritma Genetika memiliki kemampuan dalam menyelesaikan berbagai masalah yang cukup kompleks dalam menghadapi masalah optimasi.

IV. BAB 4 Perancangan

Pada bab ini berisi tentang pembahasan perancangan yang digunakan oleh sistem peramalan permintaan barang Semen dengan menggunakan metode Algoritma Genetika dan Fuzzy Tsukamoto yang meliputi perancangan *user interface* dan *perancangan pengujian*.

V. BAB 5 Implementasi

Pada bab ini akan mengulas penjelasan mengenai implementasi dari metode Algoritma Genetika dan Fuzzy Tsukamoto pada studi kasus peramalan permintaan barang Semen, yang meliputi *user interface* dan *source code*.

VI. BAB 6 Pengujian dan Analisis

Pada bab ini berisi tentang penjelasan mengenai proses pengujian dan hasil yang dilakukan oleh sistem peramalan permintaan barang Semen dengan menggunakan metode Algoritma Genetika dan Fuzzy Tsukamoto dalam mendapatkan nilai optimal.

VII. BAB 7 Penutup

Pada bab ini berisikan tentang kesimpulan yang didapatkan pada pembuatan dan pengujian yang dikembangkan dalam skripsi ini, serta saran yang akan digunakan untuk dapat dikembangkan lebih lanjut.

BAB 2 LANDASAN KEPUSTAKAAN

Pada Bab ini akan menjelaskan tentang landasan pustaka dan dasar teori yang mendukung dalam penilitian yang dilakukan. Beberapa refensi yang telah diperoleh memiliki tujuan yaitu untuk mendapatkan sejumlah informasi terkait dengan penelitian yang dilakukan diantaranya adalah kajian pustaka, teori tentang peramalan dan teori tentang Algoritma Genetika.

2.1 Kajian Pustaka

Penelitian yang membahas tentang rekomendasi dan optimasi telah banyak dilakukan oleh beberapa pakar dan beberapa peneliti dengan objek yang berbeda-beda. Pada penelitian ini, menggunakan beberapa refensi yang relevan sebagai rujukan proses penelitian.

Penelitian yang dilakukan Panharesi dan Mahmudy (2015) yang menerapkan Algoritma Genetika untuk dapat meminimalisir biaya dalam distribusi barang. Tujuan dalam penelitian ini adalah agar dapat memperoleh biaya minimal, dengan menentukan rute yang akan ditempuh oleh truck dengan meminimalkan sisa muatan truck. Selain itu penelitian ini juga menggunakan pemilihan truck yang tepat yang akan digunakan dalam proses ditribusi barang. Penelitian ini menggunakan representasi kromosom dengan permutasi 2 segmen. Segmen pertama merupakan permutasi untuk agen dan segmen kedua merupakan permutasi untuk truk. Panjang kromosom tergantung banyaknya jumlah agen dan truk dalam sebuah perusahaan. Untuk mengukur solusi tersebut maka diperlukan perhitungan *fitness* yang didapatkan dari total biaya dan sisa muatan truk. Dari hasil uji coba, dengan ukuran populasi 140, banyaknya generasi 50, nilai cr 0.4 dan nilai mr 0.6 maka akan didapatkan rata-rata *fitness* terbaik. Selain itu dengan menggunakan persentase untuk tiap jenis mutasi sebesar 100% untuk mutasi segmen 1, 0% mutasi segmen 2 dan 0% mutasi kedua segmen maka akan mendapatkan rata – rata *fitness* terbaik. Hasil dari pengujian yang dilakukan, Algoritma Genetika mampu memberikan solusi yang optimal dengan total biaya terkecil dan dengan sisa muatan terkecil.

Penelitian selanjutnya, yang dilakukan oleh Indroprasto, dan Suryani (2012) dengan menerapkan metode EOQ menggunakan Algoritma Genetika dalam pengendalian persediaan produk. Tujuan dalam penelitian ini adalah dapat membantu PT. XYZ dalam mempertimbangkan dan menentukan kebijakan pengendalian persediaan barang agar dapat berjalan dengan efektif dan efisien. Proses algoritma dalam penelitian ini dimulai dengan generate populasi awal sebanyak jumlah populasi yang ditentukan. Untuk representasi kromosom dilakukan dengan representasi real code yang mewakili kualitas barang. Kemudian dilakukannya evaluasi pada setiap kromosom dalam populasi tersebut. Proses reproduksi dilakukan dengan cara crossover dan mutasi untuk menghasilkan keturunan baru. Individu yang terpilih sebagai solusi adalah individu yang mempunyai nilai fitnes yang tertinggi. Hasil akhir dalam penelitian ini, Algoritma Genetika mampu memberikan solusi yang optimal dalam mengetahui jumlah



barang yang harus dipesan dan dapat mengoptimalkan biaya persediaan yang harus dikeluarkan.

Penelitian sejenis dilakukan oleh Mahmudah, Irhamah, dan Mumpuni (2011) dengan permasalahan pada multi-item dalam optimasi persediaan pada perusahaan PT. UWBM. Penelitian ini bertujuan Mahmudah, Irhamah, dan Mumpuni (2011) untuk menentukan jumlah persediaan yang harus dilakukan oleh PT UWBM, agar dapat meminimalisir biaya total persediaan. Proses awal dalam penelitian ini adalah dengan menggenerate populasi awal sebanyak jumlah populasi yang telah ditentukan. Kemudian reprentasi kromosom dilakukan dengan reprentasi real code yang mewakili 3 item yang digunakan pada PT. UWBM dalam memproduksi, antara lain : tepung terigu, tepung tapioka, dan gula. Setelah itu melakukan proses crossover dan mutasi untuk mendapatkan individu baru. Proses selanjutnya yaitu proses seleksi dengan cara menghitung total biaya persediaan pada masing-masing individu tersebut dann individu yang terpilih sebagai solusi adalah individu dengan komposisi jumlah bahan baku yang menghasilkan total biaya persediaan dengan biaya yang minimal.

Secara garis besar, rangkuman dari penilitian-penilitian tersebut dapat ditampilkan pada Tabel 2.1 berikut.

Tabel 2.1 Kajian Pustaka

No.	Judul	Obyek	Metode	Hasil
1.	Optimasi Distribusi Barang dengan Algoritma Genetika	- Rute yang ditempuh oleh truck pengangkut - pemilihan truck yang tepat	Metode yang digunakan dalam penilitian ini adalah Algoritma Genetika	Algoritma Genetika mampu memberikan suatu solusi dengan total biaya terkecil dan dengan sisa muatan terkecil
2.	Analisis Pengendalian Persediaan Produk dengan Metode EOQ menggunakan Algoritma Genetika untuk Mengifisienkan Biaya Persediaan	Jumlah barang yang dipasok oleh PT. XYZ	Metode yang digunakan dalam penilitian ini adalah metode EOQ dan Algoritma Genetika	Metode EOQ dan Algoritma Genetika dapat memberikan hasil yang optimal dalam meminimalisir suatu biaya persediaan

Tabel 2.2 Kajian Pustaka (Lanjutan)

No.	Judul	Obyek	Metode	Hasil
3	Optimasi Persediaan Multi-Item Fuzzy EOQ dengan Algoritma Genetika	Persediaan bahan baku seperti tepung terigu, tepung tapioka, dan gula	Metode yang digunakan dalam penelitian ini adalah Fuzzy EOQ dan Algoritma Genetika	Metode yang digunakan dapat memberikan hasil yang optimal dalam persediaan Multi-item pada PT UWBM

2.2 Peramalan

Peramalan (*forecasting*) merupakan seni atau ilmu yang digunakan dalam memperkirakan kejadian dimasa depan. Hal tersebut dapat dilakukan dengan melibatkan pengambilan dari data sebelumnya atau data masa lalu dan menenmpatkannya pada masa yang akan datang dengan suatu bentuk model yang sistematis (Render & Heizer, 2001). Menurut Rangkuti (2005) perencanaan kapasitas produksi yang fleksibel adalah dengan perencanaan kapasitas produksi yang sesuai dengan besarnya kebutuhan permintaan. Apabila kapasitas produksi yang direncanakan terlalu besar sehingga melebihi kebutuhan permintaan yang sebenarnya, maka perusahaan tersebut akan mengalami kerugian yang cukup besar. Perencanaan kapasitas produksi yang baik harus sesuai dengan besarnya kebutuhan permintaan (Rangkuti, 2005).

Menurut Sumayang (2003), peramalan merupakan perhitungan yang objektif dengan menggunakan beberapa data masa lalu yang digunakan dalam menentukan sesuatu dimasa yang akan datang. Model pada peramalan secara umum dapat dikemukakan sebagai berikut.

$$Y_t = \text{Pola} + \text{error}$$

Data yang didapat dibedakan menjadi komponen yang dapat diidentifikasi (pola) dan data yang tidak dapat diidentifikasi (error). Pada penggunaan metode peramalan didapatkan untuk mengidentifikasi suatu model peramalan yang sedemikian rupa untuk menhasilkan nilai error yang seminimal mungkin.

2.2.1 Jenis-Jenis Peramalan

Peramalan merupakan sebuah proses untuk melakukan suatu perkiraan tentang sesuatu yang mungkin akan terjadi dimasa yang akan datang berdasarkan informasi atau data dari masa lalu. Jenis peramalan yang digunakan setiap perusahaan memiliki sifat dan tujuan yang berbeda-beda. Menurut Syaffrudin (2014) terdapat dua metode dalam melakukan peramalan, antara lain sebagai berikut.

1. Peramalan Kualitatif

Peramalan kualitatif merupakan peramalan yang didasarkan pada data kualitatif masa lalu. Hasil peramalan dengan menggunakan metode ini sangat bergantung pada setiap pemikiran pada individu yang menyusunnya. Pemikiran individu tersebut biasanya didapatkan dari hasil opini, pengetahuan dan pengalaman pada individu tersebut miliki.

2. Peramalan Kuantitatif

Peramalan kuantitatif merupakan peramalan yang berdasarkan data-data kuantitatif masa lalu yang direpresentasikan secara tegas ke dalam bentuk angka maupun nilai, sehingga dapat memprediksi nilai data pada masa yang akan datang. Penggunaan metode tersebut sangat memperngaruhi hasil akhir peramalan, sebab dengan menggunakan metode yang berbeda akan memperoleh hasil peramalan yang cukup berbeda. Selain itu, hasil yang didapatkan antara data aktual dan data hasil peramalan akan mempengaruhi besarnya suatu penyimpangan.

2.3 Permintaan

Permintaan merupakan suatu kegiatan transaksi dalam perekonomian dengan terdapat 2 aspek yang saling berhubungan, yaitu permintaan (*demand*) dan penawaran (*supply*). Permintaan adalah berbagai jumlah barang yang diminta oleh konsumen dengan tingkatan harga yang berbeda sesuai dengan periode tertentu.

Menurut Virgantari (2011), fungsi permintaan merupakan hubungan jumlah permintaan dan faktor-faktor yang mempengaruhi pada tempat dan waktu tertentu. Pada permintaan memiliki fungsi tersendiri yang dapat diturunkan menjadi 2 cara, yaitu

- Memaksimumkan kepuasan dengan kendala jumlah anggaran dan harga barang.
- Meminimumkan biaya dan memaksimumkan output pada tingkat pengeluaran yang tetap.

Dalam suatu permintaan dipengaruhi oleh beberapa faktor, yaitu harga barang, harga barang subsitusi, selera, jumlah penduduk dan tingkat pendapatan (Danniel, 2004).

2.4 Algoritma Genetika

Algoritma Genetika (Genetic Algorithms, Gas) merupakan tipe Algoritma Evolusi yang paling populer dalam mengatasi permasalahan yang cukup kompleks. Algoritma Genetika berkembang seiring dengan perkembangan teknologi informasi. Algoritma Genetika banyak digunakan dalam bidang fisika, biologi, ekonomi, sosiologi, dan lain-lainnya seiring dengan perkembangan zaman. Dalam bidang industri, Algoritma Genetika dapat digunakan dalam hal perencanaan dan penjadwalan produksi (Mahmudy, Marian & Loung, 2013). Algoritma Genetika juga dapat digunakan dalam hal optimasi penugasan

mengajar bagi dosen (Mahmudy, 2006) dan dapat juga diterapkan dalam kompresi citra (Ciptayani, Mahmudy & Widodo, 2009).

Algoritma Genetika merupakan simulasi dari proses evolusi Darwin dan operasi genetika atas kromosom. Secara garis besar, prosedur awal pada Algoritma Genetika dengan menetapkan suatu set solusi awal potensial yang disebut kromosom, kemudian melakukan perubahan dengan beberapa iterasi untuk mendapatkan solusi yang terbaik. Menurut Michalewics (1996) Algoritma Genetika berbeda dalam beberapa hal, apabila dibandingkan dengan prosedur pencarian dan optimasi yang sederhana. Berikut penjelasan menurut Michalewics (1996) :

- Manipulasi dilakukan terhadap kode himpunan parameter (chromosome), tidak secara langsung terhadap parameternya sendiri.
- Proses pencarian dilakukan dari beberapa titik dalam suatu populasi, tidak dari satu titik saja.
- Proses pencarian menggunakan informasi dari fungsi tujuan.
- Pencarinya menggunakan stochastic operators yang bersifat probabilistik, tidak menggunakan aturan deterministik.

Kromosom yang dibentuk secara random berupa susunan angka binary yang digenerate dan dipilih. Keseluruhan set dari kromosom yang diobservasi mewakili suatu populasi. Kemudian, kromosom-kromosom tersebut akan berevolusi dalam beberapa tahap iterasi yang disebut generasi. Generasi baru (*offsprings*) dibentuk dengan melakukan teknik kawin silang (*crossover*) dan kemudian melakukan teknik mutasi (*mutation*).

Cross over meliputi pemecahan (*splitting*) dua kromosom dan kemudian mengkombinasikan setengah bagian dari masing-masing kromosom dengan pasangan-pasangan lainnya. Sedangkan mutasi meliputi penggantian (*flipping*) satu bagian kromosom dengan satu bagian lain dari kromosom lain yang menjadi pasangannya.

Kromosom-kromosom ini kemudian berevolusi dengan suatu kriteria kesesuaian (*fitness*) yang ditetapkan dan hasil terbaik akan dipilih sementara yang lainnya diabaikan. Selanjutnya, proses dilakukan berulang-ulang sampai dengan suatu kromosom yang mempunyai kesesuaian terbaik (*best fitness*) akan diambil sebagai solusi terbaik dari permasalahan.

2.4.1 Langkah Penyelesaian Algoritma Genetika

Berikut merupakan langkah sederhana dalam menyelesaikan permasalahan dengan menggunakan Algoritma Genetika :

1. [Start] : Generate populasi awal secara random sebanyak n individu.
2. [fitness] : Evaluasi nilai *fitness* $f(x)$ dari setiap individu x di dalam populasi.
3. [New Population] : Membentuk populasi baru dengan melakukan pengulangan pada langkah-langkah Selection, Crossover, dan Mutation.



4. [Selection] : Pilih 2 individu sebagai orang tua (*parent*) yang sesuai dengan *fitness* mereka. Semakin baik nilai *fitness* yang diperoleh, maka semakin besar peluang mereka untuk dapat dipilih.
5. [Crossover] : Melakukan persilangan antara kedua orang tua (*parent*) sesuai dengan probabilitas crossover untuk membentuk keturunan baru. Apabila terjadi persilangan maka keturunan yang dihasilkan akan mewarisi nilai yang sama persis dengan orang tua (*parent*) tersebut.
6. [Mutation] : Melakukan mutasi pada setiap keturunan yang baru sesuai dengan probabilitas setiap gen.
7. [Accepting] : Menempatkan keturunan yang baru sebagai populasi yang baru.
8. [Replace] : Menggunakan populasi yang baru untuk menjalankan Algoritma Genetika.
9. [Test] : Apabila kondisi akhir terpenuhi maka berhenti (*stop*) dan menampilkan solusi yang didapat pada populasi.
10. [Loop] : Kembali pada point 2 (Kusumadewi et al, 2005).

2.4.2 Penerapan Algoritma Genetika

Pada Algoritma Genetika mempunyai beberapa cara untuk mengodekan solusi dari suatu permasalahan dan pengodean ini dipilih sesuai dengan jenis permasalahan. Berikut beberapa pengodean beserta penjelasan mengenai pengodean yang sering digunakan menurut Handayani (2010).

1. Pengodean Biner

Pengodean Biner merupakan pengkodean yang sering digunakan dan paling sederhana. Pengkodean biner pada setiap kromosom direpresentasikan dalam barisan bit 0 atau 1. Berikut contoh Tabel 2.2 mengenai pengkodean biner.

Tabel 2.3 Contoh Pengkodean Biner

Kromosom A	1.345, 4.534, 7.654, 8.789
Kromosom B	ABC, ADC, CBC, BCA
Kromosom C	1, 3, 4, 7, 5
Kromosom D	Foward, Backward, Right, Left



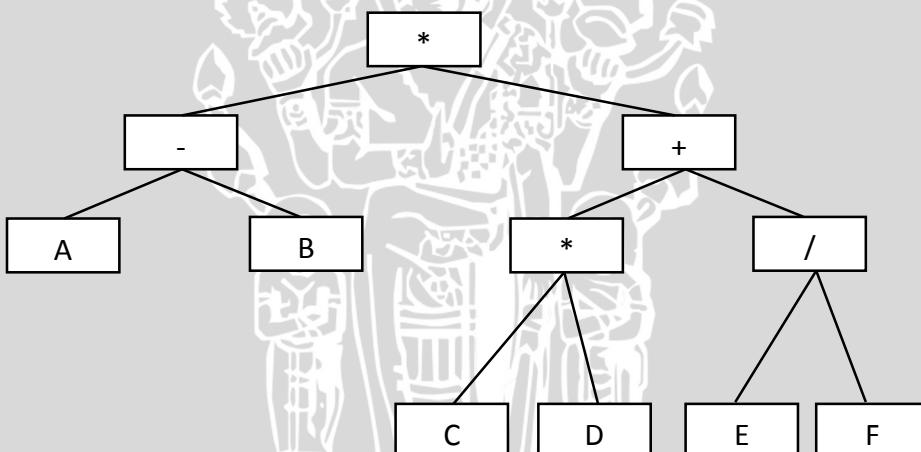
Pengkodean biner memberikan banyak kemungkinan untuk kromosom walaupun dengan jumlah nilai-nilai yang mungkin terjadi dalam suatu gen sedikit (0 atau 1). Pengkodean biner sering tidak sesuai dalam beberapa masalah terkadang harus melakukan pengoreksian setelah operasi crossover dan mutasi.

2. Pengodean Nilai (Value Encoding)

Pada pengkodean nilai, setiap kromosom merupakan string dari suatu nilai dimana nilai yang dikodekan langsung merupakan representasi dari masalah seperti bilangan bulat, desimal, maupun karakter.

3. Pengkodean Pohon

Pengkodean pohon digunakan dalam menyusun sebuah program atau ekspresi di dalam Algoritma Genetika. Pada pengkodean pohon setiap kromosom dinyatakan sebagai sebuah pohon dari setiap objek, seperti fungsi atau perintah dalam bahasa pemrograman. Dalam bahasa pemrograman LISP biasanya sering memanfaatkan pengkodean pohon, sebab didalam program dapat direpresentasikan ke dalam bentuk tersebut, dan dapat dengan mudah di parse menjadi sebuah pohon sehingga crossover dan mutasi dapat dilakukan dengan mudah. Berikut contoh Gambar 2.1 mengenai pengkodean pohon.



Gambar 2.1 Pengkodean Pohon

4. Pengodean Permutasi (Permutation Encoding)

Pengodean permutasi merupakan pengodean yang dilakukan dalam masalah pengurutan data (ordering problem). Pada pengodean ini setiap kromosom merupakan barisan angka yang mempresentasikan angka dalam urutan. Pengodean ini berguna pada permasalahan ordering, bahkan beberapa kolerasi terhadap kromosom harus dilakukan untuk menjaga konsistensi reprentasi kromosom setelah proses crossover dan mutasi. Berikut contoh Tabel 2.3 dalam pengodean permutasi.

Tabel 2.4 Contoh Pengkodean Permutasi

Kromosom A	2 6 7 5 1 3 4 9 8 10
Kromosom B	10 5 4 9 7 1 3 2 6 8

2.4.3 Membangun Generasi Awal

Populasi awal dapat dibangkitkan secara random sehingga didapatkan solusi awal. Populasi itu sendiri terdiri atas sejumlah kromosom yang merepresentasikan solusi yang diinginkan (Desiani et al, dalam Tobing, 2010). Beberapa cara dalam membangkitkan populasi awal, diantaranya adalah

- *Random Generator*
Inti dari random generator adalah melibatkan pembangkitan bilangan random untuk nilai setiap gen sesuai dengan reprentasi kromosom yang digunakan.
- *Pendekatan Tertentu*
Pendekatan tertentu dalam populasi awal adalah dengan cara memasukkan nilai tertentu ke dalam gen dari populasi awal terbentuk.
- *Permutasi Gen*
Cara yang digunakan permutasi gen dalam populasi awal yaitu dengan menggunakan permutasi (pengurutan data).

Populasi awal yang akan diterapkan dalam tugas akhir ini adalah dengan menggunakan bilangan yang dibangkitkan secara random dengan range yang telah ditentukan.

2.4.4 Reprentasi Chromosom

Reprentasi Kromosom merupakan proses pengodean dari penyelesaian asli atau permasalahan. Pengodean kandidat dalam penyelesaian ini disebut dengan kromosom. Pengodean tersebut meliputi penyandian gen dengan satu gen, mewakili satu variabel (Mitchell, dalam Tobing, 2010).

Terdapat beberapa macam kromosom, diantaranya adalah reprentasi biner, reprentasi real code, reprentasi integer, reprentasi matriks, dan reprentasi permutasi (Mahmudy, 2013). Pada penilitian ini menggunakan reprentasi *real code* atau *RealCode Genethic Algoritm*.

2.4.5 Reproduksi

Setelah melakukan inisialisasi langkah selanjutnya yang harus dilakukan adalah melakukan reproduksi. Reproduksi adalah proses untuk menghasilkan keturunan (*offspring*) yang menggunakan induk kromosom secara acak agar setiap kromosom (individu) yang dihasilkan dapat berevolusi menjadi lebih baik lagi (Mahmudy, 2013). Hasil dari setiap individu terbaik dapat beradaptasi sesuai dengan lingkungan serta memiliki peluang yang besar untuk melewati tahap seleksi. Proses reproduksi memiliki dua cara yaitu: proses Crossover dan proses Mutation. Berikut merupakan penjelasan mengenai proses Crossover dan Mutation pada penelitian ini.

2.4.5.1 Proses Crossover

Crossover atau Persilangan merupakan suatu proses untuk menyilangkan dua kromosom sehingga membentuk kromosom anak (*offspring*) yang diharapkan akan lebih baik dari pada induknya. Tujuan dalam tahapan crossover adalah dapat menambah keaneragaman string dalam populasi dengan penyilangan antar-string yang diperoleh dari sebelumnya.

Parameter yang terpenting dalam proses crossover adalah crossover rate yang merupakan nilai perbandingan jumlah kromosom yang diharapkan akan mengalami crossover terhadap jumlah kromosom dalam satu populasi.

Menurut Mahmudy (2013) crossover dilakukan dengan beberapa langkah sederhana, yaitu

1. Dari populasi yang ada, diambil 2 individu atau parent secara acak untuk disilangkan. Metode crossover yang digunakan adalah *crossover one-cut-point*
2. Memilih lokasi pada populasi dalam kromosom tersebut secara random, sehingga masing-masing kromosom dalam populasi tersebut terbagi menjadi dua segmen.
3. Menukar antar segmen kromosom dari induk untuk menghasilkan *offspring*.

Dalam penelitian ini menggunakan metode extends intermediate. Dimana melakukan dengan cara memilih secara acak dua induk (parent). Dengan menggunakan metode extends intermediate crossover akan menghasilkan offspring dari kombinasi nilai dua induk yang terpilih.

2.4.5.2 Mutasi

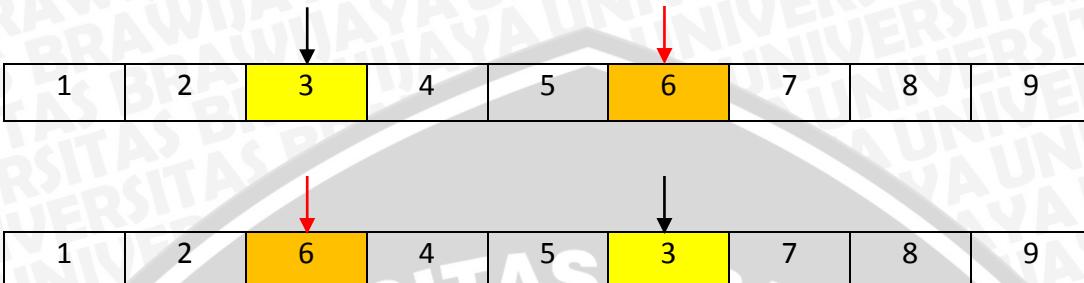
Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Individu yang telah melewati proses seleksi dan crossover akan menghasilkan individu baru (*offspring*) yang kemudian akan dimutasi untuk membantu mempercepat terjadinya perbedaan individu dalam suatu populasi.

Mutasi berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan muncul kembali pada gen yang tidak muncul saat inisialisasi populasi.



Dalam penelitian ini menggunakan metode mutasi exchange point. Dalam exchange mutation akan melakukan dengan cara memilih dua gen secara acak kemudian posisi gen pertama ditukar dengan posisi gen kedua. Berikut contoh Tabel 2.4 proses mutasi dengan metode exchange mutation.

Tabel 2.5 Contoh Exchange Mutation



2.4.6 Fitnes Cost

Fitnes merupakan suatu ukuran kinerja di setiap individu agar tetap bertahan hidup dalam lingkungannya. *Fitness* digunakan dalam proses evaluasi kromosom agar memperoleh kromosom yang diinginkan. Seberapa baik kromosom yang dihasilkan maka dapat diketahui dengan membedakan kualitas kromosom, hal tersebut merupakan fungsi dari fitnes (Tobing, 2010). Dalam penelitian ini, fungsi *fitness* digunakan untuk memaksimalkan keuntungan yang diperoleh dengan meminimalisir biaya pengeluran.

2.4.7 Pengurutan (sorting)

Pengurutan (sorting) digunakan untuk mencari individu yang terbaik pada populasi yang ada (sebagai solusi terbaik sementara) dengan mengurutkan individu yang terdapat pada populasi berdasarkan *fitness cost*. Solusi Algoritma Genetika merupakan solusi sementara, tergantung dari beberapa generasi, populasi akan diulang. Pada tahap ini juga berperan dalam pemilihan solusi yang tepat dimana solusi yang tersedia (Tobing, 2010).

2.4.8 Seleksi

Di dalam proses seleksi memiliki tujuan memberikan kesempatan reproduksi yang lebih besar bagi individu-individu yang terdapat dalam suatu populasi yang paling baik. Langkah awal dalam seleksi ini adalah dengan pencarian nilai *fitness*. Nilai *fitness* tersebut akan digunakan dalam tahap seleksi berikutnya (Kusumadewe, 2003). Seleksi dilakukan untuk dapat mempertahankan individu dari himpunan populasi dan offspring yang dipilih untuk hidup pada generasi berikutnya (Mahmudy, 2013). Ada beberapa metode yang terdapat pada proses seleksi, antara lain :

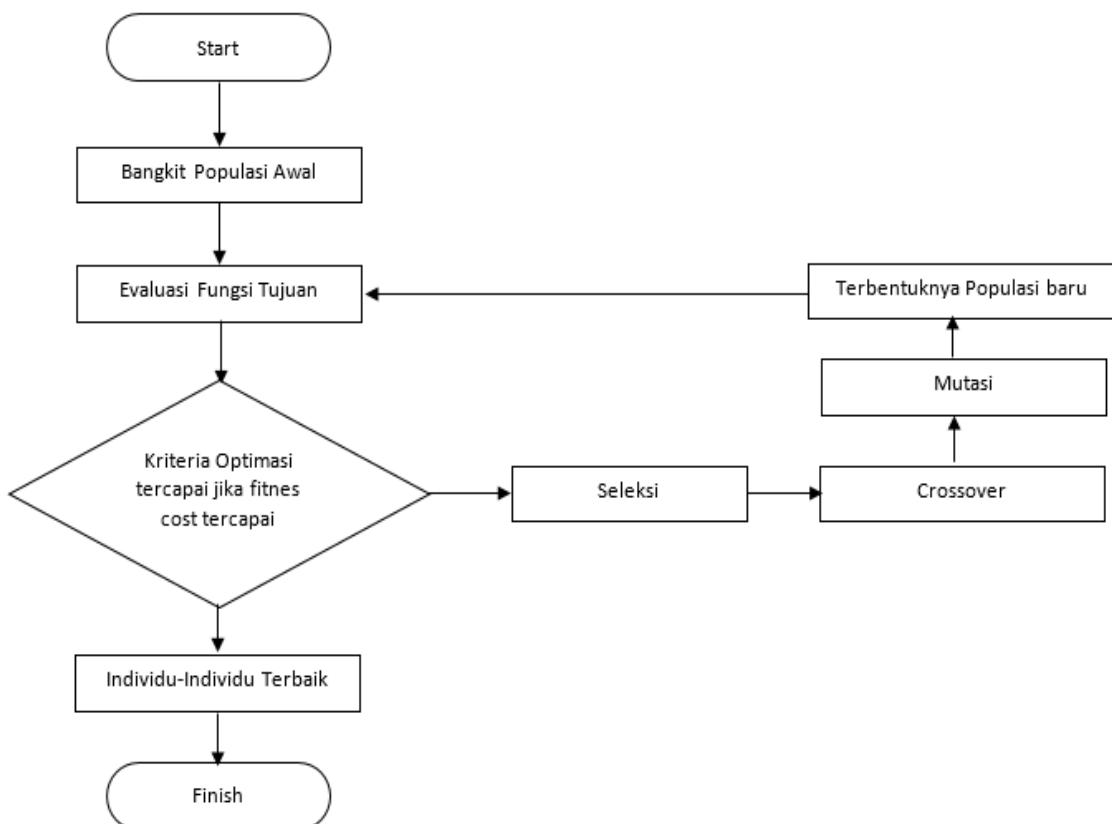
- Seleksi Roda Rolet (*roulette wheel selection*)
- Seleksi ranking (*rank selection*)
- Seleksi Turnament (*tournament selection*)

Pada penilitian ini, metode seleksi yang digunakan adalah elitsm selection. Dimana popSize dengan individu terbaik dipilih dari kumpulan individu di populasi (parent) dan offspring. Metode elitsm selection menjamin individu yang terbaik akan selalu lolos untuk masuk ke dalam generasi berikutnya (Mahmudy, 2013).

2.4.9 Tahap Pengulangan

Pada tahap ini dilakukan setelah proses regenerasi selesai sejumlah generasi yang dikehendaki. Gen dari generasi sebelumnya digantikan posisinya dengan generasi baru. Individu yang diperoleh dari mutasi dan crossover akan dianggap sebagai populasi awal lagi. Pada akhir proses perulangan tersebut diharapkan diperoleh individu terbaik dengan nilai fitnes terbaik.

2.4.10 Diagram Alir Algoritma Genetika



Gambar 2.2 Diagram Alir Algoritma Genetika

Sumber: (Tobing, 2010)

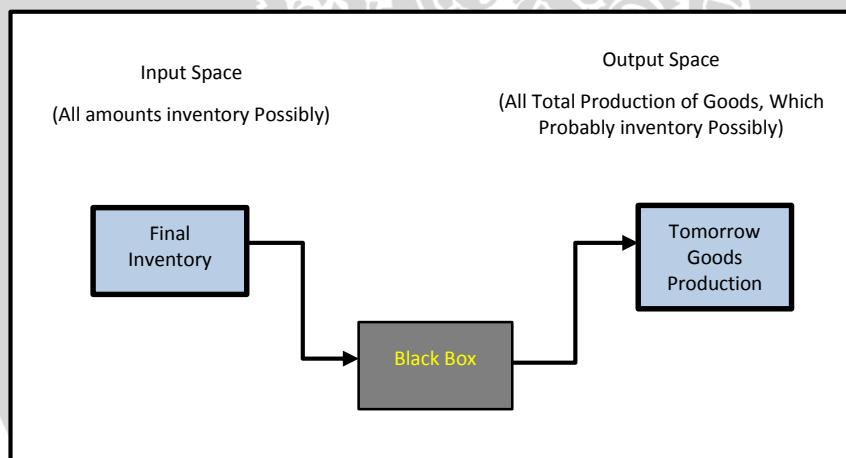
Pada Gambar 2.2, menjelaskan bahwa pada langkah pertama yang dilakukan dalam Algoritma Genetika adalah menggenerate suatu populasi awal secara random sebanyak n individu. Kemudian pada langkah kedua, populasi awal akan dievaluasi dengan rumusan fitnes tertentu agar populasi awal tersebut mendapatkan kromosom-kromosom yang diinginkan. Pada tahap selanjutnya adalah proses seleksi, crossover, mutasi dan terbentuknya populasi baru. Jika

pada kondisi akhir dapat dipenuhi maka berhenti atau selesai dan akan ditampilkan solusi dari populasi.

Metode penilitian dalam skripsi ini dilakukan dengan 7 tahapan, yaitu mencari dan memilih studi literatur, pengambilan data, mengolah data, perancangan sistem, implementasi, pengujian dan analisis, dan pengambilan kesimpulan.

2.5 Fuzzy Inferensi System Tsukamoto

Logika Fuzzy merupakan suatu alternatif untuk menyampaikan suatu informasi sebuah data. Selain itu logika fuzzy merupakan pendeskripsian atau data atau pengetahuan yang masih tidak pasti. Logika fuzzy merupakan sistem komputer yang bersifat soft computing, sebab pada logika fuzzy memberikan alasan akan ketidakpastian. Logika fuzzy pertama kali diperkenalkan oleh Lotfi A. Zadeh pada tahun 1965. Pada teori himpunan fuzzy, peran derajat keanggotaan sebagai penentu adanya unsur dalam satu set sangat penting. Derajat fungsi keanggotaan sebagai karakteristik utama dari fuzzy logic penalaran. Dalam beberapa hal, logika fuzzy digunakan sebagai cara dalam memetakan masalah input yang menuju output yang diharapkan. Berikut merupakan gambaran pemetaan input-output dalam bentuk grafik seperti pada Gambar 2.1 berikut ini.



Gambar 2.3 Mapping input dan output

Sumber: (Bon & Utami, 2014)

Metode Fuzzy Tsukamoto pertama kali diperkenalkan oleh Tsukamoto pada tahun 1979. Fuzzy Tsukamoto merupakan salah satu metode pengambilan keputusan. Metode ini berlaku untuk setiap penggunaan dengan penelaran monoton aturan, dengan maksud untuk menggunakan sistem dengan hanya satu aturan. Implikasi yang didapatkan dari setiap aturan dengan bentuk implikasi "sebab-akibat" atau dengan implikasi berupa "If-then" dimana dari setiap implikasi tersebut dapat berhubungan. Pada setiap aturan yang digunakan akan diwakili dengan asosiasi fuzzy, dengan derajat fungsi keanggotaan yang monoton. Kemudian dalam menentukan hasil dari sebuah Cirsp Solution, menggunakan rumus pernyataan (deffuzyifikasi) yang disebut juga dengan



berpusat metode rata-rata. Menurut Maryaningsih, dkk (2013) menjelaskan bahwa metode Tsukamoto terdapat 4 tahapan. Berikut merupakan tahapan-tahapan di dalam metode Tsukamoto.

1. Fuzzyifikasi
2. Pembentukan basis pengetahuan Fuzzy atau Rule (IF-THEN atau SEBAB-AKIBAT).
3. Menggunakan fungsi implikasi MIN untuk mendapatkan nilai α -predikat pada setiap rule. Kemudian nilai α -predikat akan digunakan untuk menghitung keluaran hasil inferensi secara tegas (Crisp) masing-masing rule.
4. Defuzzyifikasi

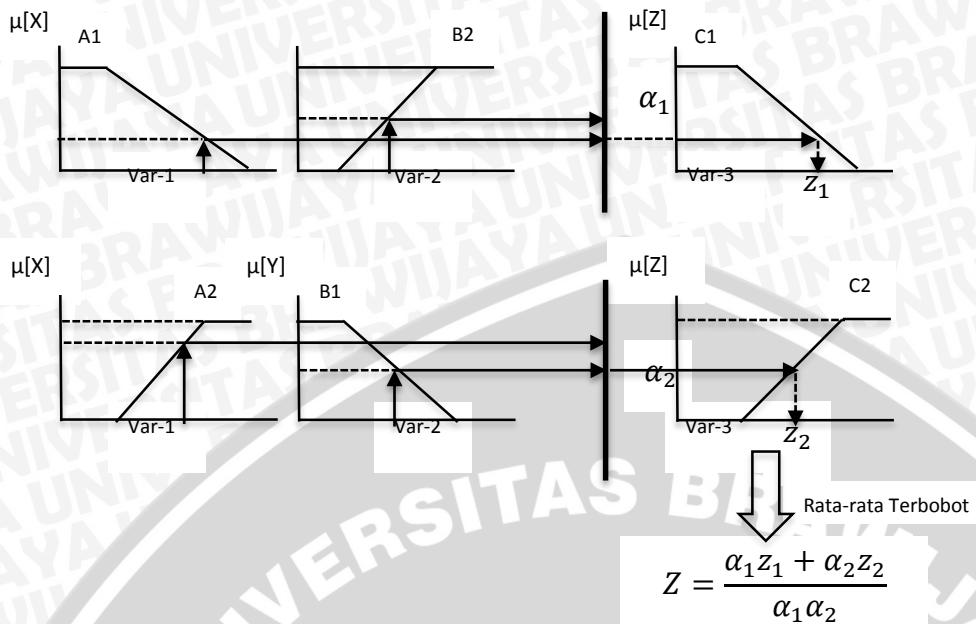
Dengan menggunakan metode rata-rata (average) formulasi tersebut dapat menetukan nilai output cr yang akan menjadi hasil keputusan (z), kemudian dengan mengubah input (dalam bentuk aturan fuzzy) menjadi beberapa fuzzy set di dalam domain. Dengan keterangan variabel nialai Z maka variabel output, αi merupakan nilai α predikat dan Zi merupakan nilai variabel output. Nilai keanggotaan (α) akan dicari pada setiap rule yang digunakan dan apabila aturan yang digunakan lebih dari satu, maka akan dilakukan proses agregasi pada semua aturan. Kemudian pada nilai agregasi tersebut akan dilakukan defuzzy, dengan tujuan mendapatkan nilai crisp sebagai nilai output (Soraya, Siti, 2014). Berikut ini merupakan persamaan dalam proses mendapatkan nilai crisp.

$$Z = \frac{\sum_{i=1}^n aizi}{\sum_{i=1}^n ai}$$

Pada sistem inferensi fuzzy dengan menggunakan metode Tsukamoto didasarkan pada konsep penalaran yang monoton. Penggunaan konsep penalaran monoton tersebut pada metode tsukamoto pada setiap konsekuensi pada aturan berbentuk jika harus direpresentasikan dengan suatu himpunan fuzzy dengan fungsi keanggotaan yang monoton. Keluara hasil inferensi dari setiap aturan diberikan secara tegas (crisp) berdasarkan α -predikat (Gunawan, Rouf, 2013).

Pada metode tsukamoto, setiap konsekuensi pada aturan yang berbentuk IF-Then direpresentasikan dengan suatu himpunan fuzzy dengan fungsi keanggotaan yang monoton. Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot. (Farouq & Miftahus, 2014). Proses pengambilan keputusan merupakan salah satu proses yang penting untuk memecahkan permasalahan yang sedang dihadapi dengan memilih satu dari beberapa alternatif penyelesaian yang dimungkinkan. Dengan menggunakan metode logika fuzzy tsukamoto maka akan lebih mudah dalam pembobotan dan pengambilan keputusan (Kumar & Jain, 2012).





Gambar 2.4 Sistem Inferensi Fuzzy Metode Tsukamoto

Pada gambar 2.4 merupakan hasil aturan fuzzy yang telah direpresentasikan untuk mendapatkan nilai crisp Z dengan menggunakan aturan IF-THEN. Dalam metode tsukamoto menggunakan metode deffuzifikasi (penegasan) dengan rata-rata terpusat (Center Average Deffuzifieer). Pada penelitian ini didapatkan beberapa aturan atau rule dalam metode Fuzzy Tsukamoto, pada Tabel 2.5 merupakan aturan atau rule yang digunakan pada penelitian ini dengan jumlah aturan sebanyak 30 rule atau aturan.

Tabel 2.6 Aturan Fuzzy Tsukamoto

Rule	IF			THEN
	T3 (3 minggu sebelumnya)	T2 (2 minggu sebelumnya)	T1 (1 minggu sebelumnya)	T (minggu selanjutnya)
1	Tinggi	Rendah	Sedang	Tinggi
2	Rendah	Sedang	Tinggi	Rendah
3	Sedang	Tinggi	Rendah	Sedang
4	Tinggi	Tinggi	Sedang	Tinggi
5	Tinggi	Sedang	Tinggi	Tinggi
6	Sedang	Tinggi	Tinggi	Tinggi
7	Sedang	Sedang	Tinggi	Sedang
8	Sedang	Tinggi	Sedang	Sedang
9	Tinggi	Sedang	Sedang	Sedang
10	Rendah	Rendah	Tinggi	Rendah
11	Rendah	Tinggi	Rendah	Rendah
12	Tinggi	Rendah	Rendah	Rendah
13	Rendah	Rendah	Sedang	Rendah
14	Rendah	Sedang	Rendah	Rendah
15	Sedang	Rendah	Rendah	Rendah

Tabel 2.7 Aturan Fuzzy Tsukamoto (Lanjutan)

Rule	IF			THEN
	T3 (3 minggu sebelumnya)	T2 (2 minggu sebelumnya)	T1 (1 minggu sebelumnya)	T (minggu selanjutnya)
16	Sedang	Sedang	Rendah	Sedang
17	Sedang	Rendah	Sedang	Sedang
18	Rendah	Sedang	Sedang	Sedang
19	Tinggi	Tinggi	Rendah	Tinggi
20	Tinggi	Rendah	Tinggi	Tinggi
21	Rendah	Tinggi	Tinggi	Tinggi
22	Tinggi	Tinggi	Tinggi	Tinggi
23	Rendah	Rendah	Rendah	Rendah
24	Sedang	Sedang	Sedang	Sedang
25	Sedang	Rendah	Tinggi	Tinggi
26	Rendah	Tinggi	Sedang	Sedang
27	Tinggi	Sedang	Rendah	Rendah
28	Rendah	Tinggi	Sedang	Tinggi
29	Tinggi	Sedang	Rendah	Sedang
30	Sedang	Rendah	Tinggi	Rendah

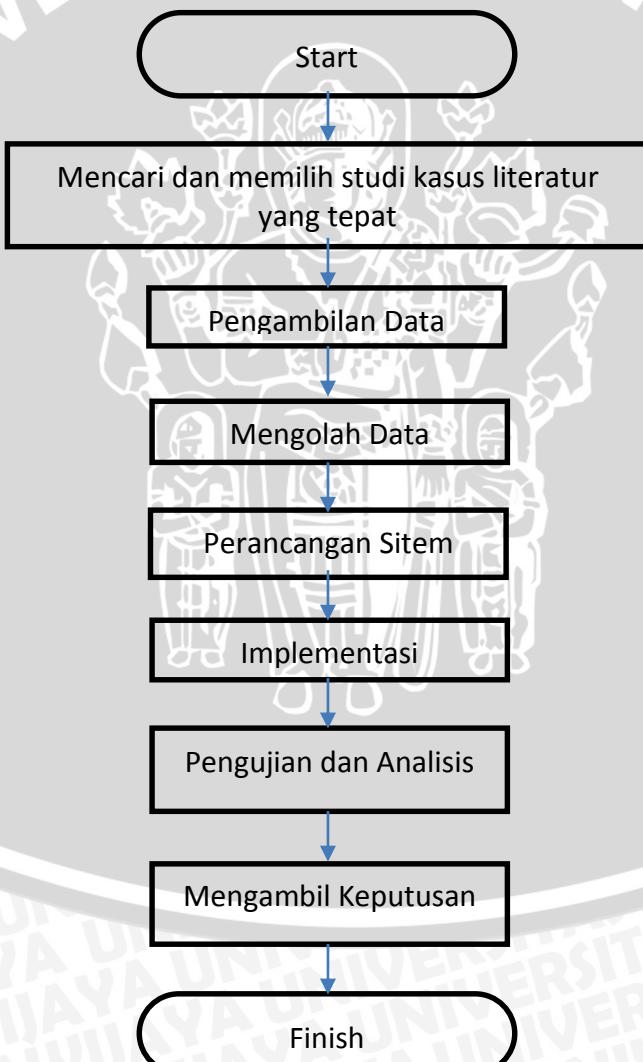


BAB 3 METODOLOGI

Skripsi dengan judul “Optimasi Fuzzy Tsukamoto dengan Algoritma Genetika Studi Kasus Peramalan Permintaan Barang Semen” ini dapat digolongkan sebagai penelitian yang implementatif dengan pendekatan perancangan (design). Penelitian ini akan menghasilkan suatu purwarupa (prototype) berupa perangkat lunak (software) yang bisa digunakan dalam hal mengoptimasikan suatu permintaan barang.

3.1 Tahapan Penelitian

Pada tahap ini akan membahas tentang tahapan penelitian yang dilakukan untuk membuat sistem optimasi persediaan barang dalam produksi Semen dengan menggunakan Algoritma Genetika. Metode penelitian ini menggunakan diagram waterfall, seperti digambarkan pada Gambar 3.1



Gambar 3.1 Diagram Waterfall

Metode penitian dalam skripsi ini dilakukan dengan 7 tahapan, yaitu mencari dan memilih studi literatur, pengambilan data, mengolah data, perancangan sistem, implementasi, pengujian dan analisis, dan pengambilan kesimpulan.

3.1.1 Studi Literatur

Studi literatur yang digunakan dalam skripsi ini berasal dari buku, artikel ilmiah, paper dan informasi yang tersedia pada internet baik berupa dokumen maupun artikel. Metode yang digunakan untuk memperoleh suatu dasar teori sebagai sumber acuan untuk penulisan skripsi dan pengembangan suatu aplikasi. Studi literatur yang dibutuhkan terkait dengan pembuatan skripsi ini, meliputi :

- a. Manajemen Permintaan Barang Semen.
- b. Algoritma Genetika.
- c. Fuzzy Tsukamoto

3.1.2 Pengambilan Data

Pada tahap kedua pada penitian ini adalah dengan melakukan suatu pengambilan data. Dalam tahap ini digunakan agar mendapatkan sampel data sebagai acuan untuk menyelesaikan masalah pada kasus prediksi permintaan barang semen. Dalam pengambilan data terdapat beberapa langkah yang harus dilakukan, antara lain :

1. Pada pengamatan langsung terdapat data jumlah permintaan barang semen di setiap minggunya.
2. Menghitung total jumlah permintaan barang yang dihasilkan pada perusahaan Koperasi Warga Semen Gresik (KWSG) dengan mengakumulasikan jumlah permintaan dan jumlah persediaan pada setiap minggunya.

Data sampel yang didapatkan berupa data primer yaitu data permintaan barang semen yang dilakukan oleh Perusahaan Koperasi Semen Gresik (KWSG) setiap minggunya. Data permintaan barang semen didapatkan sebanyak 25 minggu dimulai dari bulan Januari sampai dengan bulan Juli 2015.

3.1.3 Mengolah Data

Pada tahap ketiga ini digunakan untuk menganalisis data yang diperoleh dan diolah menjadi data yang menghasilkan urutan penjualan dan persediaan yang ada dari setiap minggunya. Dalam pengolahan data dilakukan beberapa tahap dalam mendapatkan hasil yang optimal, tahapan yang dilakukan antara lain :

- 1) Membuat populasi awal secara radom kemudian melakukan reproduksi dengan crossover dan permutasi pada populasi tersebut.
- 2) Seleksi dengan metode elitism



- 3) Menentukan rumusan fitnes agar mendapatkan solusi akhir yang optimal
- 4) Iterasi dilakukan untuk generasi berikutnya

3.1.4 Perancangan Sistem

Pada tahap keempat ini merupakan tahap dimana penulis memulai merancang suatu sistem yang sesuai dengan tujuan dari penelitian yang telah dibuat. Data sample dan teori-teori yang didapatkan dari pustaka digabungkan agar dapat diimplementasikan dalam merancang dan mengembangkan sistem optimasi persediaan barang dalam produksi semen.

3.1.5 Implementasi

Pada tahap implementasi ini dilakukan dengan menggunakan bahasa pemrograman berorientasi objek. Java Netbeans merupakan software yang digunakan dalam bahasa pemrograman java, yang digunakan untuk jumlah popsize dan iterasi yang diinginkan. Output yang dihasilkan pada program yang dibuat adalah data solusi dari setiap iterasi sesuai dengan popsize dan jumlah iterasi yang diinginkan user dan dengan kromosom yang mempunyai nilai fitnes yang lebih besar. Apabila nilai fitnes menunjukkan nilai yang lebih besar maka solusi yang didapatkan terbaik.

3.1.6 Pengujian dan Analisis

Pada tahap ini dilakukan pengujian berdasarkan implementasi yang telah dilakukan melalui perhitungan fitnes sistem optimasi persediaan barang bila dibandingkan dengan sistem manual. Pada tahap ini juga dilakukan analisa dan uji coba dengan data permintaan yang telah didapatkan, serta menguji metode yang digunakan apakah dapat memberikan hasil yang optimal. Hasil fitnes dari uji coba menggunakan Algoritma Genetika akan dilakukan evaluasi, untuk mendapatkan persediaan barang yang optimal.

Jika direpresentasikan pada penelitian yang sesungguhnya, maka tahapannya adalah sebagai berikut :

1. Menganalisis permasalahan dan kebutuhan sistem optimasi persediaan barang dalam produksi semen dengan Algoritma Genetika dengan tujuan dapat meminimalisirkan biaya persediaan.
2. Mengumpulkan data dengan cara primer, yaitu dengan mendapatkan data secara langsung melalui bagian kepala SDM di Perusahaan .
3. Merancang sistem sesuai dengan kebutuhan yang telah ditentukan.
4. Membangun sistem sesuai dengan kebutuhan yang telah dirancang sebelumnya.
5. Melakukan uji coba terhadap sistem yang dibuat untuk mengetahui solusi maksimal dalam permasalahan peramalan permintaan barang Semen.
6. Melakukan evaluasi terhadap hasil uji coba yang telah diperoleh.



3.1.7 Pengambilan Kesimpulan

Pada tahap ini merupakan tahapan pengambilan kesimpulan terhadap semua tahapan penilitian yang selesai dilakukan. Kesimpulan didapatkan dari hasil analisis dan perhitungan yang telah dilakukan terhadap optimasi persediaan barang dalam produksi semen dengan menggunakan Algoritma Genetika. Pada tahap terakhir dalam penelitian ini adalah saran, saran tersebut dimaksudkan agar dapat memperoleh masukan untuk memperbaiki kesalahan-kesalahan yang terjadi sebelumnya dan menyempurnakan hasil penulisan serta memberikan pertimbangan atau pengembangan dalam analisis berikutnya.

3.2 Teknik Pengumpulan Data

Asal data yang didapatkan dalam penelitian ini dari data biaya pemasukan dan data biaya pengeluaran serta biaya-biaya tambahan seperti biaya pajak maupun biaya pengiriman barang pada gudang Koperasi Semen Gresik yang berada di daerah Jawa Barat. Data yang didapatkan setiap minggunya dari bulan Januari sampai dengan bulan Juni.

3.3 Algoritma yang Digunakan

Pada penelitian ini menggunakan 2 Algoritma, yaitu Algoritma Genetika dan Algoritma FIS Tsukamoto seperti yang telah dijelaskan pada Bab 2. Algoritma tersebut digunakan sebab mampu menyelesaikan suatu permasalahan yang tidak pasti, sehingga penyelesaian masalah tersebut dapat menggunakan Algoritma Logika fuzzy. Kemudian dalam menentukan batasan-batasan dalam penyelesaian FIS Tsukamoto, maka Algoritma Genetika mampu memberikan solusi yang optimal.

3.4 Kebutuhan Sistem

Kebutuhan sistem yang dilakukan dalam mengidentifikasi semua kebutuhan (requirements) sistem agar dapat berjalan dengan optimal. Aplikasi yang dibuat pada penelitian ini adalah aplikasi aplikasi yang menerapkan Algoritma Genetika untuk mengoptimasi persediaan barang dalam produksi Semen. Implementasi akan dilakukan dengan menggunakan bahasa pemrograman java dengan tipe Netbeans 8.0.2 dan diimplementasikan pada Laptop dengan ukuran prosessor 64 bit dengan kapasitas RAM 2 gb.

3.5 Pengujian Algoritma Genetika

Pada tahap ini dilakukan untuk merancang pengujian Algoritma Genetika. Pengujian Algoritma Genetika ini digunakan agar dapat mengetahui tingkat akurasi dalam hal optimasi persediaan barang pada gudang. Berikut merupakan tahapan pengujian dalam Algoritma Genetika :

1. Pengujian dalam mencari kombinasi *crossover rate* dan *mutation rate* yang terbaik.



2. Pengujian dalam menentukan ukuran populasi awal.
3. Pengujian dalam menentukan ukuran generasi.

3.5.1 Pengujian Kombinasi *Crossover Rate* dan *Mutation Rate*

Pengujian ini digunakan agar dapat mengetahui kombinasi *crossover rate* dan *mutation rate* yang terbaik dalam menghasilkan solusi yang optimal. Pada uji coba kombinasi *crossover rate* dan *mutation rate* bertujuan untuk mengetahui kombinasi cr dan mr yang optimal untuk menghasilkan nilai fitness yang terbaik. Nilai yang digunakan pada crossover rate dan mutation rate adalah antara 0 sampai dengan 1. Sedangkan untuk banyaknya generasi yang digunakan adalah 20. Setiap generasi akan dilakukan percobaan sebanyak 10 kali dan dihitung nilai rata-rata fitness yang didapatkan. Berikut merupakan tabel uji coba kombinasi crossover rate dan mutation rate yang dapat dilihat pada Tabel 3.1.

Tabel 3.1 Perancangan Uji Coba Kombinasi Crossover Rate dan Mutation Rate

Kombinasi		Nilai Fitness										Rata-rata fitness
		Percobaan Kombinasi Cr dan Mr ke-										
Cr	Mr	1	2	3	4	5	6	7	8	9	10	
1	0											
0,9	0,1											
0,8	0,2											
0,7	0,3											
0,6	0,4											
0,5	0,5											
0,4	0,6											
0,3	0,7											
0,2	0,8											
0,1	0,9											
0	1											

3.5.2 Pengujian Ukuran Populasi

Pada uji coba ini, dilakukan agar dapat mengetahui berapa ukuran populasi yang optiamal untuk menghasilkan rata-rata nilai *fitness* tertinggi dalam optimasi persediaan barang. Pada pengujian ini menggunakan ukuran populasi dengan kelipatan 10 mulai 10 sampai dengan 110 dan untuk rata-rata nilai *fitness* akan dilakukan 10 kali dari setiap ukuran populasi. Tabel perancangan uji coba ukuran populasi dapat dilihat pada Tabel 3.2.

Tabel 3.2 Perancangan Uji Coba Ukuran Populasi

No	Banyak Populasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>	
		Percobaan Populasi ke-											
		1	2	3	4	5	6	7	8	9	10		
1	10												
2	20												
3	30												
4	40												
5	50												
6	60												
7	70												
8	80												
9	90												
10	100												
11	110												

3.5.3 Pengujian Ukuran Generasi

Pada pengujian ini dilakukan untuk mengetahui banyaknya jumlah generasi yang optimal untuk menghasilkan nilai rata-rata *fitness* tertinggi. Banyak generasi yang digunakan adalah kelipatan 50 dimulai dari 50 sampai dengan 600 generasi.setiap generasi akan dilakukan pengujian sebanyak 10 kali percobaan dan dihitung nilai rata-rata *fitness*nya.

Tabel 3.3 Perancangan Uji Coba Ukuran Generasi

No	Banyak Generasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>	
		Percobaan Populasi ke-											
		1	2	3	4	5	6	7	8	9	10		
1	50												
2	100												
3	150												
4	200												
5	250												
6	300												
7	350												
8	400												
9	450												



Tabel 3.4 Perancangan Uji Coba Ukuran Generasi (Lanjutan)

No	Banyak Generasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>
		Percobaan Populasi ke-										
10	500											
11	550											
12	600											

3.5.4 Pengujian Jumlah Data

Uji coba jumlah data dilakukan untuk mendapatkan jumlah data yang akan menghasilkan solusi optimal. Pada uji coba ini jumlah data yang digunakan adalah 15-45. Perancangan tabel uji coba jumlah data akan ditampilkan pada Tabel 4.19. Parameter yang digunakan pada uji coba ini adalah :

Pada pengujian jumlah data dilakukan agar mendapatkan nilai *fitness* yang optimal dari jumlah data yang digunakan. Pada pengujian jumlah data ini, data yang digunakan antara 5-25. Berikut pada tabel 3.4 merupakan rancangan tabel uji coba jumlah data pada studi kasus peramalan permintaan barang.

Tabel 3.5 Perancangan Uji Coba Jumlah Data

Jumlah Data	Nilai <i>fitness</i> percobaan ke-										Rata-rata Fitness
	1	2	3	4	5	6	7	8	9	10	
5											
10											
15											
20											
25											

BAB 4 PERANCANGAN

4.1 Formulasi Permasalahan

Pada penelitian ini akan menjelaskan tentang permasalahan yang akan diselesaikan. Permasalahan pada penelitian ini adalah peramalan permintaan barang semen. Tujuan dalam penelitian ini adalah dapat mengoptimalkan suatu nilai error pada peramalan atau prediksi dalam menentukan jumlah barang untuk di waktu yang akan datang. Data gudang yang digunakan dalam penelitian ini merupakan Gudang milik Koperasi Warga Semen Gresik yang berada pada Provinsi Jawa Barat di kota Bandung, dalam wilayah tersebut permintaan barang Semen Gresik sangat sulit untuk dipasarkan dengan skala cukup tinggi. Akibatnya barang produksi Semen Gresik mengalami penumpukan pada gudang dan pengadaraan barang yang tidak teratur.

Pada penelitian ini, data yang diambil merupakan data permintaan barang produksi semen pada gudang selama 25 minggu terakhir. Pada data yang diperlukan terdapat kriteria minggu dan data kriteria permintaan. Berikut pada Tabel 4.1 merupakan Data Permintaan barang produksi semen yang diperoleh pada Gudang perusahaan Koperasi Warga Semen Gresik (KWSG).

Tabel 4.1 Data Permintaan Barang

Minggu	Permintaan
1	13543
2	11957
3	15016
4	9990
5	8785
6	8389
7	11776
8	12297
9	11067
10	11239
11	9394
12	9275
13	11772
14	10135
15	11535
16	8202
17	10067
18	5725
19	8766
20	11468
21	9910

Tabel 4.2 Data Permintaan Barang (Lanjutan)

Minggu	Permintaan
22	10088
23	5078
24	8300
25	5381

4.1.1 Contoh Permasalahan

Pada penelitian ini digunakan untuk dapat mengoptimasi dalam memprediksi pengadaan suatu barang produksi yang sesuai dengan data permintaan sebelumnya. Dalam proses memprediksi suatu permintaan barang produksi, maka dibutuhkan metode perhitungan yang sesuai dengan batasan-batasan yang dimiliki pada data permintaan.

Metode Fuzzy Tsukamoto merupakan metode yang sesuai dalam memprediksi jumlah permintaan selanjutnya. Dalam perhitungan Fuzzy Tsukamoto memiliki 4 Kriteria, antara lain T3, T2, T1, dan T. T3 merupakan kriteria data permintaan barang 3 minggu sebelumnya, kemudian pada T2 merupakan kriteria data permintaan barang 2 minggu sebelumnya, lalu pada T1 merupakan kriteria data permintaan barang 1 minggu sebelumnya, dan pada T merupakan kriteria data permintaan barang minggu selanjutnya. Pada subbab ini akan diberikan contoh untuk menghitung jumlah minggu selanjutnya, yaitu menghitung permintaan barang pada minggu ke 10, dapat dilihat pada tabel 4.2.

Tabel 4.3 Contoh Permasalahan

Minggu	Permintaan
7	11776
8	12297
9	11067
10	

Berikut merupakan tahapan-tahapan perhitungan dengan menggunakan metode Fuzzy Tsukamoto.

1. Fuzzyifikasi

- a. Fungsi Keanggotaan permintaan barang 3 minggu sebelumnya (T3)

- Rendah

$$\mu_{Rendah}[x] = \begin{cases} 1, & x \leq 8000 \\ \frac{8500 - x}{500}, & 8000 \leq x \leq 8500 \\ 0, & x \geq 8500 \end{cases}$$

- Sedang

$$\mu_{Sedang}[x] \begin{cases} 0, & x \leq 8000 \\ \frac{x - 8000}{500}, & 8000 \leq x \leq 8500 \\ 1, & 8500 \leq x \leq 11000 \\ \frac{11500 - x}{500}, & 11000 \leq x \leq 11500 \\ 0, & x \geq 11500 \end{cases}$$

- Tinggi

$$\mu_{Tinggi}[x] \begin{cases} 0, & x \leq 11000 \\ \frac{x - 11000}{500}, & 11000 < x < 11500 \\ 1, & x \geq 11500 \end{cases}$$

- b. Fungsi Keanggotaan permintaan barang 2 minggu sebelumnya (T2)

- Rendah

$$\mu_{Rendah}[x] \begin{cases} 1, & x \leq 8000 \\ \frac{8500 - x}{500}, & 8000 \leq x \leq 8500 \\ 0, & x \geq 8500 \end{cases}$$

- Sedang

$$\mu_{Sedang}[x] \begin{cases} 0, & x \leq 8000 \\ \frac{x - 8000}{500}, & 8000 \leq x \leq 8500 \\ 1, & 8500 \leq x \leq 11000 \\ \frac{11500 - x}{500}, & 11000 \leq x \leq 11500 \\ 0, & x \geq 11500 \end{cases}$$

- Tinggi

$$\mu_{Tinggi}[x] \begin{cases} 0, & x \leq 11000 \\ \frac{x - 11000}{500}, & 11000 < x < 11500 \\ 1, & x \geq 11500 \end{cases}$$

- c. Fungsi Keanggotaan permintaan barang 1 minggu sebelumnya (T1)

- Rendah

$$\mu_{Rendah}[x] \begin{cases} 1, & x \leq 8000 \\ \frac{8500 - x}{500}, & 8000 \leq x \leq 8500 \\ 0, & x \geq 8500 \end{cases}$$

- Sedang

$$\mu_{Sedang}[x] = \begin{cases} 0, & x \leq 8000 \\ \frac{x - 8000}{500}, & 8000 \leq x \leq 8500 \\ 1, & 8500 \leq x \leq 11000 \\ \frac{11500 - x}{500}, & 11000 \leq x \leq 11500 \\ 0, & x \geq 11500 \end{cases}$$

- Tinggi

$$\mu_{Tinggi}[x] = \begin{cases} 0, & x \leq 11000 \\ \frac{x - 11000}{500}, & 11000 < x < 11500 \\ 1, & x \geq 11500 \end{cases}$$

Setelah mendapatkan fungsi keanggotaan yang dibangun, maka selanjutnya adalah mengkonversikan dengan nilai kriteria yg telah ditentukan. Sehingga didapatkan nilai fuzzyifikasi seperti pada Tabel 4.3

Tabel 4.4 Fuzzyifikasi nilai kriteria

No.	Jenis Kriteria	Rendah	Sedang	Tinggi
1.	T3	0	0	1
2.	T2	0	0	1
3.	T1	0	0,866	0,134

2. Membuat Basis Aturan

Dalam permasalahan ini, hasil yang didapatkan dari sistem merupakan nilai tinggi, sedang, ataupun rendah. Nilai masukan yang ada sebesar 4 kriteria dan fungsi keanggotaan setiap kriteria memiliki 3 yaitu rendah, sedang, dan tinggi. Untuk basis aturan yang didapatkan sebanyak 30 aturan, yang dapat dilihat pada Tabel 4.4.

Tabel 4.5 Basis Aturan

No	T3	T2	T1	T
1	T	R	S	T
2	R	S	T	R
3	S	T	R	S
4	T	T	S	T
5	T	S	T	T
6	S	T	T	T
7	S	S	T	S
8	S	T	S	S
9	T	S	S	S
10	R	R	T	R

Tabel 4.6 Basis Aturan (Lanjutan)

No	T3	T2	T1	T
10	R	R	T	R
11	R	T	R	R
12	T	R	R	R
13	R	R	S	R
14	R	S	R	R
15	S	R	R	R
16	S	S	R	S
17	S	R	S	S
18	R	S	S	S
19	T	T	R	T
20	T	R	T	T
21	R	T	T	T
22	T	T	T	T
23	R	R	R	R
24	S	S	S	S
25	S	R	T	T
26	R	T	S	S
27	T	S	R	R
28	R	T	S	T
29	T	S	R	S
30	S	R	T	R

3. Defuzzyfikasi

Pada permasalahan ini, hasil dari fungsi keanggotaan yang didapatkan akan diletakkan berdasarkan setiap aturan yang ada pada tabel aturan, sehingga terbentuklah pada Tabel 4.5 sebagai berikut.

Tabel 4.7 Konversi Fuzzifikasi pada Basis Aturan

T3	T2	T1	T	α -predikat	z	α -predikat \times z
1	0	0,866	T	0	11000	0
0	0	0,134	R	0	8500	0
0	1	0	S	0	8000	0
1	1	0,866	T	0,866	11433	9900,978
1	0	0,134	T	0	11000	0
0	1	0,134	T	0	11000	0
0	0	0,134	S	0	8000	0
0	1	0,866	S	0	8000	0
1	0	0,866	S	0	8000	0
0	0	0,134	R	0	8500	0
0	1	0	R	0	8500	0
1	0	0	R	0	8500	0

T3	T2	T1	T	α -predikat	z	α -predikat \times z
0	0	0,866	R	0	8500	0
0	0	0	R	0	8500	0
0	0	0	R	0	8500	0
0	0	0	S	0	8000	0
0	0	0,866	S	0	8000	0
0	0	0,866	S	0	8000	0
1	1	0	T	0	11000	0
1	0	0,134	T	0	11000	0
0	1	0,134	T	0	11000	0
1	1	0,134	T	0,134	11067	1482,978
0	0	0	R	0	8500	0
0	0	0,866	S	0	8000	0
0	0	0,134	T	0	11000	0
0	1	0,866	S	0	8000	0
1	0	0	R	0	8500	0
0	1	0,866	T	0	11000	0
1	0	0	S	0	8000	0
0	0	0,134	R	0	8500	0
Total				1	11383,956	

Hasil Z totalnya adalah $(\alpha\text{-predikat} \times z) / \alpha\text{-predikat}$, maka hasil yang didapatkan adalah $11383,956/1 = 11383,956$. Maka untuk hasil permintaan untuk periode selanjutnya pada minggu ke-25 didapatkan sebesar 11383,956.

Dari hasil peramalan dengan perhitungan Fuzzy Tsukamoto, maka dapat dilihat pada tabel 4.6 yang merupakan nilai hasil prediksi yang didapat dengan menggunakan metode Fuzzy Tsukamoto dan pada tabel 4.7 merupakan data asli pada data permintaan barang.

Tabel 4.8 Hasil Prediksi Permintaan

Minggu	Permintaan
7	11776
8	12297
9	11067
10	11384,0

Tabel 4.9 Data Asli Permintaan

Minggu	Permintaan
7	11776
8	12297
9	11067
10	11239

Kemudian untuk mengetahui nilai error dari hasil prediksi dengan menggunakan metode Fuzzy Tsukamoto maka dapat diperhitungkan dengan menggunakan perhitungan RMSE. Pada penelitian ini, pada bagian perhitungan RMSE data yang digunakan sebanyak 10 minggu. Berikut merupakan tabel 4.8 merupakan tabel Error Fuzzy Tsukamoto, yang berfungsi untuk mengetahui jumlah selisih yang didapat pada hasil prediksi Tsukamoto dengan data asli.

Tabel 4.10 Error Fuzzy Tsukamoto

Minggu	Data Permintaan	Hasil Prediksi Tsukamoto	Error (Data Permintaan-Hasil Prediksi)	Error (Data Permintaan-Hasil Prediksi) ²
1	13543	N/A	N/A	N/A
2	11957	N/A	N/A	N/A
3	15016	N/A	N/A	N/A
4	9990	11500	-1510	2280100
5	8785	11500	-2715	7371225
6	8389	8500	-111	12321
7	11776	8327,3	3448,7	11893531,69
8	12297	8883,5	3413,5	11651982,25
9	11067	11327,3	-260,3	67756,09
10	11239	11384	-145	21025
Total				33297941,03

Kemudian untuk melihat total hasil error dari permalan tersebut maka digunakan rumusan RSME sebagai berikut.

$$RMSE = \sqrt{\frac{\sum(Y_t - Y_{t+1})^2}{n}} \quad (1)$$

Dari rumusan RMSE tersebut didapatkan total hasil error dari minggu ke 4 sampai dengan minggu ke 10 adalah sebagai berikut.

$$RMSE = \sqrt{\frac{33297941,03}{7}} = 5770,44$$

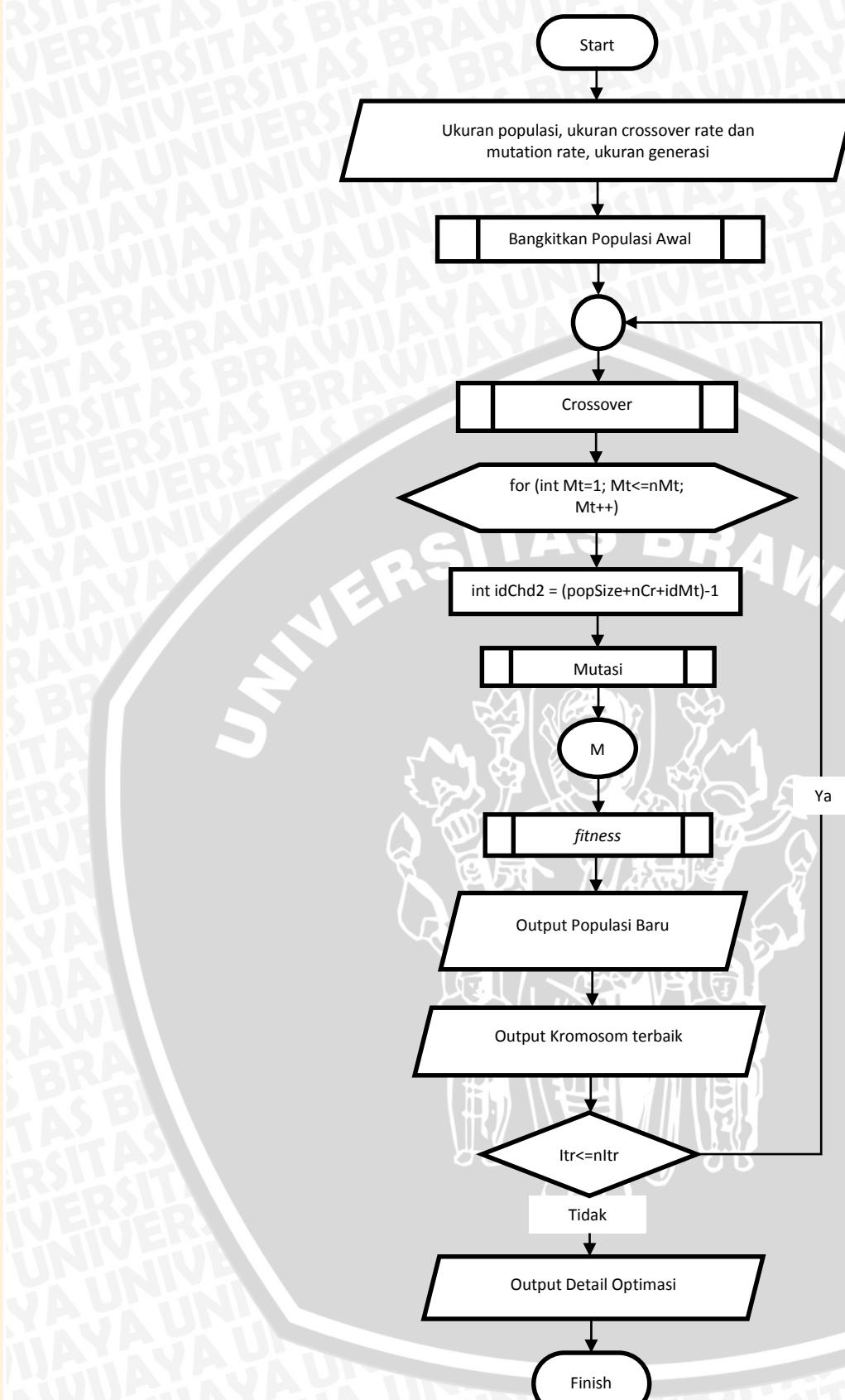
Dari hasil perhitungan RMSE tersebut didapatkan, bahwa hasil perhitungan peramalan dengan menggunakan metode Fuzzy Tsukamoto mendapatkan total nilai error yang cukup besar. Maka untuk dapat menghasilkan total nilai error dalam peramalan permalan yang optimal atau lebih kecil, maka dilakukan penambahan metode Algoritma Genetika. Pada perhitungan dengan metode Algoritma Genetika bertujuan untuk menentukan nilai batasan-batasan kriteria yang digunakan. Hal tersebut dilakukan, agar hasil dari perhitungan Fuzzy Tsukamoto menghasilkan total nilai error yang optimal dalam permasalahan peramalan permintaan.



4.2 Siklus Algoritma Genetika

Pada subbab ini akan menjelaskan proses dalam memperoleh jumlah persediaan barang yang optimal di setiap minggunya. Pada penilitian ini terdapat data permintaan barang semen selama 25 minggu. Data permintaan tersebut akan direpresentasikan menjadi kromosom, kemudian dibangkitkan populasi sebanyak parameter yang diinginkan. Reproduksi dalam penelitian ini dilakukan dengan cara crossover dan mutasi. Offspring yang dihasilkan $popSize \times Crossover rate (cr)$ kemudian ditambah $popSize \times mutation rate (mr)$. Kemudian individu dari populasi awal dan offspring hasil crossover dan mutasi digabungkan untuk proses seleksi. Seleksi dilakukan dengan menghitung fitnes pada masing-masing individu. Individu terbaik merupakan individu yang memiliki *fitness* terbaik setelah n generasi. Berikut alur diagram proses Algoritma Genetika yang direpresentasikan menjadi diagram alir yang diambil pada studi kasus optimasi persediaan barang produksi jilbab oleh Ramuna, MDT & Mahmudy, WF (2015), dapat dilihat pada Gambar 4.1.





Gambar 4.1 Diagram Alir Proses Algortima Genetika

Pada Gambar 4.1 menggambarkan diagram alir Algoritma Gentika untuk meyelesaikan masalah optimasi. Pada langkah pertama dilakukan generate populasi awal secara random n individu. Kemudian pada langkah kedua, melakukan crossover dengan memilih dua induk (parent) secara acak dari

populasi. Langkah berikutnya, yaitu melakukan proses mutasi pada kromosom induk yang dipilih secara random.

Kemudian langkah selanjutnya adalah melakukan perhitungan nilai *fitness* pada masing-masing kromosom. Pada langkah terakhir, melakukan proses seleksi dengan metode elitsm selection dengan tujuan menentukan individu yang akan dibangkitkan pada generasi berikutnya. Hasil akhir pada Algoritma Genetika ini dengan mendapatkan kromosom yang terbaik pada generasi terakhir. Apabila kondisi pada akhir terpenuhi, maka hasil yang didapat merupakan solusi yang terbaik pada seluruh generasi.

4.3 Siklus Penyelesaian Masalah Menggunakan Algoritma Genetika

Pada subbab ini akan menjelaskan penyelesaian suatu permasalahan optimasi prediksi permintaan barang semen dengan menggunakan Algoritma Genetika dan Fuzzy Tsukamoto.

4.3.1 Repräsentasi Kromosom dan Perhitungan Fitness

Repräsentasi kromosom merupakan proses pengkodean dari penyelesaian suatu permasalahan (Mahmudy, 2013). Menurut Gen dan Cheng (2000) didalam metode pengkodean dibedakan berdasarkan jenis symbol yang digunakan sebagai nilai suatu gen, berikut beberapa jenis pengkodean menurut Gen dan Cheng : pengkodean biner, pengkodean real, bilangan bulat dan struktur data.

Repräsentasi kromosom pada penelitian ini menggunakan pengkodean real (real code). Pada pengkodean real memiliki hasil yang optimal dalam permasalahan optimasi, sebab pengkodean real dapat menjangkau beberapa titik solusi jika range solusi berada pada daerah kontinyu (Herrera, Lozano & Verdegay, dalam Mahmudy, 2013). Repräsentasi kromosom dengan real code juga digunakan oleh Pramesti, D., Mahmudy, WF & Indriati (2015) dalam permasalahan optimasi komposisi pakan kambing potong.

Pembentukan kromosom pada penelitian ini dengan membangkitkan nilai random yang didapatkan dari jumlah permintaan barang semen dari suplier pada setiap minggunya. Contoh reprentasi kromosom pada penelitian ini dapat dilihat pada gambar 4.2.

P(t)	Kromosom															
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4
P1	13543	9990	8389	11776	11957	15016	8785	12297	9275	10135	11535	8202	10067	11468	5725	8300

Gambar 4.2 Contoh Repräsentasi Kromosom [Pi] Random

Pada Gambar 4.2 merupakan reprentasi kromosom awal yang dilakukan dengan nilai random setiap kriteria yang digunakan. Kemudian untuk menghasilkan nilai berurutan pada setiap kriteria yang diguanakan, maka dapat dilihat pada Gambar 4.3 yang merupakan hasil dari pengurutan dari setiap nilai random pada setiap kriteria.

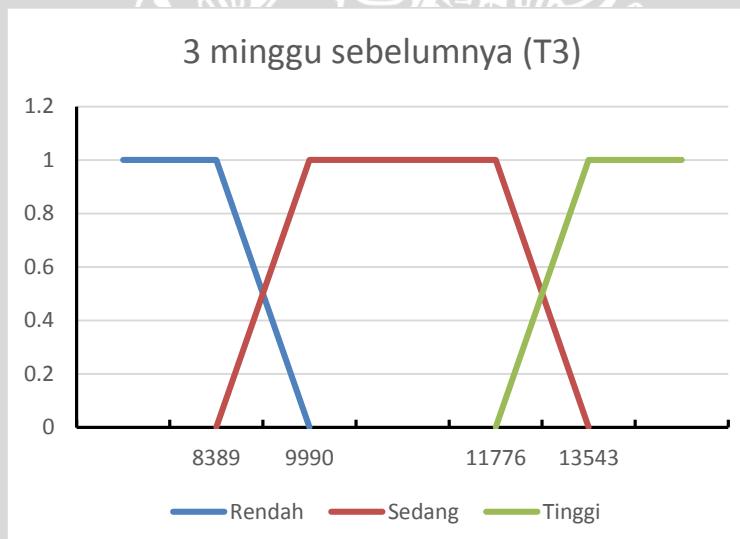
P(t)	Kromosom															
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4
P1	8389	9990	11776	13543	8785	11957	12297	15016	8202	9275	10135	11535	5725	8300	10067	11468

Gambar 4.3 Contoh Representasi Kromosom [Pi] Berurutan

Berikut merupakan keterangan pada Gambar 4.3, sebagai berikut.

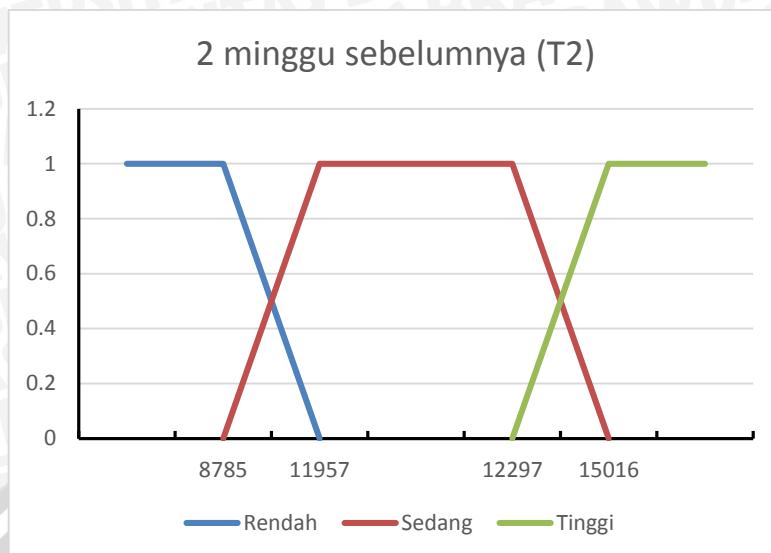
- Pada kolom A merupakan fungsi keanggotaan pada kriteria 3 minggu sebelumnya (T3).
- Pada kolom B merupakan fungsi keanggotaan pada kriteria 2 minggu sebelumnya (T2).
- Pada kolom C merupakan fungsi keanggotaan pada kriteria 1 minggu sebelumnya (T1).
- Pada kolom D merupakan fungsi keanggotaan pada kriteria minggu selanjutnya (T)

Pada Gambar 4.3 tersebut menjelaskan kromosom tersebut memiliki 16 gen yang didapatkan dari setiap batasan kriteria peramalan permintaan, kemudian kromosom tersebut akan dikonversikan sesuai dengan nilai minimal dan maksimal pada masing-masing kriteria yang digunakan. Pada penggunaan batasan tersebut dapat dilihat pada Gambar 4.4 sampai dengan Gambar 4.7 berikut ini.



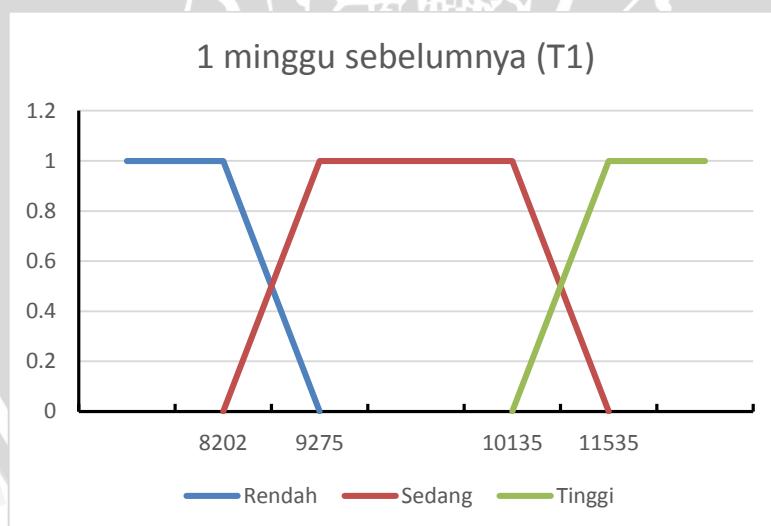
Gambar 4.4 Grafik Kriteria T3

Pada Gambar 4.4 merupakan batasan-batasan pada kriteria 3 minggu sebelumnya (T3). Pada batas atas fungsi keanggotaan rendah terdapat nilai 8389, kemudian batas bawah pada fungsi keanggotaan rendah dengan nilai 9990. Pada batas bawah fungsi keanggotaan sedang dengan nilai 8389, kemudian pada fungsi keanggotaan sedang pada batas atas dengan nilai 9990 sampai dengan batas nilai 11776, dan pada batas bawah pada fungsi keanggotaan sedang dengan nilai 13543. Pada batas bawah fungsi keanggotaan tinggi dengan nilai 11776, dan untuk batas atas pada fungsi keanggotaan tinggi dengan nilai 13543.



Gambar 4.5 Grafik Kriteria T2

Pada Gambar 4.5 merupakan batasan-batasan pada kriteria 2 minggu sebelumnya (T2). Pada batas atas fungsi keanggotaan rendah terdapat nilai 8785, kemudian batas bawah pada fungsi keanggotaan rendah dengan nilai 11957. Pada batas bawah fungsi keanggotaan sedang dengan nilai 8785, kemudian pada fungsi keanggotaan sedang pada batas atas dengan nilai 11957 sampai dengan batas bawah pada fungsi keanggotaan sedang dengan nilai 15016. Pada batas bawah fungsi keanggotaan tinggi dengan nilai 12297, dan untuk batas atas pada fungsi keanggotaan tinggi dengan nilai 15016.



Gambar 4.6 Grafik Kriteria T1

Pada Gambar 4.6 merupakan batasan-batasan pada kriteria 1 minggu sebelumnya (T3). Pada batas atas fungsi keanggotaan rendah terdapat nilai 8202, kemudian batas bawah pada fungsi keanggotaan rendah dengan nilai 9275. Pada batas bawah fungsi keanggotaan sedang dengan nilai 8202, kemudian pada fungsi keanggotaan sedang pada batas atas dengan nilai 9275 sampai dengan batas bawah pada fungsi keanggotaan sedang dengan nilai 10135, dan pada batas bawah pada fungsi keanggotaan tinggi dengan nilai 10135, dan untuk batas atas pada fungsi keanggotaan tinggi dengan nilai 11535.

dengan nilai 11535. Pada batas bawah fungsi keanggotaan tinggi dengan nilai 10135, dan untuk batas atas pada fungsi keanggotaan tinggi dengan nilai 11535.



Gambar 4.7 Grafik Kriteria T

Pada Gambar 4.7 merupakan batasan-batasan pada kriteria minggu selanjutnya (T). Pada batas atas fungsi keanggotaan rendah terdapat nilai 5725, kemudian batas bawah pada fungsi keanggotaan rendah dengan nilai 8300. Pada batas bawah fungsi keanggotaan sedang dengan nilai 5725, kemudian pada fungsi keanggotaan sedang pada batas atas dengan nilai 8300 sampai dengan batas nilai 10067, dan pada batas bawah pada fungsi keanggotaan sedang dengan nilai 11468. Pada batas bawah fungsi keanggotaan tinggi dengan nilai 10067, dan untuk batas atas pada fungsi keanggotaan tinggi dengan nilai 11468.

Pada perhitungan nilai fitnes digunakan untuk mendapatkan solusi yang layak dimasukkan ke dalam algoritma genetika dengan menilai seberapa bagus sebuah kromosom (Mahmudy, 2013). Pada Tabel 4.9 merupakan hasil konversi dari kromosom yang ada pada Gambar 4.2.

Tabel 4.11 Error Fuzzy Tsukamoto

Minggu	Data Permintaan	Hasil Prediksi Tsukamoto	Error (Data Permintaan-Hasil Prediksi)	Error (Data Permintaan-Hasil Prediksi) ²
1	13543	N/A	N/A	N/A
2	11957	N/A	N/A	N/A
3	15016	N/A	N/A	N/A
4	9990	11468	-1478	2184484
5	8785	10767,5	-1982,5	3930306,25
6	8389	8610,4	-221,4	49017,96
7	11776	7896	3880	15054400
8	12297	7639,333333	4657,666667	21693858,78
9	11067	5725	5342	28536964
10	11239	10067	1172	1373584
Total				72822614,99

Kemudian untuk menghitung Nilai *Error* dari keseluruhan total error, maka diperhitungkan dengan menggunakan RMSE yang telah dijelaskan pada bab sebelumnya. Berikut merupakan persamaan perhitungan dengan RMSE.

$$RMSE = \sqrt{\frac{72822614,99}{7}} = 3225,40$$

Untuk perhitungan fitness pada kromosom yang telah dikonversikan, maka dapat dilihat pada persamaan perhitungan berikut ini.

$$fitness = \frac{10000}{RMSE} \quad (2)$$

$$fitness = \frac{10000}{3225,40} = 3,10039$$

Hasil dari perhitungan *fitness* dari pembentukan kromosom sebelumnya dapat dilihat pada tabel 4.10.

Tabel 4.12 Hasil Perhitungan Fitness

P(t)	Kromosom															Fitness = 1/RMSE	
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4	
P1	8389	9990	11776	13543	8785	11957	12297	15016	8202	9275	10135	11535	5725	8300	10067	11468	3,10039

4.3.2 Inisialisasi Populasi Awal

Inisialisasi populasi awal merupakan tahapan dalam metode algoritma genetika dengan membangkitkan sejumlah kromosom yang sesuai dengan ukuran populasi untuk dijadikan anggota dalam populasi awal. Populasi awal dibuat dengan menyesuaikan popSize yang telah diinisialisasikan di awal. Populasi awal yang dibangun pada penilitian ini menggunakan bilangan real dengan range yang telah ditentukan sebelumnya. Berikut pada Tabel 4.11 merupakan contoh inisialisasi populasi awal sebanyak 5 individu yang dilakukan secara Random.

Tabel 4.13 Inisialisasi Populasi Awal Secara Random

P(t)	Kromosom															
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4
P1	13543	9990	8389	11776	11957	15016	8785	12297	9275	10135	11535	8202	10067	11468	5725	8300
P2	11239	11776	9990	8389	10135	8202	5725	11468	10088	8300	15016	12297	11535	9910	5078	13543
P3	9394	11772	11535	10067	12297	5725	11957	8300	9275	5078	11468	8766	11957	11239	15016	5725
P4	11535	10067	5725	8202	8300	11772	11535	9990	15016	11957	8389	9394	11468	8766	8300	11776
P5	10088	12297	9275	10067	11239	8785	8389	15016	11957	8300	5078	5725	11535	11468	8766	13543

Pada Tabel Inisialisasi awal tersebut, maka nilai pada setiap populasi awal tersebut harus diurutkan berdasarkan kriteria-kriteria yang digunakan. Berikut pada Tabel 4.12 merupakan hasil pengurutan inisialisasi populasi awal berdasarkan kriteria-kriteria yang dibutuhkan.

Tabel 4.14 Inisialisasi Populasi Awal Secara Berurutan

P(t)	Kromosom															
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4
P1	8389	9990	11776	13543	8785	11957	12297	15016	8202	9275	10135	11535	5725	8300	10067	11468
P2	8389	9990	11239	11776	5725	8202	10135	11468	8300	10088	12297	15016	5078	9910	11535	13543
P3	9394	10067	11535	11772	5725	8300	11957	12297	5078	8766	9275	11468	5725	11239	11957	15016
P4	5725	8202	10067	11535	8300	9990	11535	11772	8389	9394	11957	15016	8300	8766	11468	11776
P5	9275	10067	10088	12297	8389	8785	11239	15016	5078	5725	8300	11957	8766	11468	11535	13543

4.3.3 Reproduksi

Pada tahap reproduksi terdapat 2 proses, diantaranya proses crossover dan proses mutasi. Pada penitian ini proses crossover yang dilakukan adalah dengan menggunakan metode one-cut point, sedangkan pada proses mutasi menggunakan metode exchange point.

4.3.3.1 Perhitungan Crossover

Pada perhitungan crossover menggunakan metode extends intermediate. Metode extends intermediate melakukan dengan cara memilih secara acak dua induk (parent) secara acak. Dengan menggunakan metode extends intermediate crossover akan menghasilkan offspring dari kombinasi nilai dua induk yang terpilih. Misalkan P_1 dan P_2 merupakan 2 induk yang terpilih, maka offspring C1 dan C2 dapat dibangkitkan dengan persamaan (3) dan (4).

$$C_1 = P_1 + \alpha(P_2 - P_1) \quad (3)$$

$$C_2 = P_2 + \alpha(P_1 - P_2) \quad (4)$$

Pada contoh permasalahan pada perhitungan crossover menggunakan nilai cr sebesar 0,4 dan ukuran popSize sebesar 5, maka jumlah offspring yang akan dihasilkan sebanyak $0,4 * 5 = 2$ offspring. Kemudian untuk nilai alpha (α) yang digunakan dalam perhitungan crossover extends intermediate sebesar 1,25. Berikut pada Tabel 4.13 merupakan contoh hasil Crossover dengan metode extends intermediate.

Tabel 4.15 Crossover Extends Intermediate

P(t)	Kromosom							
	A1	A2	A3	A4	B1	B2	B3	B4
P1	8389	9990	11776	13543	8785	11957	12297	15016
P5	9275	10067	10088	12297	8389	8785	11239	15016
C1	9496,5	10086,25	9666	11985,5	8290	7992	10974,5	15016
C2	8167,5	9970,75	12198	13854,5	8884	12750	12561,5	15016

Tabel 4.16 Crossover Extends Intermediate (Lanjutan)

P(t)	Kromosom							
	C1	C2	C3	C4	D1	D2	D3	D4
P1	8202	9275	10135	11535	5725	8300	10067	11468
P5	5078	5725	8300	11957	8766	11468	11535	13543
C1	4297	4837,5	7841,25	12062,5	9526,25	12260	11902	14061,75
C2	8983	10162,5	10593,75	11429,5	4964,75	7508	9700	10949,25

4.3.3.2 Perhitungan Mutasi

Pada perhitungan mutasi ini menggunakan metode exchange point. Metode exchange point ini dengan melakukan penukaran kromosom satu dengan kromosom lain.

Berikut pada Tabel 4.14 merupakan contoh mutasi dengan menggunakan metode exchange point.

Tabel 4.17 Mutation Exchange Point

P(t)	Kromosom															
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4
P2	8389	9990	11239	11776	5725	8202	10135	11468	8300	10088	12297	15016	5078	9910	11535	13543
P4	5725	8202	10067	11535	8300	9990	11535	11772	8389	9394	11957	15016	8300	8766	11468	11776
C3	8389	9990	10088	11776	5725	8202	10135	11468	8300	11239	12297	15016	5078	9910	11535	13543
C4	5725	8202	10067	11535	8300	8766	11535	11772	8389	9394	11957	15016	8300	9990	11468	11776

4.3.4 Evaluasi dan Seleksi

Pada proses evaluasi merupakan suatu proses yang digunakan dalam mengevaluasi keseluruhan nilai *fitness* yang ada pada setiap kromosom. Dalam proses evaluasi terdapat baik atau buruknya dari nilai *fitness* di setiap kromosom yang akan dievaluasi, semakin tinggi nilai *fitness* maka semakin baik kromosom tersebut untuk dijdaikan solusi (Mahmudy, 2013). Dari hasil yang didapatkan dari proses *crossover* dan proses *mutasi* akan dilakukan pencarian nilai *fitness*. Kromosom yang terpilih nantinya akan dijadikan *parent* pada generasi berikutnya dengan memiliki nilai *ftness* yang baik. Berikut pada Tabel 4.15 merupakan tabel hasil dari parent dan seluruh *offspring* yang dihasilkan dari proses *crossover* dan proses *mutasi* sebelumnya.

Tabel 4.18 Evaluasi

P(t)	Kromosom							
	A1	A2	A3	A4	B1	B2	B3	B4
P1	8389	9990	11776	13543	8785	11957	12297	15016
P2	8389	9990	11239	11776	5725	8202	10135	11468
P3	9394	10067	11535	11772	5725	8300	11957	12297

P(t)	Kromosom							
	A1	A2	A3	A4	B1	B2	B3	B4
P4	5725	8202	10067	11535	8300	9990	11535	11772
P5	9275	10067	10088	12297	8389	8785	11239	15016
C1	9496,5	10086,25	9666	11985,5	8290	7992	10974,5	15016
C2	8167,5	9970,75	12198	13854,5	8884	12750	12561,5	15016
C3	8389	9990	10088	11776	5725	8202	10135	11468
C4	5725	8202	10067	11535	8300	8766	11535	11772

Tabel 4.196 Evaluasi (Lanjutan)

P(t)	Kromosom								Fitness = 1/RMSE
	C1	C2	C3	C4	D1	D2	D3	D4	
P1	8202	9275	10135	11535	5725	8300	10067	11468	3,10039
P2	8300	10088	12297	15016	5078	9910	11535	13543	5,63024
P3	5078	8766	9275	11468	5725	11239	11957	15016	2,06155
P4	8389	9394	11957	15016	8300	8766	11468	11776	6,03305
P5	5078	5725	8300	11957	8766	11468	11535	13543	0,02956
C1	4297	4837,5	7841,25	12062,5	9526,25	12260	11902	14061,75	0,02669
C2	8983	10162,5	10593,75	11429,5	4964,75	7508	9700	10949,25	2,65199
C3	8300	11239	12297	15016	5078	9910	11535	13543	4,34590
C4	8389	9394	11957	15016	8300	9990	11468	11776	5,90188

Dari setiap individu menghasilkan nilai fitness untuk mengetahui kebugaran pada setiap individu yang dimiliki, kemudian hasil fitness dari setiap individu tersebut akan dilakukan proses seleksi. Pada tahap selanjutnya adalah tahap seleksi, tahap seleksi dilakukan untuk mendapatkan hasil kromosom terbaik yang akan dipertahankan untuk dijadikan populasi pada generasi berikutnya. Pada penelitian ini proses seleksi menggunakan metode elitism. Tahapan seleksi dengan metode elitism akan mengurutkan kromosom berdasarkan nilai *fitness* yang tertinggi hingga terendah, kemudian mengambil kromosom sebanyak jumlah populasi awal. Metode *elitsm* akan menjamin bahwa salah satu individu yang terbaik akan selalu lolos ke dalam generasi berikutnya (Mahmudy, 2013). Berikut pada Table 4.16 merupakan table hasil seleksi pada setiap individu dengan menggunakan metode *elitsm*.

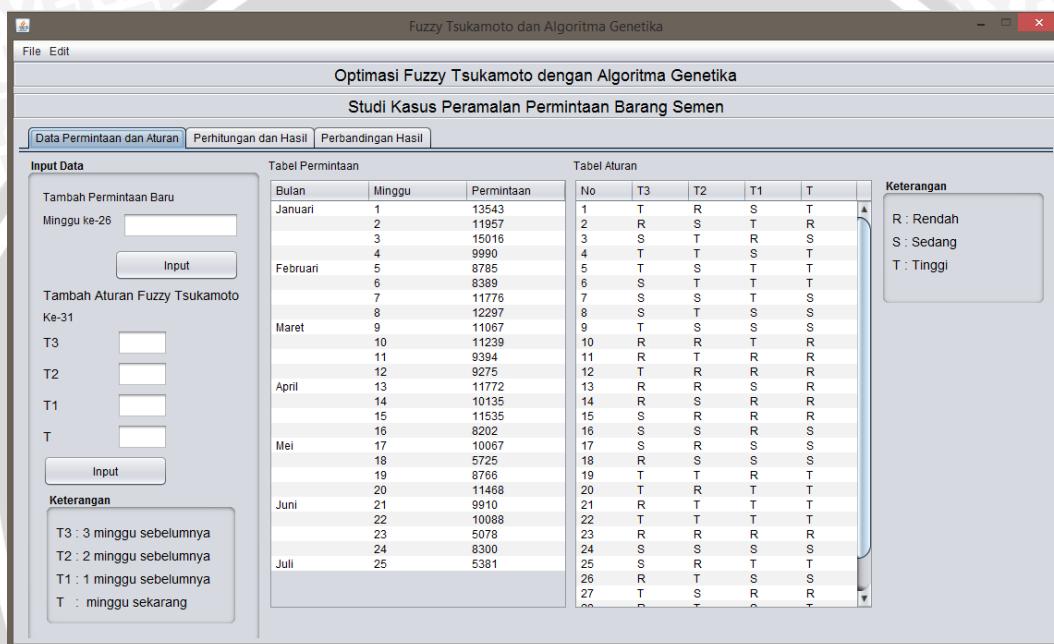
Tabel 4.20 Seleksi Elitsm

P(t-1)	P(t) asal	Fitness
P1	P4	6,03305
P2	C4	5,90188
P3	P2	5,63024
P4	C3	4,34590
P5	P1	3,10039

4.4 Perancangan User Interface

Dalam peracangan user *interface* aplikasi yang dibuat terdapat 3 halaman utama antara lain halaman input data & aturan fuzzy, halaman proses perhitungan & hasil perhitungan, dan halaman hasil proses peramalan. Berikut merupakan rancangan masing-masing halaman yang akan dijabarkan pada sub-bab selanjutnya.

4.4.1 Rancangan User Interface Halaman Input Data Permintaan dan Aturan Fuzzy



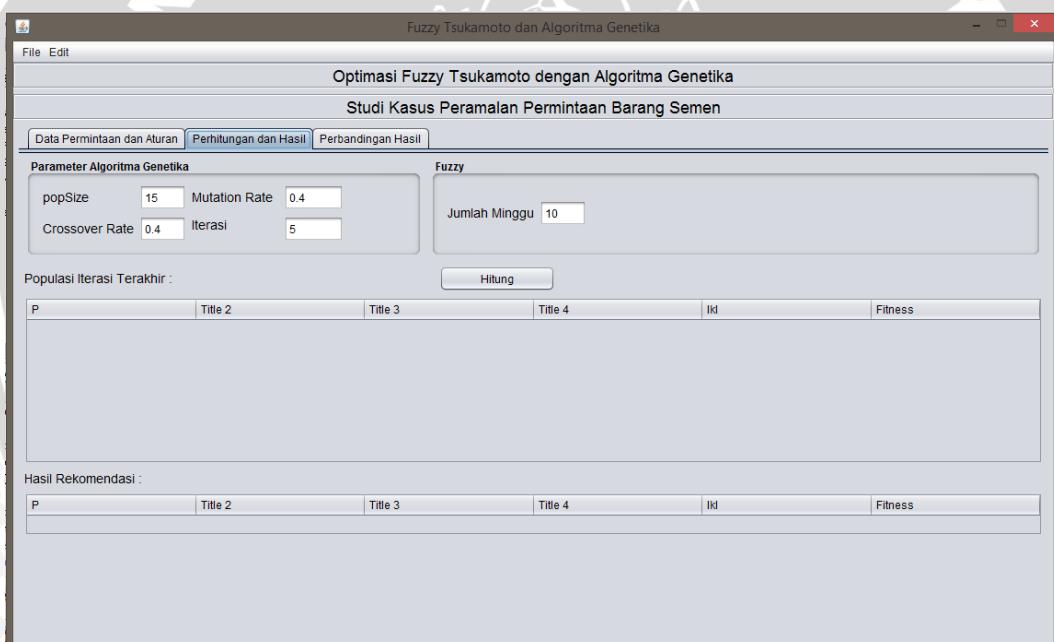
Gambar 4.8 Rancangan User Interface Halaman Input Data Permintaan dan Aturan Fuzzy

Gambar 4.8 merupakan rancangan user interface pada halaman utama pada aplikasi Peramalan Permintaan Barang Produksi Semen menggunakan Fuzzy Tsukamoto dan Algoritma Genetika. Berikut merupakan beberapa penjelasan menegenai textfield dan tabel yang digunakan dalam aplikasi Optimasi Peramalan Permintaan menggunakan Fuzzy Tsukamoto dan Algoritma Genetika, yaitu :

1. Textfield ‘Tambah Permintaan Baru’ digunakan untuk menginputkan data jumlah permintaan barang yang sudah terjual pada minggu tersebut.
2. Textfield ‘Tambah Aturan Fuzzy Tsukamoto’ digunakan untuk menginputkan aturan-aturan fuzzy.
3. Textfield ‘T3’ digunakan untuk menginputkan aturan fuzzy pada kriteria 3 minggu sebelumnya.

4. Textfield 'T2' digunakan untuk menginputkan aturan *fuzzy* pada kriteria 2 minggu sebelumnya.
5. Textfield 'T1' digunakan untuk menginputkan aturan *fuzzy* pada kriteria 1 minggu sebelumnya.
6. Textfield 'T' digunakan untuk menginputkan aturan *fuzzy* pada kriteria minggu sekarang.
7. Tabel 'Permintaan' merupakan tabel data permintaan yang telah diinputkan ke dalam database, data tersebut digunakan untuk melakukan perbandingan antara hasil Peramalan dengan hasil data asli dan data tersebut juga digunakan untuk proses pemilihan kromosom awal secara random.
8. Tabel 'Aturan' merupakan tabel hasil aturan yang telah diinputkan ke dalam database, aturan tersebut digunakan untuk melakukan proses perhitungan fuzzyifikasi dalam mendapatkan hasil peramalan.

4.4.2 Rancangan User Interface Halaman Perhitungan dan Hasil



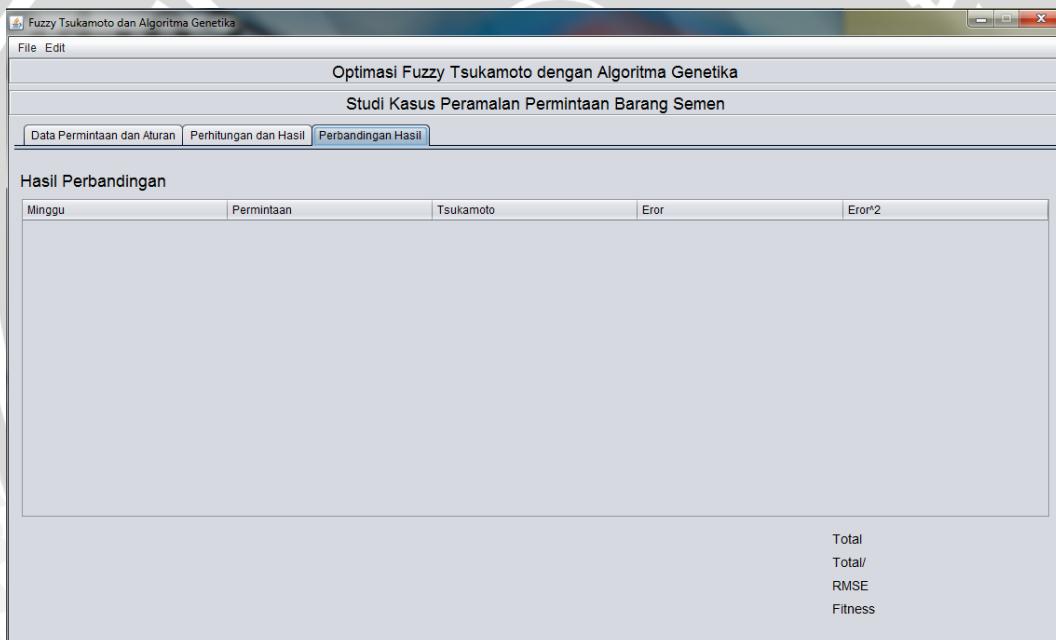
Gambar 4.9 Rancangan User Interface Halaman Perhitungan Dan Hasil

Gambar 4.9 merupakan rancangan user interface pada halaman kedua pada aplikasi Optimasi Peralaman Permintaan Barang Produksi Semen menggunakan Fuzzy Tsukamoto dan Algoritma Genetika. Berikut merupakan penjelasan yang digunakan dalam Optimasi Peramalan Permintaan menggunakan Fuzzy Tsukamoto dan Algoritma Genetika, yaitu :

1. Textfield 'PopSize' digunakan untuk inputan ukuran populasi awal dalam perhitungan Algoritma Genetika.
2. Textfield 'Crossover Rate' digunakan untuk inputan ukuran proses crossover dalam perhitungan Algoritma Genetika.

3. Textfield 'Mutation Rate' digunakan untuk inputan ukuran proses mutation dalam perhitungan Algoritma Genetika.
4. Textfield 'Iterasi' digunakan untuk inputan proses perulangan dalam perhitungan dengan metode Algoritma Genetika dan Fuzzy Tsukamoto.
5. Textfield 'Jumlah Minggu' digunakan untuk inputan jumlah minggu yang digunakan dalam perhitungan Algoritma Genetika dan Fuzzy TSukamoto.
6. Tabel 'Populasi Iterasi Terakhir' digunakan untuk menampilkan hasil dari proses perhitungan peramalan dengan menggunakan metode Algoritma Genetika dan Fuzzy Tsukamoto.
7. Tabel 'Hasil Rekomendasi' digunakan untuk menampilkan hasil perhitungan dengan nilai fitness yang terbesar, dari hasil nilai fitness tersebut akan digunakan untuk rekomendasi dalam perhitungan peramalan.

4.4.3 Rancangan *User Interface* Halaman Perbandingan Hasil

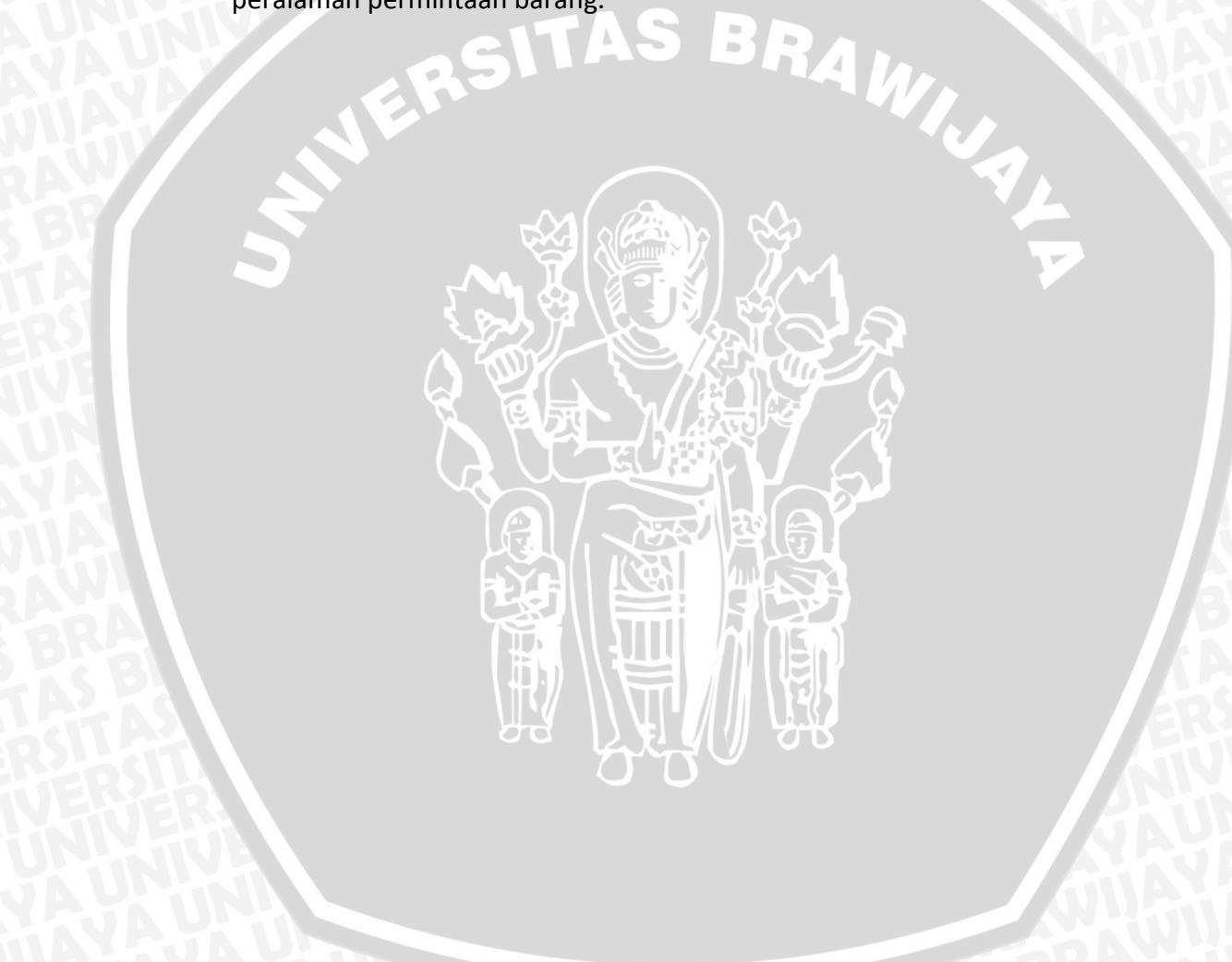


Gambar 4.10 Rancangan *User Interface* Halaman Perbandingan Hasil

Gambar 4.10 merupakan rancangan user interface pada halaman ketiga pada aplikasi Optimasi Peralaman Permintaan Barang Produksi Semen menggunakan Fuzzy Tsukamoto dan Algoritma Genetika. Berikut merupakan penjelasan yang digunakan dalam Optimasi Peramalan Permintaan menggunakan Fuzzy Tsukamoto dan Algoritma Genetika, yaitu :

1. Tabel 'Hasil Perbandingan' merupakan tabel hasil perbandingan antara data asli dengan hasil peramalan menggunakan Algoritma Genetika dan Fuzzy Tsukamoto, hasil peramalan menggunakan Algoritma Genetika dan Fuzzy Tsukamoto didapatkan dari nilai fitness yang terbesar.

2. Textfield ‘Total’ merupakan total dari jumlah nilai error yang didapatkan dari hasil pengurangan antara data asli dengan hasil peramalan.
3. Textfield ‘Total/’ merupakan hasil pembagian antara total jumlah error dengan jumlah minggu yang digunakan dalam perbandingan data asli dengan hasil peramalan.
4. Textfield ‘RMSE’ merupakan hasil perhitungan dari jumlah nilai error yang didapatkan dari hasil pembagian antara total jumlah error dengan jumlah minggu yang digunakan dalam perbandingan, dimana nilai RMSE digunakan untuk menentukan nilai fitness.
5. Textfield ‘fitness’ merupakan hasil akhir untuk rekomendasi dalam peralaman permintaan barang.



BAB 5 IMPLEMENTASI

5.1 Implementasi Program

Pada subbab berikut akan menjelaskan tentang mengenai struktur data dan source code yang digunakan dalam aplikasi Peramalan Permintaan Barang menggunakan Fuzzy Tsukamoto dan Algoritma Genetika.

5.1.1 Proses Ambil Data dan Aturan Fuzzy

Proses pengambilan data yang digunakan dalam metode Algoritma Genetika dan Fuzzy Tsukamoto adalah dengan mengambil data yang sebelumnya telah disimpan dalam database dengan nama file data_peramalan.sql. Jumlah data keseluruhan pada data permintaan terdapat 25 permintaan, dan pada data peraturan fuzzy terdapat 25 aturan yang digunakan. Data permintaan yang digunakan dalam penilitian ini diperoleh dari perusahaan Koperasi Warga Semen Gresik (KWSG). Berikut merupakan source code dalam proses pengambilan data dari database yang digunakan untuk proses peramalan dengan Algoritma Genetika dan Fuzzy Tsukamoto, yang ditunjukkan pada Tabel 5.1.

Tabel 5.1 Source Code Ambil Data

1	try {
2	Class.forName("com.mysql.jdbc.Driver");
3	con =
4	DriverManager.getConnection("jdbc:mysql://localhost/data_peramalan
5	", "root", "");
6	st = con.createStatement();
7	rs = st.executeQuery("SELECT * from aturan");
8	rs.last();
9	int size = rs.getRow();
10	rs.beforeFirst();
11	int no = 0;
12	aturan = new String[size][4];
13	while (rs.next()) {
14	aturan[no][0] = rs.getString(2); //t3
15	aturan[no][1] = rs.getString(3); //t2
16	aturan[no][2] = rs.getString(4); //t1
17	aturan[no][3] = rs.getString(5); //t
18	no++;
19	}
20	rs = st.executeQuery("SELECT * from permintaan");
21	rs.last();
22	size = rs.getRow();
23	permintaan = new int[size];
24	rs.beforeFirst();
25	no = 0;
26	while (rs.next()) {
27	permintaan[no] = rs.getInt(2);
28	System.out.println(permintaan[no] + " ");
29	no++;
30	}
31	} catch (ClassNotFoundException SQLException ex) {
32	System.out.println("eror: " + ex);
33	}



5.1.2 Proses Inisialisasi Kromosom

Proses pembangkitan kromosom awal pada populasi dilakukan berdasarkan ukuran populasi (*popSize*) yang diinputkan. Kromosom pada penelitian ini dibangkitkan secara random, nilai kromosom yang digunakan didapatkan dari jumlah data permintaan pada setiap minggunya. Panjang kromosom dalam penelitian ini ditentukan sepanjang 16, sehingga populasi kromosom dibangkitkan sepanjang jumlah populasi dengan lebar 16. Berikut pada Tabel 5.2 merupakan source code proses inisialisasi kromosom.

Tabel 5.2 Source Code Inisialisasi Kromosom

1	public static void main(String[] args) {
2	FIST_ALGEN htg = new FIST_ALGEN();
3	
4	htg.setWidth(16); //width = panjang cromosom
5	htg.setPopsize(5);
6	htg.setCr(0.6);
7	htg.setMr(0.4);
8	htg.setJumlahMinggu(15);
9	htg.setChildSize();

5.1.3 Proses Generate Random Parent Awal

Pada proses Generate random parent awal merupakan proses pembentukan parent awal yang dilakukan secara random. Nilai dari pembentukan parent awal didapatkan dari data permintaan perminggu yang diambil secara random, setelah didapatkan parent dengan nilai random maka selanjutnya adalah mengurutkan nilai pada parent tersebut sesuai dengan kriteria yang di gunakan. Berikut merupakan source code pada proses generate random parent awal.

Tabel 5.3 Source Code Generate Random Parent Awal

1	int width = 16;
2	Object parent1[][][], offspring[][][], evalSel[][][];
3	System.out.println("parent random:");
4	parent1 = htg.generateRandParent(width); // perhitungan
5	pertama, hasilkan parent dengan cromosom random dgn panjang sesuai
6	variabel width
7	System.out.println("parent urut:");
8	for (int i = 0; i < htg.popsize; i++) {
9	for (int j = 0; j < width + 2; j++) {
10	System.out.print(parent1[i][j] + " ");
11	}
12	System.out.println("");
13	}

5.1.4 Proses Crossover

Pada perhitungan Crossover dilakukan dengan menggunakan Extends Intermediate. Proses dalam crossover Extends Intermediate adalah dengan memilih 2 parent secara random, kemudian nilai pada parent tersebut dimasukkan ke dalam rumusan sebagai berikut



$$C_1 = P_1 + \alpha(P_2 - P_1)$$

$$C_2 = P_2 + \alpha(P_1 - P_2)$$

Nilai alpha yang digunakan dalam penelitian peramalan permintaan sebesar 1,25. Berikut pada Tabel 5.4 merupakan source code dalam proses crossover extends intermediate.

Tabel 5.4 Source Code Crossover

<pre> 1 Object[][][] crossover(Object[][][] parent) { 2 int offspring = (int) Math.round(cr * popsize), rand = 3 0;//set ukuran offspring sepanjang cr kali popsize 4 double alpha = 1.25; 5 // System.out.println(""); 6 randParent = shuffledArray(0, popsize - 1);//hasilkan 7 parent yg telah dishuffle/random sebanyak popsize 8 for (int i = 0; i < offspring; i = i + 2) {//perulangan 9 perhitungan crossover dengan increment 2, i=(i+2), karena dilakukan 10 untuk satu pasang parent 11 //perhitungan child 1 12 child[i][0] = parent[randParent[i]][0] + " C" + (i + 13 1); //child[i][0]-->indeks 0 isi nama sesuai hasil parent yg 14 terandom 15 for (int j = 1; j < width + 1; j++) {//perhitungan 16 crossover 17 child[i][j] = 18 duaAngka.format(Double.valueOf(parent[randParent[i]][j].toString()) 19) 20 + bulatkan(alpha * 21 (Double.valueOf(parent[randParent[i + 1]][j].toString()) - 22 Double.valueOf(parent[randParent[i]][j].toString()), 1)); 23 } 24 child[i][width + 1] = 25 fitness(child[i]); //child[i][width + 1]--> indeks terakhir diisi 26 fitness 27 //child 2 28 child[i + 1][0] = parent[randParent[i + 1]][0] + " C" 29 + (i + 2); 30 for (int j = 1; j < width + 1; j++) { 31 child[i + 1][j] = 32 duaAngka.format(Double.valueOf(parent[randParent[i + 33 1]][j].toString()) 34 + bulatkan(alpha * 35 (Double.valueOf(parent[randParent[i]][j].toString()) - 36 Double.valueOf(parent[randParent[i + 1]][j].toString()), 1)); 37 } 38 child[i + 1][width + 1] = fitness(child[i]); 39 } 40 } </pre>

5.1.5 Proses Mutation

Pada proses mutasi, dilakukan dengan pemilihan titik mutasi (mutation exchange point) secara random. Pemilihan titik mutasi berdasarkan parent yang telah ditentukan sebelumnya dengan memilih parent secara random. Proses dalam pemilihan titik untuk melakukan proses mutasi dengan pemilihan titik pada kedua parent yang kemudian menukar nilai antara titik yang terpilih pada parent 1 dengan parent 2. Berikut pada Tabel 5.4 merupakan source code proses mutation exchange point.



Tabel 5.5 Source Code Mutasi

```

1   Object[][] mutasi(Object[][][] parent) {
2       int offspring = (int) Math.round(mr * popsize),
3       offspringCR = (int) Math.round(cr * popsize), point[]; //set ukuran
4       offspring sepanjang mr kali popsize
5       for (int i = offspringCR; i < offspring + offspringCR;
6       i++) {//perulangan dimulai dari banyak offsping sebelumnya pada
7       crossover
8           point = shuffledArray(1, width); //hasilkan point
9           random unutk mutasi
10          System.arraycopy(parent[randParent[i]], 1, child[i],
11          1, width);
12          child[i][0] = parent[randParent[i]][0] + " C" + (i +
13          1); //nama
14          child[i][point[0] + 1] =
15          parent[randParent[i]][point[1] + 1]; //swap point mutasi
16          child[i][point[1] + 1] =
17          parent[randParent[i]][point[0] + 1]; //swap point mutasi
18          child[i][width + 1] =
19          fitness(child[i]); //child[i][width + 1]--> indeks terakhir diisi
20          fitness
21      }
22
23      return child;
24  }

```

5.1.6 Proses Evaluasi Seleksi

Pada proses evaluasi individu parent dan child akan digabung menjadi satu ke dalam array gabungan. Kemudian hasil dalam array gabungan tersebut akan dilakukan proses seleksi. Seleksi yang digunakan adalah seleksi elitsm, dimana semua individu parent dan child tersebut akan diurutkan berdasarkan nilai fitness yang terbesar hingga yang terkecil. Individu yang terpilih merupakan individu yang memiliki nilai fitness yang terbesar. Individu yang terpilih merupakan individu teratas sesuai dengan jumlah PopSize awal. Berikut pada Tabel 5.6 merupakan source code Evaluasi Seleksi pada peramalan permintaan barang dengan Algoritma Genetika dan Logika Fuzzy.

Tabel 5.6 Source Code Evaluasi Seleksi

```

1   Object[][] evaluasiSeleksi(Object[][][] parent, Object[][][] child)
2  {
3      int offspring = (int) Math.round(mr * popsize) + (int)
4      Math.round(cr * popsize);
5      Object newParent[][], gabungan[][] = new Object[popsize +
6      offspring][parent[0].length];
7      //gabung parent+child lalu disimpan pada array gabungan
8      System.arraycopy(parent, 0, gabungan, 0, popsize); //array
9      gabungan pada indeks 0 sampai popsize diisi parent
10     System.arraycopy(child, 0, gabungan, popsize,
11     offspring); //array gabungan pada indeks popsize sampai offspirng
12     diisi child
13     //      System.out.println("\nevaluasi");
14     //      for (int i = 0; i < popsize * 2; i++) {
15     //          System.out.println("");
16     //          for (int j = 0; j < width + 2; j++) {
17     //              System.out.print(gabungan[i][j] + " ");

```



```

18    //          }
19    //      }
20    //seleksi
21    newParent = insertionSort(gabungan); //sort fitnesses
menggunakan algoritma insertionsort
22    for (int i = 0; i < popsize; i++) { //set nama baru untuk
parent yg telah diurutkan
23        newParent[i][0] = "P" + (i + 1);
24    }
25    //      System.out.println("\nnsort fitnes masih sama");
26    //      for (int i = 0; i < popsize*2; i++) {
27    //          System.out.println("");
28    //          for (int j = 0; j < width + 2; j++) {
29    //              System.out.print(newParent[i][j] + " ");
30    //          }
31    //      }
32    //      System.out.println("");
33    //      System.arraycopy((insertionSort(gabungan)), 0, newParent,
34    //      0, popsize); //10 fitness tertinggi dijadikan parent pada generasi
selanjutnya jika popsize=10
35    //      return newParent;
36    }
37
38
39
}

```

5.1.7 Proses Perhitungan Error dan Fitness

Pada proses perhitungan error digunakan untuk menentukan hasil yang didapatkan dari peramalan menggunakan algoritma Genetika dan Logika Fuzzy. Perhitungan error dalam penelitian ini menggunakan perhitungan RMSE, hasil dari RMSE tersebut akan digunakan untuk menghitung nilai fitness. Berikut pada Tabel 5.7 merupakan source code perhitungan Error dan Fitness.

Tabel 5.7 Source Code Error dan Fitness

1 double fitness(Object chromos[]) { 2 int no = -1; 3 double fitness, eror = 0, rmse, fist[]; 4 fist = fuzzy(chromos); 5 setFIST(fist); 6 for (int i = 4; i < jumlahMinggu + 1; i++) { 7 eror = eror + Math.pow(permintaan[i-1] - fist[i], 2); 8 } 9 rmse = Math.sqrt(eror / (jumlahMinggu - 3)); 10 fitness = 10000 / rmse; 11 return fitness; 12 }

5.1.8 Proses Perhitungan Fuzzyifikasi

Proses Perhitungan Fuzzyifikasi menentukan hasil peramalan dengan mengambil nilai parent dari setiap kriteria yang digunakan yang kemudian dimasukkan ke dalam rumusan fuzzyifikasi berdasarkan kriteria yang digunakan dalam penelitian ini. Berikut merupakan Tabel 5.8 source code proses Fuzzyifikasi.

Tabel 5.8 Source Code Fuzzyifikasi

1 double[] fuzzy(Object chromos[]) { 2 double Tsukamoto = 0, X, Chromosom[] = 3 convertObjtoDbl(chromos), konversiBatas[][] = new double[4][4],

```
4 fist[] = new double[jumlahMinggu + 1];
5 //      System.out.println("");
6 //      for (int i = 0; i < 16; i++) {
7 //          System.out.print(Chromosom[i] + " ");
8 //      }
9 //      System.out.println("");
10 //batas tiap minggu
11 //      for (int i = 4; i < jumlahMinggu + 1; i++) {//perulangan
12 tiap minggu
13 //l untuk indek pengambilan 4 chromosom
14 //j untuk ambil data dari array permintaan
15 //T untuk indek T1,T2,T3
16 //      for (int j = i - 4, l = 0, T = 0; j < i - 1; j++, l =
17 l + 4, T++) {
18 //          konversiBatas[T][0] = permintaan[j];
19 //          X = konversiBatas[T][0];
20 //Rendah
21 //          if (X <= Chromosom[1]) {
22 //              konversiBatas[T][1] = 1;
23 //          } else if (X > Chromosom[1]
24 //                     && X < Chromosom[1 + 1]) {
25 //              konversiBatas[T][1] = Chromosom[1 + 1] - (X /
26 (Chromosom[1 + 1] - Chromosom[1]));
27 //          } else if (X > Chromosom[1 + 1]) {
28 //              konversiBatas[T][1] = 0;
29 //          }
30 //Sedang
31 //          if (X < Chromosom[1]) {
32 //              konversiBatas[T][2] = 0;
33 //          } else if (X > Chromosom[1]
34 //                     && X < Chromosom[1 + 1]) {
35 //              konversiBatas[T][2] = X - (Chromosom[1] /
36 (Chromosom[1 + 1] - Chromosom[1]));
37 //          } else if (X >= Chromosom[1 + 1]
38 //                     && X <= Chromosom[1 + 2]) {
39 //              konversiBatas[T][2] = 1;
40 //          } else if (X > Chromosom[1 + 2]
41 //                     && X < Chromosom[1 + 3]) {
42 //              konversiBatas[T][2] = Chromosom[1 + 3] - (X /
43 (Chromosom[1 + 3] - Chromosom[1 + 2]));
44 //          } else if (X > Chromosom[1 + 3]) {
45 //              konversiBatas[T][2] = 0;
46 //          }
47 //Tinggi
48 //          if (X < Chromosom[1 + 2]) {
49 //              konversiBatas[T][3] = 0;
50 //          } else if (X > Chromosom[1 + 2]
51 //                     && X < Chromosom[1 + 3]) {
52 //              konversiBatas[T][3] = X - (Chromosom[1 + 2] /
53 (Chromosom[1 + 3] - Chromosom[1 + 2]));
54 //          } else if (X > Chromosom[1 + 3]) {
55 //              konversiBatas[T][3] = 1;
56 //          }
57 //          System.out.println(" -" + permintaan[j]);
58 //          System.out.print(konversiBatas[T][1] + " ");
59 //          System.out.print(konversiBatas[T][2] + " ");
60 //          System.out.print(konversiBatas[T][3] + " ");
61 //      }
62 //Tsukamoto
63 fist[i] = Tsukamoto(Chromosom, konversiBatas);
64 //      System.out.println("");
65 }
66
67 return fist;
68 }
```

5.1.9 Proses Inferensi

Pada proses Inferensi merupakan proses mengambil nilai minimal dari setiap kriteria yang digunakan. Pengambilan nilai minimal tersebut dilakukan berdasarkan aturan yang telah dimasukkan nilai fuzzyifikasi sebelumnya. Berikut pada Tabel 5.9 source code proses Inferensi dalam metode Tsukamoto.

Tabel 5.9 Source Code Inferensi

```

1 double Tsukamoto(double Chromosom[], double konversiBatas[][][]) {
2     double predikat, jmlPredikat = 0, z, predikatKaliZ,
3     tsukamoto = 0, konversiAturan[][] = new double[30][4];
4     for (int j = 0; j < aturan.length; j++) {//jumlah aturan
5
6         for (int k = 0; k < 3; k++) {//perulangan t3 t2 t1
7             System.out.print(aturan[j][k] + " ");
8             if (aturan[j][k].equalsIgnoreCase("R")) {
9                 konversiAturan[j][k] = konversiBatas[k][1];
10            } else if (aturan[j][k].equalsIgnoreCase("S")) {
11                konversiAturan[j][k] = konversiBatas[k][2];
12            } else {
13                konversiAturan[j][k] = konversiBatas[k][3];
14            }
15        }
16        System.out.println("");
17        System.out.println(konversiAturan[j][0] + " " +
18        konversiAturan[j][1] + " " + konversiAturan[j][2] + " ");
19        predikat = cekMin(konversiAturan[j]);
20    }

```

5.1.10 Proses Perhitungan Defuzzyifikasi

Proses perhitungan Defuzzyifikasi adalah menentukan nilai z yang nantinya akan digunakan untuk mendapatkan nilai peramalan dengan menggunakan metode Tsukamoto. Dalam perhitungan defuzzyifikasi dilakukan dengan memasukkan nilai yang didapatkan dari inferensi yang kemudian dimasukkan ke dalam rumusan defuzzyifikasi berdasarkan kriteria yang digunakan. Berikut pada Tabel 5.10 merupakan proses Defuzzyifikasi dalam penelitian ini.

Tabel 5.10 Source Code Defuzzyifikasi

```

1 jmlPredikat = jmlPredikat + predikat;
2     if (aturan[j][3].equalsIgnoreCase("R")) {
3         z = Chromosom[13] - (predikat * (Chromosom[13] -
4 Chromosom[12]));
5     } else if (aturan[j][3].equalsIgnoreCase("S")) {
6         z = Chromosom[12] - (predikat * (Chromosom[13] -
7 Chromosom[12]));
8     } else {//t1
9         z = Chromosom[14] - (predikat * (Chromosom[15] -
10 Chromosom[14]));
11     }

```

5.1.11 Proses Perhitungan α -predikat x z

Proses perhitungan α -predikat x z merupakan proses dimana nilai inferensi yang didapatkan dikalikan dengan nilai z berdasarkan kriteria yang digunakan. Hasil



dari perhitungan α -predikat $x \cdot z$ nantinya akan digunakan dalam tahap Tsukamoto. Berikut pada Tabel 5.11 merupakan source code perhitungan α -predikat $x \cdot z$ dalam penelitian peramalan permintaan barang.

Tabel 5.11 Source Code Perhitungan α -predikat $x \cdot z$

1	predikatKaliZ = predikat * Z;
2	// System.out.println("kaliZ " + predikatKaliZ);
3	tsukamoto = tsukamoto + predikatKaliZ;
4	}

5.1.12 Proses Perhitungan Tsukamoto

Pada proses perhitungan Tsukamoto merupakan proses akhir dalam perhitungan Fuzzy Tsukamoto. Proses untuk menentukan hasil Tsukamoto dengan menghitung nilai rata-rata pada nilai α -predikat $x \cdot z$ dan nilai inferensi. Kemudian nilai hasil rata-rata tersebut dibagi antara nilai α -predikat $x \cdot z$ dan nilai inferensi. Berikut merupakan Tabel 5.12 proses perhitungan Tsukamoto dalam penelitian peramalan permintaan barang.

Tabel 5.12 Source Code Perhitungan Tsukamoto

1	tsukamoto = tsukamoto / jmlPredikat;
2	if (jmlPredikat == 0) {
3	tsukamoto = 0;
4	}
5	// System.out.println("tsu " + tsukamoto);
6	return tsukamoto;
7	}

5.2 Implementasi User Interface

User Interface dalam penelitian ini terdapat 3 halaman dalam aplikasi peramalan permintaan barang menggunakan Algoritma Genetika dan Fuzzy Tsukamoto. Pada halaman pertama menjelaskan tentang inputan data permintaan dan inputan aturan yang digunakan dalam Fuzzy Tsukamoto, kemudian pada halaman kedua menjelaskan tentang perhitungan peramalan permintaan barang menggunakan Algoritma Genetika dan Fuzzy Tsukamoto, dan pada halaman ketiga menjelaskan tentang hasil perbandingan antara data asli dan data perhitungan peramalan.

5.2.1 Halaman Input Data dan Aturan Fuzzy

Pada halaman ini merupakan halaman default ketika aplikasi dijalankan pertama kali. Terdapat *textfield* untuk memasukkan nilai data permintaan barang dan *textfield* untuk memasukkan aturan-aturan yang digunakan dalam setiap kriteia yang digunakan. Pada button Input merupakan proses untuk menyimpan data yang telah diinputkan ke dalam database yang telah digunakan. Berikut pada Gambar 5.1 merupakan halaman Input data dan aturan Fuzzy.



File Edit

Optimasi Fuzzy Tsukamoto dengan Algoritma Genetika

Studi Kasus Peramalan Permintaan Barang Semen

[Data Permintaan dan Aturan | Perhitungan dan Hasil | Perbandingan Hasil]

Input Data Tambah Permintaan Baru Minggu ke-26 <input type="text"/> <input type="button" value="Input"/> Tambah Aturan Fuzzy Tsukamoto Ke-31 T3 <input type="text"/> T2 <input type="text"/> T1 <input type="text"/> T <input type="text"/> <input type="button" value="Input"/> Keterangan T3 : 3 minggu sebelumnya T2 : 2 minggu sebelumnya T1 : 1 minggu sebelumnya T : minggu sekarang	Tabel Permintaan <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Bulan</th> <th>Minggu</th> <th>Permintaan</th> </tr> </thead> <tbody> <tr><td>Januari</td><td>1</td><td>13543</td></tr> <tr><td></td><td>2</td><td>11957</td></tr> <tr><td></td><td>3</td><td>15016</td></tr> <tr><td></td><td>4</td><td>9990</td></tr> <tr><td>Februari</td><td>5</td><td>8785</td></tr> <tr><td></td><td>6</td><td>8389</td></tr> <tr><td></td><td>7</td><td>11776</td></tr> <tr><td></td><td>8</td><td>12297</td></tr> <tr><td>Maret</td><td>9</td><td>11067</td></tr> <tr><td></td><td>10</td><td>11239</td></tr> <tr><td></td><td>11</td><td>9394</td></tr> <tr><td></td><td>12</td><td>9275</td></tr> <tr><td>April</td><td>13</td><td>11172</td></tr> <tr><td></td><td>14</td><td>10135</td></tr> <tr><td></td><td>15</td><td>11535</td></tr> <tr><td></td><td>16</td><td>8202</td></tr> <tr><td>Mei</td><td>17</td><td>10067</td></tr> <tr><td></td><td>18</td><td>5725</td></tr> <tr><td></td><td>19</td><td>8766</td></tr> <tr><td>Juni</td><td>20</td><td>11468</td></tr> <tr><td></td><td>21</td><td>9910</td></tr> <tr><td></td><td>22</td><td>10088</td></tr> <tr><td></td><td>23</td><td>5078</td></tr> <tr><td></td><td>24</td><td>8300</td></tr> <tr><td>Juli</td><td>25</td><td>5381</td></tr> </tbody> </table> Tabel Aturan <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>T3</th> <th>T2</th> <th>T1</th> <th>T</th> </tr> </thead> <tbody> <tr><td>1</td><td>T</td><td>R</td><td>S</td><td>T</td></tr> <tr><td>2</td><td>R</td><td>S</td><td>T</td><td>R</td></tr> <tr><td>3</td><td>S</td><td>T</td><td>R</td><td>S</td></tr> <tr><td>4</td><td>T</td><td>T</td><td>S</td><td>T</td></tr> <tr><td>5</td><td>T</td><td>S</td><td>T</td><td>T</td></tr> <tr><td>6</td><td>S</td><td>T</td><td>T</td><td>T</td></tr> <tr><td>7</td><td>S</td><td>S</td><td>T</td><td>S</td></tr> <tr><td>8</td><td>S</td><td>T</td><td>S</td><td>S</td></tr> <tr><td>9</td><td>T</td><td>S</td><td>S</td><td>S</td></tr> <tr><td>10</td><td>R</td><td>R</td><td>T</td><td>R</td></tr> <tr><td>11</td><td>R</td><td>T</td><td>R</td><td>R</td></tr> <tr><td>12</td><td>T</td><td>R</td><td>R</td><td>R</td></tr> <tr><td>13</td><td>R</td><td>R</td><td>S</td><td>R</td></tr> <tr><td>14</td><td>R</td><td>S</td><td>R</td><td>R</td></tr> <tr><td>15</td><td>S</td><td>R</td><td>R</td><td>R</td></tr> <tr><td>16</td><td>S</td><td>S</td><td>R</td><td>S</td></tr> <tr><td>17</td><td>S</td><td>R</td><td>S</td><td>S</td></tr> <tr><td>18</td><td>R</td><td>S</td><td>S</td><td>S</td></tr> <tr><td>19</td><td>T</td><td>T</td><td>R</td><td>T</td></tr> <tr><td>20</td><td>T</td><td>R</td><td>T</td><td>T</td></tr> <tr><td>21</td><td>R</td><td>T</td><td>T</td><td>T</td></tr> <tr><td>22</td><td>T</td><td>T</td><td>T</td><td>T</td></tr> <tr><td>23</td><td>R</td><td>R</td><td>R</td><td>R</td></tr> <tr><td>24</td><td>S</td><td>S</td><td>S</td><td>S</td></tr> <tr><td>25</td><td>S</td><td>R</td><td>T</td><td>T</td></tr> <tr><td>26</td><td>R</td><td>T</td><td>S</td><td>S</td></tr> <tr><td>27</td><td>T</td><td>S</td><td>R</td><td>R</td></tr> <tr><td>28</td><td>D</td><td>T</td><td>O</td><td>O</td></tr> </tbody> </table>	Bulan	Minggu	Permintaan	Januari	1	13543		2	11957		3	15016		4	9990	Februari	5	8785		6	8389		7	11776		8	12297	Maret	9	11067		10	11239		11	9394		12	9275	April	13	11172		14	10135		15	11535		16	8202	Mei	17	10067		18	5725		19	8766	Juni	20	11468		21	9910		22	10088		23	5078		24	8300	Juli	25	5381	No	T3	T2	T1	T	1	T	R	S	T	2	R	S	T	R	3	S	T	R	S	4	T	T	S	T	5	T	S	T	T	6	S	T	T	T	7	S	S	T	S	8	S	T	S	S	9	T	S	S	S	10	R	R	T	R	11	R	T	R	R	12	T	R	R	R	13	R	R	S	R	14	R	S	R	R	15	S	R	R	R	16	S	S	R	S	17	S	R	S	S	18	R	S	S	S	19	T	T	R	T	20	T	R	T	T	21	R	T	T	T	22	T	T	T	T	23	R	R	R	R	24	S	S	S	S	25	S	R	T	T	26	R	T	S	S	27	T	S	R	R	28	D	T	O	O	Keterangan R : Rendah S : Sedang T : Tinggi
Bulan	Minggu	Permintaan																																																																																																																																																																																																																															
Januari	1	13543																																																																																																																																																																																																																															
	2	11957																																																																																																																																																																																																																															
	3	15016																																																																																																																																																																																																																															
	4	9990																																																																																																																																																																																																																															
Februari	5	8785																																																																																																																																																																																																																															
	6	8389																																																																																																																																																																																																																															
	7	11776																																																																																																																																																																																																																															
	8	12297																																																																																																																																																																																																																															
Maret	9	11067																																																																																																																																																																																																																															
	10	11239																																																																																																																																																																																																																															
	11	9394																																																																																																																																																																																																																															
	12	9275																																																																																																																																																																																																																															
April	13	11172																																																																																																																																																																																																																															
	14	10135																																																																																																																																																																																																																															
	15	11535																																																																																																																																																																																																																															
	16	8202																																																																																																																																																																																																																															
Mei	17	10067																																																																																																																																																																																																																															
	18	5725																																																																																																																																																																																																																															
	19	8766																																																																																																																																																																																																																															
Juni	20	11468																																																																																																																																																																																																																															
	21	9910																																																																																																																																																																																																																															
	22	10088																																																																																																																																																																																																																															
	23	5078																																																																																																																																																																																																																															
	24	8300																																																																																																																																																																																																																															
Juli	25	5381																																																																																																																																																																																																																															
No	T3	T2	T1	T																																																																																																																																																																																																																													
1	T	R	S	T																																																																																																																																																																																																																													
2	R	S	T	R																																																																																																																																																																																																																													
3	S	T	R	S																																																																																																																																																																																																																													
4	T	T	S	T																																																																																																																																																																																																																													
5	T	S	T	T																																																																																																																																																																																																																													
6	S	T	T	T																																																																																																																																																																																																																													
7	S	S	T	S																																																																																																																																																																																																																													
8	S	T	S	S																																																																																																																																																																																																																													
9	T	S	S	S																																																																																																																																																																																																																													
10	R	R	T	R																																																																																																																																																																																																																													
11	R	T	R	R																																																																																																																																																																																																																													
12	T	R	R	R																																																																																																																																																																																																																													
13	R	R	S	R																																																																																																																																																																																																																													
14	R	S	R	R																																																																																																																																																																																																																													
15	S	R	R	R																																																																																																																																																																																																																													
16	S	S	R	S																																																																																																																																																																																																																													
17	S	R	S	S																																																																																																																																																																																																																													
18	R	S	S	S																																																																																																																																																																																																																													
19	T	T	R	T																																																																																																																																																																																																																													
20	T	R	T	T																																																																																																																																																																																																																													
21	R	T	T	T																																																																																																																																																																																																																													
22	T	T	T	T																																																																																																																																																																																																																													
23	R	R	R	R																																																																																																																																																																																																																													
24	S	S	S	S																																																																																																																																																																																																																													
25	S	R	T	T																																																																																																																																																																																																																													
26	R	T	S	S																																																																																																																																																																																																																													
27	T	S	R	R																																																																																																																																																																																																																													
28	D	T	O	O																																																																																																																																																																																																																													

Gambar 5.1 Implementasi Halaman Input Data dan Aturan Fuzzy

5.2.2 Halaman Input Data dan Aturan Fuzzy

Pada halaman Perhitungan dan Hasil terdapat 2 tampilan. Pada tampilan pertama merupakan populasi awal generasi pertama yang terdiri dari kolom nama individu, kolom A1 hingga D4, dan nilai fitness. Kemudian pada tampilan kedua merupakan hasil perhitungan peramalan permintaan barang dengan nilai fitness yang terbesar. Berikut merupakan Gambar 5.2 Implementasi halaman perhitungan dan hasil dalam peramalan permintaan barang.

Optimasi Fuzzy Tsukamoto dengan Algoritma Genetika

Studi Kasus Peramalan Permintaan Barang Semen

[Data Permintaan dan Aturan | Perhitungan dan Hasil | Perbandingan Hasil]

Parameter Algoritma Genetika popSize <input type="text" value="15"/> Mutation Rate <input type="text" value="0.4"/> Crossover Rate <input type="text" value="0.6"/> Iterasi <input type="text" value="5"/>	Fuzzy Jumlah Minggu <input type="text" value="25"/>	Populasi Iterasi Terakhir : <input type="button" value="Hitung"/>																																																																																																																																																																																																																																								
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Parent</th> <th>A1</th> <th>A2</th> <th>A3</th> <th>A4</th> <th>B1</th> <th>B2</th> <th>B3</th> <th>B4</th> <th>C1</th> <th>C2</th> <th>C3</th> <th>C4</th> <th>D1</th> <th>D2</th> <th>D3</th> <th>D4</th> <th>Fitness</th> </tr> </thead> <tbody> <tr><td>P1</td><td>7499.40</td><td>8603.50</td><td>11365....</td><td>10589....</td><td>8907.30</td><td>9028.10</td><td>10547....</td><td>11956....</td><td>3617.90</td><td>7864.60</td><td>7404.60</td><td>10712....</td><td>9860.40</td><td>9656.10</td><td>9553.40</td><td>10481....</td><td>5.0311...</td></tr> <tr><td>P2</td><td>10662....</td><td>11559....</td><td>13978....</td><td>15901....</td><td>4675.70</td><td>8154.40</td><td>8409.70</td><td>11450....</td><td>9118.80</td><td>8493.90</td><td>12005....</td><td>12206....</td><td>8569.90</td><td>9341.60</td><td>10077....</td><td>12660....</td><td>5.0311...</td></tr> <tr><td>P3</td><td>12235....</td><td>7113.20</td><td>8493.90</td><td>8176.60</td><td>9017.30</td><td>12851....</td><td>13049....</td><td>12746....</td><td>9964.90</td><td>13405....</td><td>12444....</td><td>3310.90</td><td>9397.50</td><td>9207.70</td><td>10065....</td><td>9025.10</td><td>4.8230...</td></tr> <tr><td>P4</td><td>8888.30</td><td>12987....</td><td>9225.70</td><td>11810....</td><td>9367.20</td><td>10013....</td><td>9845.70</td><td>9607.40</td><td>4468.80</td><td>4912.30</td><td>11274....</td><td>9862.20</td><td>9981.50</td><td>10015....</td><td>10180....</td><td>12188....</td><td>4.8165...</td></tr> <tr><td>P5</td><td>5381....</td><td>8202....</td><td>9910....</td><td>9990....</td><td>8389....</td><td>11535....</td><td>11772....</td><td>12297....</td><td>8766....</td><td>11067....</td><td>11776....</td><td>11957....</td><td>9275....</td><td>9394....</td><td>10067....</td><td>10135....</td><td>4.8157...</td></tr> <tr><td>P6</td><td>3310.90</td><td>7113.20</td><td>8493.90</td><td>8176.60</td><td>9017.30</td><td>12851....</td><td>13049....</td><td>12746....</td><td>9964.90</td><td>13405....</td><td>12444....</td><td>12235....</td><td>9397.50</td><td>9207.70</td><td>10065....</td><td>9025.10</td><td>4.7048...</td></tr> <tr><td>P7</td><td>11499....</td><td>11857....</td><td>14552....</td><td>16383....</td><td>4653.70</td><td>7389.70</td><td>7909.20</td><td>11292....</td><td>7849.40</td><td>7385.60</td><td>10997....</td><td>10131....</td><td>8662.50</td><td>9341.30</td><td>9975.00</td><td>12951....</td><td>4.7048...</td></tr> <tr><td>P8</td><td>8766....</td><td>10067....</td><td>8300....</td><td>11772....</td><td>5078....</td><td>8202....</td><td>8389....</td><td>8785....</td><td>10088....</td><td>9275....</td><td>9999....</td><td>11239....</td><td>10135....</td><td>13543....</td><td>11468....</td><td>12297....</td><td>4.6571...</td></tr> <tr><td>P9</td><td>8918.90</td><td>13717....</td><td>9457.10</td><td>11820....</td><td>10439....</td><td>9959.90</td><td>9813.00</td><td>3064.00</td><td>3821.60</td><td>11595....</td><td>9518.00</td><td>9943.10</td><td>9133.00</td><td>9858.40</td><td>12160....</td><td>4.6511...</td></tr> <tr><td>P10</td><td>9766....</td><td>10067....</td><td>10088....</td><td>11772....</td><td>6078....</td><td>9202....</td><td>9380....</td><td>9795....</td><td>9200....</td><td>9275....</td><td>9000....</td><td>12642....</td><td>10126....</td><td>11468....</td><td>12297....</td><td>4.6508...</td></tr> </tbody> </table> Hasil Rekomendasi : <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Parent</th> <th>A1</th> <th>A2</th> <th>A3</th> <th>A4</th> <th>B1</th> <th>B2</th> <th>B3</th> <th>B4</th> <th>C1</th> <th>C2</th> <th>C3</th> <th>C4</th> <th>D1</th> <th>D2</th> <th>D3</th> <th>D4</th> <th>Fitness</th> </tr> </thead> <tbody> <tr><td>P1</td><td>7499.40</td><td>8603.50</td><td>11365....</td><td>10589....</td><td>8907.30</td><td>9028.10</td><td>10547....</td><td>11956....</td><td>3617.90</td><td>7864.60</td><td>7404.60</td><td>10712.00</td><td>9860.40</td><td>9656.10</td><td>9553.40</td><td>10481.40</td><td>5.0311...</td></tr> </tbody> </table>			Parent	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4	Fitness	P1	7499.40	8603.50	11365....	10589....	8907.30	9028.10	10547....	11956....	3617.90	7864.60	7404.60	10712....	9860.40	9656.10	9553.40	10481....	5.0311...	P2	10662....	11559....	13978....	15901....	4675.70	8154.40	8409.70	11450....	9118.80	8493.90	12005....	12206....	8569.90	9341.60	10077....	12660....	5.0311...	P3	12235....	7113.20	8493.90	8176.60	9017.30	12851....	13049....	12746....	9964.90	13405....	12444....	3310.90	9397.50	9207.70	10065....	9025.10	4.8230...	P4	8888.30	12987....	9225.70	11810....	9367.20	10013....	9845.70	9607.40	4468.80	4912.30	11274....	9862.20	9981.50	10015....	10180....	12188....	4.8165...	P5	5381....	8202....	9910....	9990....	8389....	11535....	11772....	12297....	8766....	11067....	11776....	11957....	9275....	9394....	10067....	10135....	4.8157...	P6	3310.90	7113.20	8493.90	8176.60	9017.30	12851....	13049....	12746....	9964.90	13405....	12444....	12235....	9397.50	9207.70	10065....	9025.10	4.7048...	P7	11499....	11857....	14552....	16383....	4653.70	7389.70	7909.20	11292....	7849.40	7385.60	10997....	10131....	8662.50	9341.30	9975.00	12951....	4.7048...	P8	8766....	10067....	8300....	11772....	5078....	8202....	8389....	8785....	10088....	9275....	9999....	11239....	10135....	13543....	11468....	12297....	4.6571...	P9	8918.90	13717....	9457.10	11820....	10439....	9959.90	9813.00	3064.00	3821.60	11595....	9518.00	9943.10	9133.00	9858.40	12160....	4.6511...	P10	9766....	10067....	10088....	11772....	6078....	9202....	9380....	9795....	9200....	9275....	9000....	12642....	10126....	11468....	12297....	4.6508...	Parent	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4	Fitness	P1	7499.40	8603.50	11365....	10589....	8907.30	9028.10	10547....	11956....	3617.90	7864.60	7404.60	10712.00	9860.40	9656.10	9553.40	10481.40	5.0311...
Parent	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4	Fitness																																																																																																																																																																																																																									
P1	7499.40	8603.50	11365....	10589....	8907.30	9028.10	10547....	11956....	3617.90	7864.60	7404.60	10712....	9860.40	9656.10	9553.40	10481....	5.0311...																																																																																																																																																																																																																									
P2	10662....	11559....	13978....	15901....	4675.70	8154.40	8409.70	11450....	9118.80	8493.90	12005....	12206....	8569.90	9341.60	10077....	12660....	5.0311...																																																																																																																																																																																																																									
P3	12235....	7113.20	8493.90	8176.60	9017.30	12851....	13049....	12746....	9964.90	13405....	12444....	3310.90	9397.50	9207.70	10065....	9025.10	4.8230...																																																																																																																																																																																																																									
P4	8888.30	12987....	9225.70	11810....	9367.20	10013....	9845.70	9607.40	4468.80	4912.30	11274....	9862.20	9981.50	10015....	10180....	12188....	4.8165...																																																																																																																																																																																																																									
P5	5381....	8202....	9910....	9990....	8389....	11535....	11772....	12297....	8766....	11067....	11776....	11957....	9275....	9394....	10067....	10135....	4.8157...																																																																																																																																																																																																																									
P6	3310.90	7113.20	8493.90	8176.60	9017.30	12851....	13049....	12746....	9964.90	13405....	12444....	12235....	9397.50	9207.70	10065....	9025.10	4.7048...																																																																																																																																																																																																																									
P7	11499....	11857....	14552....	16383....	4653.70	7389.70	7909.20	11292....	7849.40	7385.60	10997....	10131....	8662.50	9341.30	9975.00	12951....	4.7048...																																																																																																																																																																																																																									
P8	8766....	10067....	8300....	11772....	5078....	8202....	8389....	8785....	10088....	9275....	9999....	11239....	10135....	13543....	11468....	12297....	4.6571...																																																																																																																																																																																																																									
P9	8918.90	13717....	9457.10	11820....	10439....	9959.90	9813.00	3064.00	3821.60	11595....	9518.00	9943.10	9133.00	9858.40	12160....	4.6511...																																																																																																																																																																																																																										
P10	9766....	10067....	10088....	11772....	6078....	9202....	9380....	9795....	9200....	9275....	9000....	12642....	10126....	11468....	12297....	4.6508...																																																																																																																																																																																																																										
Parent	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4	Fitness																																																																																																																																																																																																																									
P1	7499.40	8603.50	11365....	10589....	8907.30	9028.10	10547....	11956....	3617.90	7864.60	7404.60	10712.00	9860.40	9656.10	9553.40	10481.40	5.0311...																																																																																																																																																																																																																									

Gambar 5.2 Implementasi Halaman Perhitungan dan Hasil

5.2.3 Halaman Perbandingan Hasil

Pada halaman perbandingan Hasil merupakan tampilan untuk perbandingan antara data asli atau data yang digunakan dengan data hasil perhitungan

peramalan permintaan barang. Dalam halaman ini menampilkan 5 kolom, yaitu kolom minggu, kolom permintaan, kolom Tsukamoto, kolom Error dan kolom Error². Pada kolom Error merupakan kolom hasil pengurangan antara nilai hasil perhitungan peramalan permintaan dengan data asli atau data yang digunakan. Kemudian pada kolom Error² merupakan kolom hasil kuadrat dari hasil kolom error, dimana hasil dari kolom Error² tersebut akan digunakan untuk menghitung nilai RMSE atau nilai error yang terdapat dalam hasil perhitungan peramalan permintaan barang. Berikut pada Gambar 5.3 merupakan hasil Implementasi halaman perbandingan hasil.

Optimasi Fuzzy Tsukamoto dengan Algoritma Genetika				
Studi Kasus Peramalan Permintaan Barang Semen				
Data Permintaan dan Aturan				
Perbandingan dan Hasil				
Minggu	Permintaan	Tsukamoto	Error	Error ²
1	13543	-	-	-
2	11957	-	-	-
3	15015	-	-	-
4	9990	1364.600000000004	1862133.15000001	25472.1600000016
5	8785	159.600000000036	5103834.02777775	914031.6024999986
6	8389	-956.049999999993	9327526.81000002	3143617.650624999
7	11776	2259.166666666666	3054.100000000004	6830904.960000002
8	12297	-	-	-
9	11067	1773.024999999995	590745.9600000005	184792.51562499843
10	11239	2613.600000000004	2914873.2900000038	4942.09000000154
11	9394	768.600000000004	-429.8749999999982	3349357.5156250065
12	9275	-429.8749999999982	1707.30000000001	179267.55999999968
13	11772	-	-	-
14	10135	70.3000000000109	8412320.159999998	362.125000000018
15	11535	1830.125000000018	-1247.624999999982	1556568.1406249956
16	8202	-423.3999999999964	1556568.1406249956	3807051.3611111087
17	10067	362.125000000018	-2900.399999999996	-
18	5725	-2900.399999999996	-	-
19	8766	-1247.624999999982	-	-
20	11468	1951.166666666666	-	-
Total : 8.691481755874999E7				
Total/22 : 3950673.525397727				
RMSE : 1987.6301279155855				
Fitness : 5.0311171377176365				

Gambar 5.3 Implementasi Halaman Perbandingan Hasil

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan menjelaskan tentang pengujian dan analisis system peramalan permintaan barang menggunakan Algoritma Genetika dan Fuzzy Tsukamoto. Pengujian dilakukan dibuat sesuai dengan perancangan pengujian yang telah dibuat.

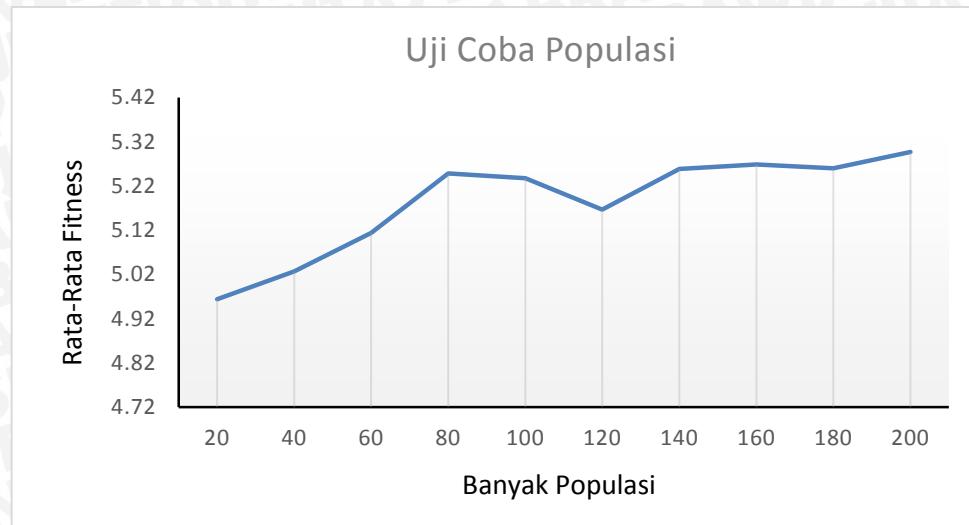
6.1 Hasil dan Analisa Uji Coba Ukuran Populasi

Uji coba yang pertama dilakukan adalah uji coba ukuran populasi terhadap perubahan nilai *fitness*. Dalam pengujian ini jumlah generasi yang digunakan adalah 5 dengan *crossover rate* sebesar 0,6 dan *mutation rate* sebesar 0,4. Jumlah data yang digunakan adalah 25 data permintaan barang permimpungnya pada bulan Januari sampai dengan bulan Juni. Pada setiap ukuran populasi dilakukan sebanyak 10 kali percobaan sehingga memperoleh nilai rata-rata *fitness*. Ukuran populasi diuji pada kelipatan 20, mulai dari 20 sampai dengan 200, sehingga terdapat 10 ukuran *popSize* yang berbeda. Berikut pada Tabel 6.1 merupakan hasil uji coba ukuran populasi.

Tabel 6.1 Hasil Pengujian Ukuran Populasi

No	Banyak Populasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>	
		Percobaan Populasi ke-											
		1	2	3	4	5	6	7	8	9	10		
1	20	5.18301	4.86492	4.85179	4.91966	5.01511	5.10982	4.73845	5.00469	5.00391	4.95303	4.964439175	
2	40	5.12708	5.08753	5.22317	4.93852	5.03083	4.89784	5.03856	5.01877	4.92699	4.98445	5.027373312	
3	60	5.02967	5.14341	5.13754	5.18305	5.23159	5.04622	4.95756	5.14786	5.19702	5.06267	5.113658211	
4	80	5.20531	5.37505	5.36772	5.2499	5.26935	5.25674	5.05793	5.20375	5.36253	5.13456	5.248283825	
5	100	5.29812	5.17674	5.17863	5.2051	5.406	5.58031	4.94951	5.23783	5.08565	5.26023	5.23781078	
6	120	5.08984	5.07364	5.22358	5.18332	5.17286	5.08771	5.2741	5.06956	5.23508	5.25872	5.166840673	
7	140	5.37902	5.11645	5.22055	5.29524	5.22559	5.18539	5.2506	5.01441	5.66196	5.23373	5.258296013	
8	160	5.22694	5.21488	5.26537	5.28185	5.22666	5.55746	5.14065	5.23228	5.32838	5.21187	5.268634213	
9	180	5.27211	5.16763	5.18223	5.22797	5.38102	5.26442	5.26008	5.28585	5.35148	5.2063	5.259910164	
10	200	5.14791	5.59569	5.39338	5.27817	5.31627	5.40098	5.35319	5.29465	5.06959	5.1232	5.297303676	

Hasil uji coba ukuran populasi pada Tabel 6.1 akan diformulasikan ke dalam bentuk grafik, dengan tujuan dapat mengetahui titik jumlah populasi optimal. Berikut pada Gambar 6.1 merupakan hasil uji coba ukuran populasi yang telah diformulasikan ke dalam gambar grafik.



Gambar 6.1 Grafik Hasil Uji Coba Ukuran Populasi

Pada Gambar 5.3 Grafik hasil uji coba ukuran populasi, dapat terlihat perbedaan hasil pengujian ukuran populasi terhadap nilai *fitness* yang dihasilkan. Dari hasil tersebut didapatkan bahwa ukuran populasi 80 merupakan hasil ukuran populasi yang dianggap optimal dengan rata-rata nilai *fitness* 5.248283825, sebab setelah jumlah populasi 80 sulit untuk mendapatkan nilai *fitness* yang lebih baik.

Besar nilai *fitness* dalam penelitian ini sangat dipengaruhi oleh besarnya ukuran populasi. Semakin besar ukuran populasi maka semakin besar juga nilai *fitness* yang dihasilkan. Akan tetapi jika ukuran populasi terlalu besar, kemungkinan akan sulit dalam memperoleh hasil nilai yang optimal dan terlalu besar ukuran populasi yang digunakan maka waktu dalam memproses perhitungan akan semakin lama.

Kondisi seperti ini disebut juga dengan kondisi konvergensi, artinya hampir semua kromosom bernilai sama sehingga akan menghasilkan *offspring* yang hampir sama dengan induknya yang dapat menyebabkan eksplorasi dalam memberikan solusi kurang baik (Mahmudy, 2013). Pola perubahan *fitness* semacam ini juga didapatkan oleh Putri, FB, Mahmudy, WF & Ratnawati, DE (2014) yang menerapkan algoritma genetika untuk *Vehicle Routing Problem with Time Window* (VRPTW) pada studi kasus *optimasi distribusi beras bersubsidi*. Selain itu, hal serupa juga didapatkan oleh Pitaloka, DA, Mahmudy, WF & Sutrisno (2014) pada studi kasus *Vehicle Routing Problem with Time Window* (VRPTW) dengan menerapkan *hybrid algoritma genetika*.

6.2 Hasil dan Analisa Uji Coba Banyaknya Generasi

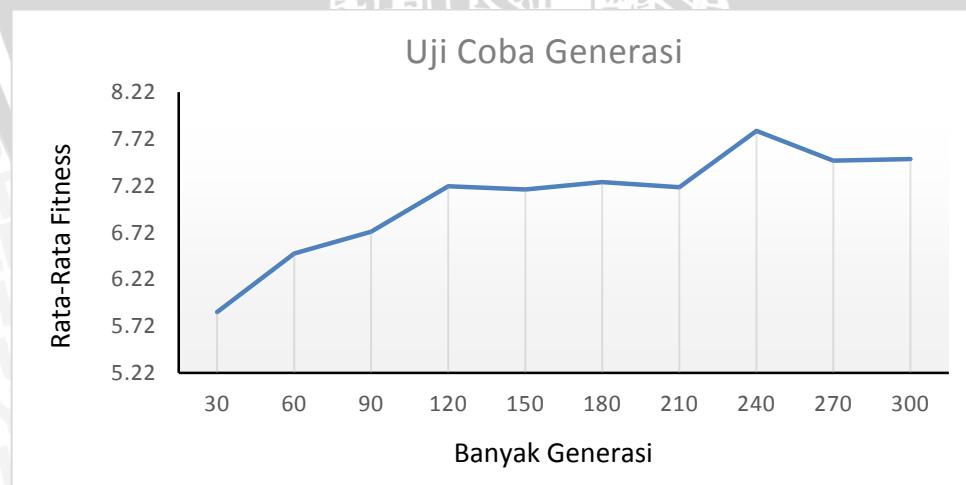
Uji coba yang kedua adalah uji coba jumlah generasi terhadap perubahan nilai *fitness*. Setelah melakukan pengujian terhadap ukuran populasi maka hasil yang didapatkan akan digunakan dalam pengujian jumlah generasi. Dari hasil pengujian jumlah generasi yang dilakukan sesuai dengan perancangan didapatkan hasil pada Tabel 6.2, dengan jumlah generasi sebanyak 50 sampai dengan 500 generasi.

Pada setiap generasi akan dilakukan pengujian nilai *fitness* sebanyak 10 kali dan didapatkan jumlah rata-rata nilai *fitness*nya. Kemudian nilai rata-rata *fitness* tersebut akan ditampilkan dalam bentuk grafik untuk mengetahui jumlah generasi yang optimal dengan ukuran populasi optimal sebesar 80, nilai crossover rate 0.6, nilai mutation rate 0.4, jumlah generasi sebanyak 5 dan jumlah data yang digunakan sebesar 25.

Tabel 6.2 Hasil Pengujian Jumlah Generasi

No	Banyak Generasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>	
		Percobaan Populasi ke-											
		1	2	3	4	5	6	7	8	9	10		
1	30	5.65349	5.7221	6.09613	6.13251	5.48126	5.96978	5.94187	6.03543	5.84208	5.8413	5.871595022	
2	60	6.51579	6.64577	6.33534	6.74148	6.3854	6.50814	6.28356	6.35544	6.72884	6.46999	6.496976279	
3	90	6.76716	7.47707	6.74791	7.12901	6.46482	6.45968	6.5913	6.3912	6.45549	6.82127	6.730491123	
4	120	7.19085	6.87125	6.89017	6.90968	7.09979	7.15789	7.24297	8.11472	6.3848	8.27506	7.213717573	
5	150	7.16348	8.21376	7.84184	7.31553	7.04621	7.37508	6.69924	7.43486	6.55819	6.17131	7.181948977	
6	180	7.25343	7.12301	7.66873	8.0376	6.91037	7.00915	8.07118	6.86524	6.61913	7.03822	7.259606626	
7	210	7.07058	7.22289	6.41003	7.29791	6.65095	7.18166	7.09086	8.58716	7.24957	7.27642	7.203800882	
8	240	7.53007	7.57044	7.70559	7.98827	7.1401	7.5922	8.87059	8.65067	6.82095	8.20667	7.807553885	
9	270	7.29267	7.35676	7.66354	7.57039	7.43009	7.42818	7.42328	8.05463	6.97047	7.71314	7.490314336	
10	300	7.45752	8.40244	7.41343	6.97934	8.10658	6.92522	7.25024	7.48337	7.68321	7.3681	7.506945985	

Hasil uji coba ukuran populasi pada Tabel 6.2 akan diformulasikan ke dalam bentuk grafik, dengan tujuan dapat mengetahui titik jumlah populasi optimal. Berikut pada Gambar 6.2 merupakan hasil uji coba jumlah generasi yang telah diformulasikan ke dalam bentuk grafik.



Gambar 6.2 Grafik Hasil Uji Coba Jumlah Generasi

Berdasarkan Gambar 6.2 jumlah generasi yang optimal didapatkan pada jumlah generasi 120 dengan nilai rata-rata fitness 7.213717573. Dari pengujian tersebut dapat disimpulkan bahwa jumlah generasi dapat mempengaruhi nilai

fitness yang dihasilkan dalam proses perhitungan Algoritma Genetika. Jumlah generasi yang semakin banyak belum tentu memberikan hasil yang optimal, sebab semakin banyak jumlah generasi maka semakin besar peluang terjadinya konvergensi dan waktu yang dibutuhkan dalam proses perhitungan tersebut akan menjadi lama. Jumlah generasi 120 merupakan solusi yang optimal dalam uji coba jumlah generasi, sebab jumlah generasi setelah 120 sangat sulit untuk menentukan nilai fitness yang lebih besar lagi. Hal tersebut terjadi karena adanya kondisi konvergensi setelah jumlah generasi 120.

6.3 Hasil dan Analisa Uji Coba Kombinasi Cr dan Mr

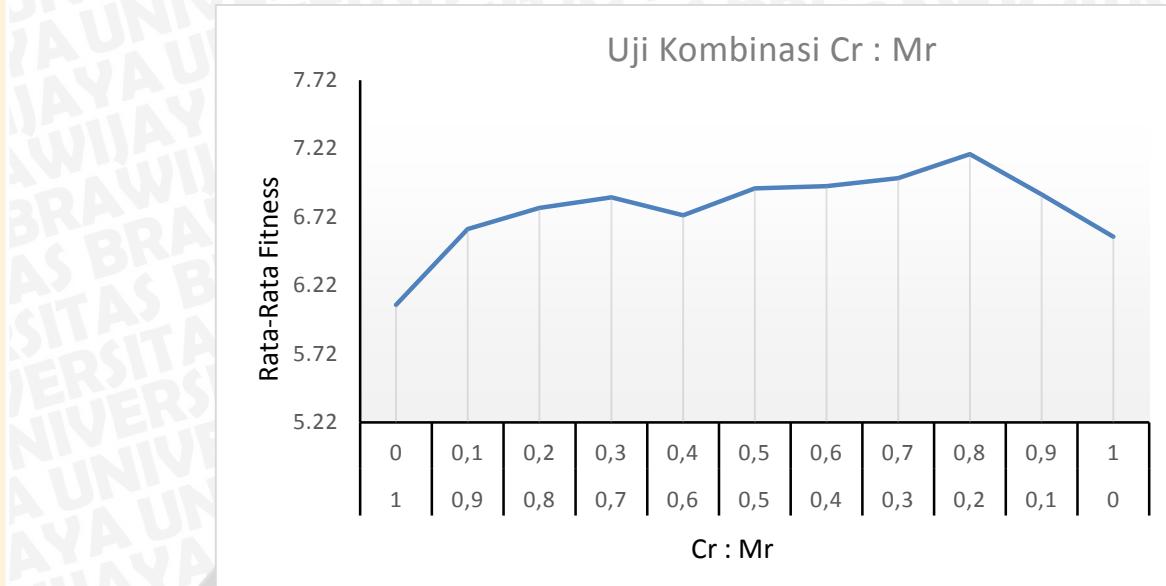
Setelah melakukan pengujian ukuran populasi dan jumlah generasi, maka pengujian selanjutnya adalah pengujian untuk mendapatkan nilai optimal dari kombinasi Cr dan Mr. berikut pada Tabel 6.3 merupakan hasil pengujian kombinasi Cr dan Mr untuk mendapatkan nilai optimal dalam kombinasi tersebut.

Dalam uji coba kombinasi Cr dan Mr jumlah populasi yang digunakan sebesar 80, jumlah generasi sebanyak 120 dan jumlah data yang digunakan sebesar 25 data.

Tabel 6.3 Hasil Pengujian Kombinasi Cr dan Mr

Kombinasi		Nilai Fitness										Rata-rata fitness
		Percobaan Kombinasi Cr dan Mr ke-										
Cr	Mr	1	2	3	4	5	6	7	8	9	10	
1	0	6.12824	6.05512	6.74122	5.80137	6.14466	6.38646	6.10699	5.70723	5.6716	6.03144	6.077433606
0,9	0,1	6.80023	7.08882	6.60014	7.56948	6.08577	6.42741	6.13244	6.68416	6.06181	6.86286	6.63131167
0,8	0,2	7.11589	6.02853	6.06	6.44497	6.43867	6.22811	6.4468	8.28952	6.934	7.89111	6.787760068
0,7	0,3	6.71944	6.12111	6.98459	6.62423	8.02682	5.77505	6.97824	6.85082	7.79231	6.76274	6.863533684
0,6	0,4	6.29683	6.79525	6.88043	6.956	7.41962	7.04894	6.50687	6.64846	6.42461	6.35909	6.733609401
0,5	0,5	7.08565	6.64537	7.09765	7.01817	7.00316	7.33971	6.42588	7.03397	7.07778	6.57352	6.930084516
0,4	0,6	6.64847	6.90655	6.98235	6.69521	7.6126	7.00813	6.8324	6.95819	7.53549	6.26711	6.944650465
0,3	0,7	7.88852	6.84575	7.16659	6.86275	6.53063	6.74149	7.08044	6.7604	7.10434	7.06005	7.004097727
0,2	0,8	7.01262	7.9664	6.87998	6.91746	7.31606	7.39758	7.02246	6.64687	7.25385	7.39373	7.180701153
0,1	0,9	7.01826	6.86671	6.82148	6.81697	7.04282	6.82058	7.22666	6.60637	7.27882	6.35725	6.885593383
0	1	6.56878	7.16782	6.20611	6.34926	6.53804	6.50613	6.73169	6.58909	6.48055	6.62372	6.5761184

Hasil uji coba kombinasi Cr dan Mr pada Tabel 6.3 akan diformulasikan ke dalam bentuk grafik, dengan tujuan dapat mengetahui titik kombinasi Cr dan Mr yang optimal. Berikut pada Gambar 6.3 merupakan hasil uji coba kombinasi Cr dan Mr yang telah diformulasikan ke dalam bentuk grafik.



Gambar 6.3 Grafik Hasil Uji Coba Kombinasi Cr dan Mr

Pada Gambar 6.3 dapat dilihat bahwa nilai *fitness* dengan rata-rata terbaik ada pada kombinasi Crossover sebesar 0.3 dan mutation sebesar 0.7 dengan nilai rata-rata *fitness* sebesar 6.863533684. Pada nilai kombinasi crossover dengan jumlah 0 dan mutation sebesar 1 merupakan nilai rata-rata fitness yang terkecil dengan nilai rata-rata *fitness* sebesar 6.5761184. Dengan nilai crossover rate yang tinggi dan mutation rate dengan nilai yang rendah, maka algoritma genetika akan mengalami penurunan kemampuan dalam menjaga diversitas populasi sehingga mengalami konvergensi dini hanya dalam beberapa generasi serta kehilangan untuk mengeksplorasi area lain dalam ruang pencarian (Mahmudy, 2013). Akan tetapi bila nilai crossover rate yang terlalu rendah dan nilai mutation rate yang lebih tinggi maka algoritma genetika akan sangat bergantung pada proses mutasi (Mahmudy, WF, Marian, RM & Loung, LHS, 2013). Kesimpulan yang didapat dalam uji coba kombinasi crossover rate dan mutation rate adalah nilai rata-rata *fitness* yang tertinggi ada pada nilai 0.3 : 0.7.

6.4 Hasil dan Analisa Uji Coba Menggunakan Parameter Terbaik

Setelah melakukan pengujian ukuran populasi, jumlah generasi, kombinasi Cr dan Mr dan jumlah data yang digunakan, maka pengujian selanjutnya adalah uji coba menggunakan parameter terbaik. Berikut pada Tabel 6.4 merupakan hasil uji coba menggunakan parameter terbaik untuk melihat hasil yang didapatkan dari hasil pengujian sebelumnya dalam mendapatkan nilai optimal.

Dalam uji coba menggunakan parameter terbaik menggunakan kombinasi Cr dan Mr yang sebesar 0.3 : 0.7, jumlah populasi yang digunakan sebesar 80, jumlah generasi sebanyak 120, dan jumlah data yang digunakan sebesar 25.

Tabel 6.4 Hasil Pengujian Menggunakan Parameter Terbaik

Percobaan ke-	Nilai <i>fitness</i>
1	7.191586222
2	6.960106285
3	7.107569854
4	7.5834942
5	7.109786478
6	7.460262386
7	6.897202871
8	6.98818738
9	7.245711885
10	7.13895194
Rata-Rata Fitness	7.168286

Pencarian nilai rata-rata *fitness* yang dilakukan pada Tabel 6.4 dilakukan sebanyak 10 kali, sehingga hasil dari uji coba pada pengujian dengan menggunakan nilai parameter yang terbaik didapatkan pada nilai rata-rata sebesar 7.168286.

Pada tabel 6.5 berikut ini merupakan hasil perhitungan metode Fuzzy Tsukamoto dan Algoritma Genetika dengan menggunakan kromosom terbaik dalam menentukan hasil nilai error pada studi kasus peramalan permintaan.

Tabel 6.5 Contoh Hasil Perhitungan Menggunakan Kromosom Terbaik

Minggu	Permintaan	Tsukamoto & Algoritma Genetika	Error (Data Permintaan-Hasil Prediksi)	Error (Data Permintaan-Hasil Prediksi) ²
1	13543	-	-	-
2	11957	-	-	-
3	15016	-	-	-
4	9990	10957.45	-967.45	935959.5025
5	8785	7098.9	1686.1	2842933.21
6	8389	7098.9	1290.1	1664358.01
7	11776	10437.16667	1338.833333	1792474.694
8	12297	11104.82	1192.18	1421293.152
9	11067	12243.63333	-1176.633333	1384466.001
10	11239	10957.45	281.55	79270.4025
11	9394	9028.175	365.825	133827.9306
12	9275	9671.266667	-396.2666667	157027.2711
13	11772	10437.16667	1334.833333	1781780.028
14	10135	11104.82	-969.82	940550.8324
15	11535	10957.45	577.55	333564.0025
16	8202	9028.175	-826.175	682565.1306
17	10067	9671.266667	395.7333333	156604.8711

Minggu	Permintaan	Tsukamoto & Algoritma Genetika	Error (Data Permintaan-Hasil Prediksi)	Error (Data Permintaan-Hasil Prediksi) ²
18	5725	7098.9	-1373.9	1887601.21
19	8766	9396.6	-630.6	397656.36
20	11468	10437.16667	1030.833333	1062617.361
21	9910	11104.82	-1194.82	1427594.832
22	10088	9671.266667	416.733333	173666.6711
23	5078	7098.9	-2020.9	4084036.81
24	8300	9396.6	-1096.6	1202531.56
25	5381	9396.6	-4015.6	1.61E+07
Total				40667423.2

Kemudian untuk menghitung Nilai *Error* dari keseluruhan total error yang didapatkan pada tabel 6.5 dengan menggunakan parameter terbaik, maka diperhitungkan dengan menggunakan RMSE yang telah dijelaskan pada bab sebelumnya. Berikut merupakan persamaan perhitungan dengan RMSE.

$$RMSE = \sqrt{\frac{40667423.2}{22}} = 1848519.237$$

Untuk perhitungan fitness pada kromosom yang telah dikonversikan, dapat dilihat pada persamaan perhitungan berikut ini.

$$\begin{aligned} \textit{fitness} &= \frac{10000}{RMSE} \\ \textit{fitness} &= \frac{10000}{1848519.237} = 7.35509 \end{aligned}$$

Dari hasil perhitungan *fitness* tersebut didapatkan hasil dengan nilai *fitness* sebesar 7.35509. maka dapat disimpulkan bahwa hasil yang didapatkan dalam studi kasus permalan permintaan dengan menggunakan metode Fuzzy Tsukamoto dan Algoritma Genetika mampu memberikan nilai error yang lebih kecil dengan menggunakan parameter terbaik.



BAB 7 PENUTUP

7.1 Kesimpulan

Kesimpulan yang didapatkan dari pengujian yang telah dilakukan pada peramalan permintaan barang semen dengan menggunakan Algoritma Genetika dan Fuzzy Tsukamoto adalah sebagai berikut :

1. Dari hasil pengujian yang telah dilakukan dalam penelitian ini, metode Algoritma Genetika dan Fuzzy Tsukamoto dapat memberikan nilai error yang lebih rendah dalam studi kasus peramalan permintaan barang. Pada Implementasi dalam peramalan permintaan barang Semen menggunakan Algoritma Genetika dan Fuzzy Tsukamoto, solusi pada ukuran populasi terbaik adalah 80 popsize, solusi pada jumlah generasi terbaik adalah 120, untuk solusi kombinasi nilai *crossover rate* (cr) dan *mutation rate* (mr) didapatkan 0.3 : 0.7.
2. Dalam penelitian ini, metode Algoritma Genetika mampu menghasilkan nilai yang optimal dalam hal memprediksi atau meramalkan suatu permintaan barang semen. Cara penyelesaian dalam Algoritma Genetika pada penelitian ini menggunakan beberapa metode seperti *Extended Intermediate Crossover*, *Reciprocal Exchange Mutation* dan *Elitism Selection*. Hasil dari penelitian ini pada peramalan menggunakan Algoritma Genetika dan Fuzzy Tsukamoto mendapatkan nilai error yang lebih rendah dibandingkan dengan hasil peramalan perhitungan Fuzzy Tsukamoto.
3. Dengan menggunakan metode Algoritma Genetika di dalam perhitungan Fuzzy Tsukamoto pada studi kasus peramalan permintaan barang mampu menghasilkan nilai yang optimal dibandingkan dengan melakukan perhitungan dengan metode Fuzzy Tsukamoto, sebab batasan-batasan yang digunakan dalam menentukan nilai Tinggi, Sedang, Rendah pada Fuzzy Tsukamoto dilakukan secara acak yang didapatkan dari nilai data permintaan minggu sebelumnya yang kemudian nilai yang didapatkan akan di urutkan berdasarkan dari nilai terkecil hingga terbesar pada setiap kriteria yang digunakan.

7.2 Saran

Berdasarkan kesimpulan yang telah dibuat, maka saran yang dapat diberikan untuk pengembangan dalam penelitian selanjutnya adalah sebagai berikut :

1. Pada penelitian selanjutnya diharapkan menggunakan metode Fuzzy selain Fuzzy Tsukamoto untuk dapat digabungkan dengan metode Algoritma Genetika dalam studi kasus peramalan atau prediksi permintaan barang.
2. Dalam penelitian selanjutnya diharapkan mampu mendapatkan jumlah data yang lebih banyak lagi dengan tujuan mendapatkan variasi pengujian

yang dilakukan agar mendapatkan nilai hasil yang lebih optimal, sebab pada penelitian ini masih terbatas dalam mendapatkan data permintaan barang semen.



DAFTAR PUSTAKA

- Tobing, RL. 2010. *Sistem Simulasi Penjadwalan Kuliah Dengan Menggunakan Algoritma Genetika*. Universitas Sumatera Utara, Medan.
- Mahmudy, WF. 2013. *Algoritma Evolusi*. Program Teknologi Informatika dan Ilmu Komputer, Universitas Brawijaya, Malang.
- Mahmudah, Ainy, Irhamah & Mumpuni, Sri. 2011. *Optimasi Persediaan Multi-Item Fuzzy EOQ Dengan Algoritma Genetika*. FMIPA, Institut Teknologi Sepuluh Nopember, Surabaya.
- Palupi, YD Ayu, dkk. 2011. *Algoritma Genetika Untuk Optimasi Persediaan Barang Dalam Proses Produksi*. FMIPA, Institut Teknologi Sepuluh Nopember, Surabaya.
- Indroprasto, Suryani, Erma, 2012. *Analisis Pengendalian Produk Dengan Metode EOQ Menggunakan Algoritma Genetika untuk Mengefisiensikan Biaya Persediaan*. Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Surabaya.
- Gunawan & Rouf, 2013. *Purwarupa Sistem Kendali Kecepatan Mobil Berdasarkan Jarak dengan Sistem Inferensi Fuzzy Tsukamoto*. PP. 117-126.
- Bon & Utami, 2014. *An Analytical Hierarchy Process and Fuzzy Inference System Tsukamoto for Production Planning : a Review and Conceptual Research*. The Business & Management Review, Vol. 5.
- Ramuna, MDT & Mahmudy, WF 2015. *Optimasi Persediaan Barang dalam Produksi Jilbab Menggunakan Algoritma Genetika*. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, Vol. 5, No.14.
- Mahmudy, WF, Marian, RM & Loung, LHS 2013. *Optimization of Part Type Selection and Loading Problem with Alternative Production Plans in Flexible Manufacturing System Using Hybrid Genetic Algorithms – Part 2 : Genetic Operators & Results*. 5th International Conference on Knowledge and Smart Technology (KST), Chonburi, Thailand, 31 Jan – 1 Feb, pp. 81-85.
- Putri, FB, Mahmudy, WF & Ratnawati, DE. 2014. *Penerapan Algoritma Genetika Untuk Vehicle Routing Problem With Time Windows (VRPTW) pada Kasus Optimasi Distribusi Beras Bersubsidi*. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, Vol. 5, No.1.
- Nasution, Hakim, A, & Prasetyawan, Y. 2008. *Perencanaan & Pengendalian Produksi*. Graha Ilmu, Yogyakarta.
- Sari, AP, Mahmudy, WF & Dewi, C. 2014. *Optimasi Asupan Gizi pada Ibu Hamil dengan Menggunakan Algoritma Genetika*. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, Vol. 4, No.5.
- Panharesi, YG & Mahmudy, WF. 2014. *Optimasi Distribusi Barang dengan Algoritma Genetika*. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, Vol. 5, No.11.



- Indroprasto, Suryani, E. 2012. *Analisis Pengendalian Persediaan dengan Metode EOQ Menggunakan Algoritma genetika untuk Mengefisiensikan Biaya Persediaan*. Jurnal Teknik, Institut Teknologi Sepuluh Nopember (ITS).
- Rangkuti, F. 2005. *Analisis SWOT : Teknik Membedah Kasus Bisnis*. Jakarta : PT. Gramedia.
- Sumayang, L. 2003. *Manajemen Produksi dan Operasi*. Jakarta : Salemba Empat.
- Ciptayani, PI, Mahmudy, WF & Widodo, AW. 2009. *Penerapan Algoritma Genetika untuk Kompresi Citra Fraktal*. Jurnal Ilmu Komputer, Vol. 2, No.1, April, pp. 1-9.
- Bon, TA, Utami, FS. 2014. *An Analytical Hierarchy Process and Fuzzy Inference System Tsukamoto for Production Planning : a Review and Conceptual Research*. Department of Production and Operations Management, Faculty of Technology Management and Business, Universiti Tun Hussein Onn Malaysia.

