



TUGAS AKHIR - TE141599

MO-GA untuk Optimasi *Multi-Objective* dalam
Distribusi Pasien Rujukan Berjenjang BPJS di Surabaya

Aulia Vica Rachman Hernanda
NRP 2212 100 189

Dosen Pembimbing
Prof. Ir. Abdullah Alkaff, M.Sc., Ph.D.
Nurlita Gamayanti, ST., MT.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



FINAL PROJECT - TE141599

MO-GA for Multi-Objective Optimization on BPJS Referral Patient Distribution in Surabaya

Aulia Vica Rachman Hernanda
NRP 2212 100 189

Supervisor
Prof. Ir. Abdullah Alkaff, M.Sc., Ph.D.
Nurlita Gamayanti, ST., MT.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2016

**MO-GA UNTUK OPTIMASI MULTI-OBJECTIVE DALAM
DISTRIBUSI PASIEN RUJUKAN BERJENJANG BPJS
DI SURABAYA**

TUGAS AKHIR

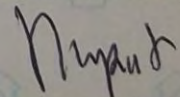
**Diajukan Guna Memenuhi Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui

Dosen Pembimbing I,

Dosen Pembimbing II,


Prof. Ir. Abdullah Alkaff, M.Sc., Ph.D.
NIP. 19550123 198003 1 002


Nurlita Gamayanti, ST., MT.
NIP. 19781201 200212 2 002



MO-GA untuk Optimasi *Multi-Objective* dalam Distribusi Pasien Rujukan Berjengjang BPJS di Surabaya

Nama : Aulia Vica Rachman Hernanda
Pembimbing : Prof. Ir. Abdullah Alkaff, M.Sc., Ph.D.
Nurlita Gamayanti, ST., MT.

ABSTRAK

Seiring dengan berjalannya program pemerintah mengenai jaminan kesehatan untuk seluruh warga Indonesia. Timbul berbagai masalah pada semua *stakeholder*-nya seperti puskesmas/dokter keluarga/klinik, rumah sakit, dan pasien. Masalah tersebut berupa penuhnya antrian pasien di Rumah sakit, sehingga mengakibatkan nilai utilisasi dari fasilitas pelayanan menjadi buruk dan tentunya meningkatkan waktu tunggu pasien. Namun disisi lain pada waktu yang bersamaan ketika antrian di satu rumah sakit sangat penuh, di rumah sakit yang lain tidak terjadi penumpukan pasien. Maka dari itu dalam tugas akhir ini dilakukan penelitian tentang pengaturan distribusi pasien di Kota Surabaya. Pengaturan distribusi ini bertujuan untuk meminimalkan rata-rata waktu tunggu dan memaksimalkan rata-rata utilisasi. Optimasi distribusi pasien ini akan menggunakan *Multi-objective Genetic Algorithm* untuk menemukan proporsi rujukan ke setiap Rumah sakit. Proporsi rujukan tersebut menjadi input untuk simulasi sistem di program ARENA. Dengan menerapkan MO-GA ini akan menghasilkan himpunan solusi *trade-off* yang kemudian akan dipilih dua solusi untuk diuji menggunakan program ARENA. Rata-rata waktu tunggu pada solusi optimal 1 yaitu 0,0514 jam dan hasil dari solusi optimal 2 bernilai 0,0561 jam. Namun untuk nilai rata-rata utilisasi lebih baik nilainya pada solusi optimal 2 yaitu 0,2938 dibandingkan hasil dari solusi optimal 1 yang bernilai 0,2915.

Kata Kunci: MO-GA, Optimasi, Sistem antrian, Solusi *trade-off*, Utilisasi, Waktu tunggu

--halaman ini sengaja dikosongkan--

MO-GA for Multi-Objective Optimization on BPJS Referral Patient Distribution in Surabaya

Name : Aulia Vica Rachman Hernanda
Advisor : Prof. Ir. Abdullah Alkaff, M.Sc., Ph.D.
Nurlita Gamayanti, ST., MT.

ABSTRACT

With the passage of the government's program on health assurance to all citizens of Indonesia. Arise various problems in all of its stakeholders such as community health centers / family doctors / clinics, hospitals, and patients. The problem in the form of full queue of patients in hospitals, resulting in the value of the utilization of service facilities for the worse and certainly increase patient waiting times. On the other hand at the same time when the queue at the hospital is very full, at another hospital is no long queues of patients. Therefore in this thesis research on managing the distribution of patients in the city of Surabaya. This distribution arrangement aims to minimize the average waiting time and maximize the average utilization. Optimization of the distribution of these patients will use the Multi-objective Genetic Algorithm to find the proportion of referrals to each hospital. The proportion of the referral becomes the input to the simulation system ARENA program. By applying MO-GA will produce the solution trade-off which will then have two solutions to be tested using the ARENA program. The average waiting time in the optimal solution 1 is 0.0514 hour and the results of the optimal solution 2 0.0561 hours. However, better utilization at optimal solution 2 is 0.2938 compared to the results of the optimal solution 1 whose value is 0.2915.

Key Word: MO-GA, Optimization, Trade-off solutions, Utilization, Queuing time, Queue system

--halaman ini sengaja dikosongkan--

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN JUDUL.....	iii
PERNYATAAN KEASLIAN TUGAS AKHIR.....	v
HALAMAN PENGESAHAN	vii
ABSTRAK	ix
<i>ABSTRACT</i>	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL.....	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan	2
1.5 Metodologi	3
1.6 Sistematika Penulisan	4
1.7 Relevansi.....	4
BAB II TEORI PENUNJANG	5
2.1 Sistem Pelayanan Kesehatan	5
2.1.1 Jaminan Kesehatan Nasional.....	5
2.1.2 Badan Penyelenggara Jaminan Sosial.....	5
2.2 Sistem Rujukan Berjenjang	6
2.3 Sistem Antrian.....	8
2.3.1 Komponen dan Notasi dalam Sistem Antrian	9
2.3.2 Sistem Antrian Single Server	11
2.3.3 Sistem Antrian Multi Server	12
2.3.4 Jaringan Antrian	13
2.3.5 Karakteristik Sistem Antrian.....	15
2.4 Optimasi Multi-Tujuan	16
2.4.1 Solusi Trade-Off.....	16
2.4.2 Optimalitas Pareto dan Dominansi	17
2.5 <i>Genetic Algorithm</i>	20
2.6 <i>Multi-objective Genetic Algorithm</i>	24
BAB III PERANCANGAN SISTEM	29
3.1 Identifikasi Masalah.....	29
3.2 Deskripsi Sistem Optimisasi Rujukan Pasien	31

3.3	Perancangan Jaringan Antrian Rujukan Pasien.....	32
3.4	Perancangan Simulasi dengan Arena	34
3.5	Perancangan Model Optimisasi	38
3.5.1	Fungsi Objektif 1.....	38
3.5.2	Fungsi Objektif 2.....	39
3.5.3	Konstrain Pertidaksamaan 1.....	39
3.5.4	Konstrain Persamaan 2	40
3.6	Perancangan MO-GA.....	40
BAB IV IMPLEMENTASI DAN ANALISIS.....		45
4.1	Pengumpulan dan Pengolahan Data.....	45
4.2	Simulasi Kondisi Awal	49
4.3	Penerapan MO-GA	52
4.3.1	Pengujian MO-GA.....	52
4.3.2	Penentuan variabel keputusan	60
4.3	Simulasi dengan Solusi Optimal 1	61
4.4	Simulasi dengan Solusi Optimal 2	66
4.5	Perbandingan Hasil	71
BAB V PENUTUPAN		75
5.1	Kesimpulan.....	75
5.2	Saran	75
DAFTAR PUSTAKA		77
LAMPIRAN A.....		79
LAMPIRAN B.....		87
LAMPIRAN C.....		97
LAMPIRAN D.....		109
RIWAYAT HIDUP.....		111

DAFTAR TABEL

Tabel 3.1 Contoh ranking Pareto MO-GA	43
Tabel 4.1 Rate Kedatangan (jam) dari puskesmas dan wilayah	46
Tabel 4.2 Proporsi penyakit setiap Puskesmas (%).....	47
Tabel 4.3 Agregasi rate kedatangan (per jam) setiap penyakit	48
Tabel 4.4 Rate pelayanan (per jam) dan jumlah dokter tiap poli	48
Tabel 4.5 Proporsi rujukan kondisi awal	49
Tabel 4.6 Perbandingan hasil pengujian.....	58
Tabel 4.7 Hasil ranking Pareto pada percobaan C	59
Tabel 4.8 Rate kedatangan solusi optimal 1	60
Tabel 4.9 Rate kedatangan solusi optimal 2	60
Tabel 4.10 Proporsi rujukan optimal 1	61
Tabel 4.11 Proporsi rujukan optimal 2	66

DAFTAR GAMBAR

Gambar 2.1 Alur rujukan.....	6
Gambar 2.2 Sistem Rujukan Berjenjang	8
Gambar 2.3 Komponen sistem antrian	9
Gambar 2.4 Proses Sistem Antrian	9
Gambar 2.5 <i>Single server</i>	11
Gambar 2.6 <i>Multi server</i> dengan 1 jalur antrian	12
Gambar 2.7 <i>Multi server</i> dengan banyak jalur antrian	12
Gambar 2.8 <i>Feed forward queue</i> [2].....	14
Gambar 2.9 Hubungan antara utilisasi dan waktu tunggu [3].....	15
Gambar 2.10 Agregasi rate kedatangan.....	15
Gambar 2.11 Partisi rate kedatangan.....	16
Gambar 2.12 Konsep dominansi.....	18
Gambar 2.13 Solusi visibel sesuai kombinasi fungsi tujuan [8]	20
Gambar 2.14 Susunan algoritma genetika [5].....	21
Gambar 2.15 <i>Stochastic Universal Selection</i> [7]	22
Gambar 2.16 Aritmatik <i>crossover</i>	23
Gambar 2.17 Diagram alir proses algoritma genetika	24
Gambar 2.18 Ranking Pareto pada MO-GA [8]	25
Gambar 2.19 Diagram alir MO-GA	28
Gambar 3.1 Grafik proporsi rujukan kondisi awal.....	31
Gambar 3.2 Alur pasien mendapat pelayanan kesehatan.....	32
Gambar 3.3 Alur rujukan pada sistem rujukan berjenjang BPJS	32
Gambar 3.4 Perancangan jaringan antrian rujukan pasien.....	33
Gambar 3.5 Modul <i>Create</i>	34
Gambar 3.6 Penentuan parameter modul <i>Create</i>	34
Gambar 3.7 Modul <i>decide</i>	35
Gambar 3.8 Menentukan jenis <i>decide N-way by chance</i>	35
Gambar 3.9 Modul <i>Process</i>	35
Gambar 3.10 Penentuan parameter modul <i>Process</i>	36
Gambar 3.11 Penentuan parameter <i>resource</i>	36
Gambar 3.12 Perancangan model simulasi untuk poli sejenis	37
Gambar 3.13 Perancangan individu algoritma genetika	40
Gambar 3.14 Perancangan metode seleksi dengan SUS.....	41
Gambar 3.15 Perancangan <i>crossover</i> menggunakan metode aritmatik	42
Gambar 3.16 Contoh ranking Pareto MO-GA	43
Gambar 4.1 Utilisasi setiap poli di setiap Rumah sakit kondisi awal	50

Gambar 4.2 Waktu tunggu setiap poli di setiap Rumah sakit kondisi awal	50
Gambar 4.3 Utilisasi kondisi awal setiap RS	51
Gambar 4.4 Waktu tunggu kondisi awal tiap RS	51
Gambar 4.5 Probabilitas <i>crossover</i> = 0,5 dan iterasi = 100	52
Gambar 4.6 Probabilitas <i>crossover</i> = 0,5 dan iterasi = 300	53
Gambar 4.7 Probabilitas <i>crossover</i> = 0,5 dan iterasi = 500	53
Gambar 4.8 Probabilitas <i>crossover</i> = 0,7 dan iterasi = 100	54
Gambar 4.9 Probabilitas <i>crossover</i> = 0,7 dan iterasi = 300	54
Gambar 4.10 Probabilitas <i>crossover</i> = 0,7 dan iterasi = 500	55
Gambar 4.11 Probabilitas <i>crossover</i> = 0,9 dan iterasi = 100	55
Gambar 4.12 Probabilitas <i>crossover</i> = 0,9 dan iterasi = 300	56
Gambar 4.13 Probabilitas <i>crossover</i> = 0,9 dan iterasi = 500	56
Gambar 4.14 Ranking Pareto	59
Gambar 4.15 Perbandingan utilisasi setiap poli (1)	62
Gambar 4.16 Perbandingan waktu tunggu setiap poli (1)	63
Gambar 4.17 Perbandingan utilisasi poli sejenis di semua RS (1)	64
Gambar 4.18 Perbandingan waktu tunggu poli sejenis di semua RS (1)	64
Gambar 4.19 Perbandingan utilisasi & waktu tunggu setiap RS (1)	65
Gambar 4.20 Perbandingan utilisasi & waktu tunggu keseluruhan (1)	65
Gambar 4.21 Perbandingan utilisasi setiap poli (2)	67
Gambar 4.22 Perbandingan waktu tunggu setiap poli (2)	68
Gambar 4.23 Perbandingan utilisasi poli sejenis di semua RS (2)	69
Gambar 4.24 Perbandingan waktu tunggu poli sejenis di semua RS (2)	69
Gambar 4.25 Perbandingan utilisasi & waktu tunggu setiap RS (2)	70
Gambar 4.26 Perbandingan utilisasi & waktu tunggu keseluruhan (2)	70
Gambar 4.27 Perbandingan utilisasi seluruh poli semua RS (3)	71
Gambar 4.28 Perbandingan waktu tunggu seluruh poli semua RS (3)	71
Gambar 4.29 Perbandingan utilisasi setiap RS (3)	72
Gambar 4.30 Perbandingan waktu tunggu setiap RS (3)	73
Gambar 4.31 Perbandingan utilisasi poli sejenis di semua RS (3)	74
Gambar 4.32 Perbandingan waktu tunggu poli sejenis di semua RS (3)	74

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di Kota Surabaya, terdapat 58 rumah sakit yang terdaftar pada Dinas Kesehatan yang memiliki fasilitas dan pelayanan yang berbeda-beda meskipun telah diatur tentang fasilitas dan pelayanan minimal yang terdapat dalam satu rumah sakit melalui Peraturan Menteri Kesehatan Republik Indonesia nomor 340 tahun 2010 tentang klasifikasi rumah sakit. Dari keseluruhan rumah sakit tersebut, terbagi menjadi rumah sakit umum yaitu, rumah sakit yang memberikan pelayanan kesehatan pada banyak bidang dan jenis penyakit, dan rumah sakit khusus yaitu rumah sakit yang memberikan pelayanan kesehatan pada satu bidang atau jenis penyakit.

Seiring dengan berjalannya program pemerintah mengenai jaminan kesehatan untuk seluruh warga Indonesia atau yang biasa disebut Jaminan Kesehatan Nasional dengan badan yang melaksanakan adalah BPJS. Sistem yang baru diterapkan ini menyisakan berbagai masalah terhadap semua *stakeholder*-nya seperti puskesmas/dokter keluarga/klinik, rumah sakit, dan pasien. Masalah tersebut berupa penuhnya antrian pasien pada puskesmas dan rumah sakit, sehingga mengakibatkan tingkat pelayanan pada pasien yang kurang memuaskan. Namun disisi lain pada waktu yang bersamaan ketika antrian di satu rumah sakit sangat penuh, di rumah sakit yang lain tidak terjadi penumpukan pasien. Sehingga tidak terjadi keseimbangan jumlah kedatangan pasien di tiap-tiap rumah sakit yang ada di Kota Surabaya.

Tidak seimbangnya pengembangan fasilitas kesehatan publik dan tingkat kedatangan pasien yang meningkat telah menimbulkan masalah. Pasien menginginkan layanan kesehatan yang cepat dan berkualitas tinggi, tetapi tidak semua rumah sakit dapat memberikan pelayanan yang efektif dan efisien bagi mereka. Rata-rata pasien lebih memilih rumah sakit dengan jarak terdekat dan berkualitas tinggi, yang membuat rumah sakit menghadapi dilema mengenai kedatangan pasien melebihi kapasitas pelayanan. Hal tersebut dianggap sebagai penghambat dalam proses pelayanan kesehatan. Dalam beberapa kondisi, hal ini dapat menyebabkan keterlambatan penanganan pasien oleh tim medis yang dapat berakibat fatal. Untuk mengatasi masalah penumpukan pasien di rumah sakit tertentu, perlu diatur distribusi pasien yang bertujuan untuk

mengoptimalkan kinerja sistem kesehatan perkotaan dengan menyesuaikan distribusi aliran pasien, yang ditinjau dari rata-rata waktu tunggu, rata-rata waktu dalam sistem dan utilisasi.

Pada penelitian tugas akhir ini akan dilakukan di 13 Puskesmas dan 3 Rumah sakit di area Surabaya. Yang difokuskan pada pembahasan rata-rata utilisasi dan rata-rata waktu tunggu pasien. Dalam penelitian tugas akhir ini digunakan MO-GA untuk menyelesaikan permasalahan *multiobjective* yaitu memaksimalkan rata-rata utilisasi dan meminimalkan rata-rata waktu tunggu.

1.2 Permasalahan

Dalam penelitian tugas akhir ini yang menjadi masalah utama adalah menumpuknya pasien di salah satu Rumah sakit tertentu, sehingga dapat menaikkan rata-rata waktu tunggu di rumah sakit tersebut. Serta menyebabkan rata-rata utilisasi menjadi buruk.

1.3 Batasan Masalah

Dalam penelitian tugas akhir ini yang menjadi batasan masalah adalah

- a. 13 puskesmas yang berada di area Surabaya Timur.
- b. 3 Rumah Sakit yaitu RS. Haji Surabaya, RS. Islam Jemursari, dan RS. Airlangga.
- c. Hanya memperhatikan rawat jalan.
- d. Kedatangan pasien diasumsikan distribusi Poisson.
- e. Pelayanan di Poli Rumah sakit diasumsikan distribusi Exponensial.
- f. Kedatangan pasien dari puskesmas lain akan tetap dipertimbangkan dan dihitung namun tidak spesifik, hanya dihitung berdasarkan wilayahnya.
- g. Menggunakan jaringan antrian *feed forward queueing*.

1.4 Tujuan

Penelitian tugas akhir ini bertujuan untuk menghasilkan perangkat lunak guna menghitung proporsi rujukan setiap penyakit ke poli penyakit yang sejenis pada tiga Rumah Sakit yang telah ditentukan tersebut. Hasil yang diharapkan dari menghitung proporsi rujukan adalah menemukan titik optimal dari dua fungsi tujuan yaitu, memaksimalkan rata-rata

utilisasi dan meminimalkan rata-rata waktu tunggu di setiap poli penyakit di Rumah Sakit.

1.5 Metodologi

Dalam penelitian Tugas Akhir ini diperlukan suatu tahapan yang merepresentasikan urutan yang harus dilaksanakan agar sesuai dengan tujuan penelitian. Tahapan tersebut ialah sebagai berikut:

a. Studi literatur

Tahap awal untuk membuat suatu penelitian ialah mengumpulkan literatur yang berkaitan dengan peraturan rujukan BPJS, teori antrian, teknik optimasi multi-tujuan, dan algoritma MO-GA. Literatur berasal dari jurnal, buku cetak, internet, dan lain-lain. Dengan adanya studi literatur, penelitian dapat dilakukan berdasarkan teori-teori yang telah ada sebelumnya.

b. Identifikasi

Tahap yang kedua ialah mendapatkan model matematika dari sistem antrian yang ada di setiap poli rumah sakit. Pemodelan ini dilakukan dengan cara melakukan pengamatan.

c. Optimasi Aliran Distribusi Pasien

Tahap yang ketiga ialah mengoptimalkan aliran distribusi pasien rujukan berjenjang BPJS. Tujuan yang akan dioptimalkan adalah rata-rata utilisasi dan rata-rata waktu tunggu. Untuk mendapatkan hasil yang optimal dari tujuan tersebut digunakan algoritma MO-GA.

d. Simulasi dan Evaluasi

Tahap yang keempat ialah mensimulasikan proporsi rujukan yang telah didapat pada dari algoritma MO-GA menggunakan *software* ARENA. Tujuan dari simulasi ialah untuk mengetahui performansi dari sistem yang telah dirubah proporsi rujukannya sesuai dengan hasil dari algoritma MO-GA. Setelah itu akan dievaluasi hasil dari simulasi tersebut apakah telah memenuhi kriteria yang diharapkan.

e. Penulisan buku Tugas Akhir

Tahap yang terakhir ialah penulisan laporan/buku Tugas Akhir. Penulisan dilakukan secara intensif bila proses pengujian telah selesai.

1.6 Sistematika Penulisan

Tahap terakhir dari sebuah penelitian adalah penulisan laporan. Pada penulisan laporan/buku Tugas Akhir ini disusun berdasarkan 5 bab, di mana setiap bab berisi mengenai permasalahan dalam penelitian. Bab tersebut ialah sebagai berikut:

BAB I PENDAHULUAN

Berisi mengenai latar belakang, permasalahan, pembatasan masalah, tujuan, metodologi, sistematika penulisan, dan relevansi pembahasan tugas akhir ini.

BAB II TEORI PENUNJANG

Berisi mengenai konsep dasar dan teori yang mendasari perancangan tugas akhir ini, meliputi peraturan rujukan BPJS, teori antrian, teknik optimasi multi-tujuan, dan algoritma MO-GA

BAB III PERANCANGAN SISTEM

Berisi mengenai spesifikasi sistem, identifikasi parameter, perancangan MO-GA dengan mengacu teori pada BAB II.

BAB IV IMPLEMENTASI DAN ANALISIS

Berisi prosedur pelaksanaan implementasi pengujian dan analisis data mengenai rata-rata utilisasi dan rata-rata waktu tunggu di setiap poli Rumah Sakit. Pengujian ini dilakukan dengan menggunakan simulasi program ARENA.

BAB V PENUTUP

Berisi mengenai kesimpulan dari penelitian Tugas Akhir dan saran untuk dapat digunakan untuk pengembangan tugas akhir ini untuk lebih lanjut.

1.7 Relevansi

Tugas akhir ini diharapkan dapat menjadi rekomendasi untuk 13 Puskesmas yang ada di wilayah Surabaya Timur yang akan merujuk pasiennya ke 3 Rumah Sakit, yaitu RS. Haji Surabaya, RS. Islam Jemursari, dan RS. Universitas Airlangga. Supaya dapat mengurangi jumlah penumpukan pasien di Rumah Sakit tertentu.

BAB II

TEORI PENUNJANG

2.1 Sistem Pelayanan Kesehatan

Kesehatan masyarakat merupakan salah satu tanggung jawab negara. Seiring dengan berkembangnya negara Indonesia, pemerintah menjamin kesehatan seluruh lapisan masyarakat. Sehingga pada tahun 2004 muncul program jaminan sosial yang disebut dengan Sistem Jaminan Sosial Nasional.

2.1.1 Jaminan Kesehatan Nasional

Jaminan Kesehatan Nasional (JKN) merupakan bagian dari Sistem Jaminan Sosial Nasional (SJSN) yang diselenggarakan dengan menggunakan mekanisme asuransi kesehatan sosial yang bersifat wajib berdasarkan Undang-Undang Nomor 40 Tahun 2004 tentang Sistem Jaminan Sosial Nasional (SJSN) dengan tujuan untuk memenuhi kebutuhan dasar kesehatan masyarakat yang layak yang diberikan kepada setiap orang yang telah membayar iuran atau iurannya dibayar oleh Pemerintah [1].

Dengan diselenggarakan program Jaminan Kesehatan Nasional ini pemerintah negara Indonesia berusaha untuk memberikan perlindungan dan kesejahteraan sosial bagi seluruh rakyat Indonesia. Dan hal yang paling utama adalah masyarakat mendapatkan jaminan kesehatan.

2.1.2 Badan Penyelenggara Jaminan Sosial

Badan Penyelenggara Jaminan Sosial (BPJS) adalah badan hukum publik yang berfungsi untuk menyelenggarakan program jaminan sosial bagi seluruh penduduk Indonesia. Berdasarkan UU No. 24 Tahun 2011, kini Badan Penyelenggara Jaminan Sosial terbagi menjadi dua lembaga besar, yaitu BPJS Kesehatan dan BPJS Ketenagakerjaan. BPJS Kesehatan dan BPJS Ketenagakerjaan berkedudukan dan berkantor di Ibu Kota Negara Republik Indonesia. BPJS dapat mempunyai kantor perwakilan di provinsi dan kantor cabang di kabupaten/kota.

BPJS Kesehatan adalah badan yang menyelenggarakan program Jaminan Kesehatan. Memberikan jaminan kesehatan yang meliputi rawat jalan dan rawat inap. BPJS Ketenagakerjaan adalah badan publik yang menyelenggarakan program jaminan kecelakaan kerja, jaminan hari tua, dan jaminan kematian. BPJS Kesehatan ini merupakan badan yang

ditunjuk untuk menggantikan PT Askes untuk menjalankan perannya, sedangkan BPJS Ketenagakerjaan merupakan badan yang ditunjuk untuk menggantikan peran PT Jamsostek.

Undang-Undang BPJS mewajibkan pembentukan BPJS Kesehatan dan BPJS Ketenagakerjaan dengan mentransformasikan penyelenggara saat ini, PT Askes dan PT Jamsostek, dari perusahaan Badan Usaha Milik Negara (BUMN) menjadi badan hukum yang bersifat publik dan nirlaba. Dengan adanya pembentukan badan baru yang mengatur dan menjamin kesehatan dan sosial dari masyarakat Indonesia, diharapkan dapat meningkatkan tingkat kesehatan dan pelayanan kesehatan oleh Puskesmas dan Rumah Sakit, serta memberikan pelayanan kesehatan yang terbaik bagi seluruh masyarakat dari seluruh lapisan ekonomi [1].

2.2 Sistem Rujukan Berjenjang

Sistem Rujukan pelayanan kesehatan adalah penyelenggaraan pelayanan kesehatan yang mengatur pelimpahan tugas dan tanggung jawab pelayanan kesehatan secara timbal balik baik vertikal maupun horizontal yang wajib dilaksanakan oleh peserta jaminan kesehatan atau asuransi kesehatan sosial, dan seluruh fasilitas kesehatan.



Gambar 2.1 Alur rujukan

Pelayanan kesehatan perorangan terdiri dari 3 (tiga) tingkatan yaitu:

- a. Pelayanan kesehatan tingkat pertama
Merupakan pelayanan kesehatan dasar yang diberikan oleh fasilitas kesehatan tingkat pertama.
- b. Pelayanan kesehatan tingkat kedua
Pelayanan kesehatan spesialisik yang dilakukan oleh dokter spesialis atau dokter gigi spesialis yang menggunakan pengetahuan dan teknologi kesehatan spesialisik.
- c. Pelayanan kesehatan tingkat ketiga
Merupakan pelayanan kesehatan sub spesialisik yang dilakukan oleh dokter sub spesialis atau dokter gigi sub spesialis yang menggunakan pengetahuan dan teknologi kesehatan sub spesialisik.

Seperti pada Gambar 2.1, dalam menjalankan pelayanan kesehatan, fasilitas kesehatan tingkat pertama dan tingkat lanjutan wajib melakukan sistem rujukan dengan mengacu pada peraturan perundangundangan yang berlaku. Peserta yang ingin mendapatkan pelayanan yang tidak sesuai dengan sistem rujukan dapat dimasukkan dalam kategori pelayanan yang tidak sesuai dengan prosedur sehingga tidak dapat dibayarkan oleh BPJS Kesehatan. Fasilitas Kesehatan yang tidak menerapkan sistem rujukan maka BPJS Kesehatan akan melakukan *recredentialing* terhadap kinerja fasilitas kesehatan tersebut dan dapat berdampak pada kelanjutan kerjasama. Pelayanan rujukan dapat dilakukan secara horizontal maupun vertikal. Sistem rujukan berjenjang dapat dilihat pada Gambar 2.2.

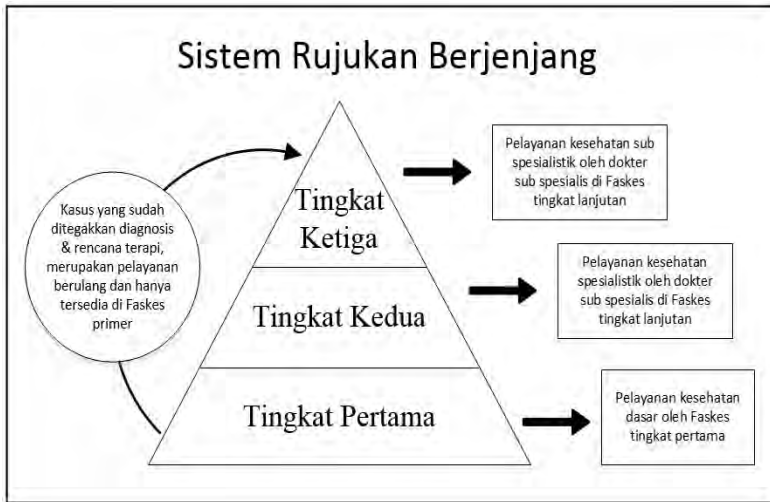
Rujukan horizontal adalah rujukan yang dilakukan antar pelayanan kesehatan dalam satu tingkatan apabila perujuk tidak dapat memberikan pelayanan kesehatan sesuai dengan kebutuhan pasien karena keterbatasan fasilitas, peralatan dan/atau ketenagaan yang sifatnya sementara atau menetap.

Rujukan vertikal adalah rujukan yang dilakukan antar pelayanan kesehatan yang berbeda tingkatan, dapat dilakukan dari tingkat pelayanan yang lebih rendah ke tingkat pelayanan yang lebih tinggi atau sebaliknya. Rujukan vertikal dari tingkatan pelayanan yang lebih rendah ke tingkatan pelayanan yang lebih tinggi dilakukan apabila [1] :

- a. Pasien membutuhkan pelayanan kesehatan spesialisik atau subspecialistik.
- b. Perujuk tidak dapat memberikan pelayanan kesehatan sesuai dengan kebutuhan pasien karena keterbatasan fasilitas.

Rujukan vertikal dari tingkatan pelayanan yang lebih tinggi ke tingkatan pelayanan yang lebih rendah dilakukan apabila [1] :

- a. Permasalahan kesehatan pasien dapat ditangani oleh tingkatan pelayanan kesehatan yang lebih rendah sesuai dengan kompetensi dan kewenangannya.
- b. Kompetensi dan kewenangan pelayanan tingkat pertama atau kedua lebih baik dalam menangani pasien tersebut.
- c. Pasien membutuhkan pelayanan lanjutan yang dapat ditangani oleh tingkatan pelayanan kesehatan yang lebih rendah dan untuk alasan kemudahan, efisiensi dan pelayanan jangka panjang.
- d. Perujuk tidak dapat memberikan pelayanan kesehatan sesuai dengan kebutuhan pasien karena keterbatasan sarana, prasarana, peralatan dan/atau ketenagaan.



Gambar 2.2 Sistem Rujukan Berjenjang

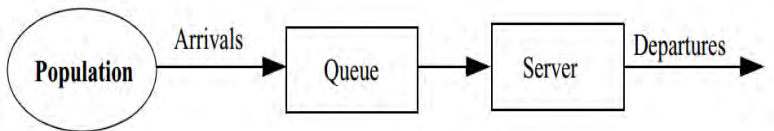
2.3 Sistem Antrian

Sistem antrian adalah suatu himpunan pelanggan, pelayan serta suatu aturan yang mengatur kedatangan pelanggan dan pemrosesan masalah pelayanan antrian dimana dicirikan oleh lima buah komponen yaitu: pola kedatangan, pola pelayanan, jumlah pelayan, kapasitas fasilitas untuk menampung para pelanggan dan aturan dalam mana para pelanggan dilayani. Dalam sistem antrian juga dikenal dengan populasi

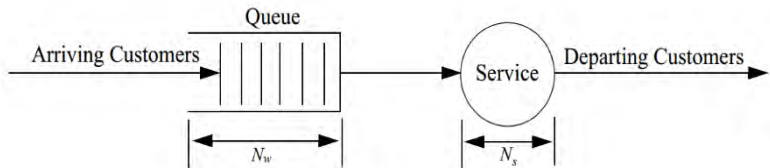
pelanggan yang datang, populasi ini dapat berupa kedatangan tak terhingga atau *infinite* dan kedatangan yang terbatas. Pada umumnya pelanggan dalam sistem antrian dilayani menurut aturan *First Come First Serve* (FCFS). Pelanggan tersebut akan dilayani oleh salah satu dari pelayanan yang tersedia, apabila semua server pelayanan sibuk maka pelanggan harus antri sampai ada salah satu server pelayanan yang sedang tersedia atau sedang tidak melayani pelanggan.

Peforma dari Sistem antrian dapat digambarkan dengan rata-rata jumlah kedatangan dalam antrian, rata-rata waktu tunggu dari suatu kedatangan dan persentase waktu luang dari pelayanan. Ukuran peforma ini dapat digunakan untuk memutuskan jumlah pelayanan yang harus diberikan, perubahan yang harus dilakukan dalam kecepatan pelayanan atau perubahan lain dalam sistem antrian. Dengan sasaran pelayanan, jumlah pelayan dapat ditentukan tanpa berpatokan pada biaya waktu tunggu [2].

2.3.1 Komponen dan Notasi dalam Sistem Antrian



Gambar 2.3 Komponen sistem antrian



Gambar 2.4 Proses Sistem Antrian

Gambar 2.3 dan 2.4 merupakan komponen dan proses sistem antrian. Dalam sistem antrian tersebut memiliki notasi-notasi yang digunakan untuk merepresentasikan variabel random. Berikut notasi yang digunakan pada sistem antrian:

- a. N_w adalah jumlah pelanggan yang menunggu untuk dilayani.
- b. N_s adalah jumlah pelanggan yang sedang dilayani.
- c. N adalah total jumlah pelanggan dalam sistem antrian tersebut.
- d. T_w adalah waktu tunggu pelanggan untuk mendapatkan layanan.
- e. T_s adalah waktu pelayanan.
- f. T adalah total waktu pelanggan dalam sistem antrian tersebut.

Population adalah jumlah pelanggan yang akan mendatangi fasilitas pelayanan dapat berupa *finite* atau *infinite*. *Arriving pattern* merupakan waktu antar kedatangan yang biasa direpresentasikan dengan distribusi *poisson* atau menyesuaikan. *Service time distribution* adalah variabel yang menentukan berapa waktu yang dibutuhkan untuk melayani pelanggan, dapat bersifat konstan maupun mengikuti distribusi tertentu. *Utilisasi* adalah peluang server sibuk melayani pelanggan, *utilisasi* merupakan probabilitas yang tentunya tidak boleh lebih dari satu. Apabila *utilisasi* bernilai lebih dari satu, dapat dikatakan sistem antrian tersebut buruk.

Hal lain yang perlu diperhatikan adalah *Capacity of the queueing facility* adalah batas kapasitas antrian untuk pelanggan, dapat bersifat *finite* atau *infinite*. Apabila sistem antrian tersebut bersifat *finite* maka pelanggan yang datang saat sistem antrian sedang sangat sibuk akan ditolak. Dengan kata lain sistem berkapasitas *finite* adalah cara untuk membatasi pelanggan yang datang. *Number of service* merupakan banyaknya server yang dapat melayani pelanggan, yang dapat lebih dari satu. Penjelasan lebih lanjut terkait banyaknya jumlah server dapat dilihat pada sub-bab berikutnya.

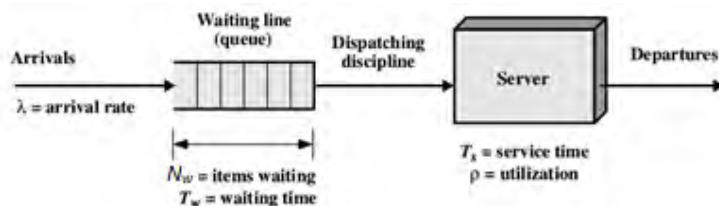
Pada proses sistem antrian juga dikenal istilah *Queueing discipline* atau yang biasa juga disebut dengan *Service discipline*. Yang menentukan aturan mengenai pelanggan mana yang akan dilayani dalam proses pelayanan berikutnya. Aturan-aturan tersebut dapat mengikuti aturan seperti dibawah ini [2]:

- a. FCFS, *first come first serve*, yang berarti pelanggan yang datang pertama akan mendapatkan pelayanan lebih dulu.
- b. LCSF, *last come first serve*, yang berarti pelanggan yang datang lebih akhir akan dilayani lebih dulu daripada pelanggan yang datang lebih awal.
- c. SIRO, *service in random order*, yang berarti pelayanan untuk pelanggan selanjutnya dilakukan secara acak dengan mengikuti probabilitas tertentu.

- d. *Priority*, yang berarti proses pemilihan pelanggan yang akan dilayani akan didasarkan pada kebutuhan pelanggan tersebut. Prioritas yang akan menjadi dasar utama untuk memilih pelanggan selanjutnya yang akan dilayani.

2.3.2 Sistem Antrian *Single Server*

Sistem antrian server tunggal adalah sebuah antrian sederhana dengan hanya ada satu server yang melayani pelanggan dan hanya ada satu jalur untuk memasuki sistem antrian tersebut. Sehingga analisis dari sistem antrian server tunggal menjadi mudah, dikarenakan hanya memiliki satu server dan satu jalur. Sistem antrian server tunggal dapat diilustrasikan pada Gambar 2.5.



Gambar 2.5 *Single server*

Peforma sistem antrian yang memiliki server tunggal seperti utilisasi server dan rata-rata lama waktu tunggu dapat dihitung dengan formula dibawah [3].

$$\rho = \frac{\lambda}{\mu} \quad (2.1)$$

$$T_w = \frac{\lambda}{\mu(\mu - \lambda)} \quad (2.2)$$

Keterangan:

ρ = Utilisasi

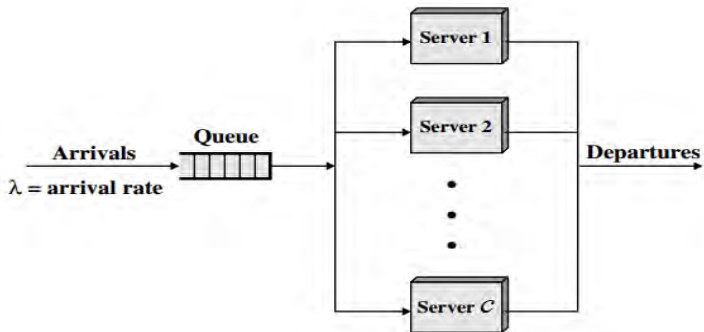
λ = Rate kedatangan

μ = Rate pelayanan

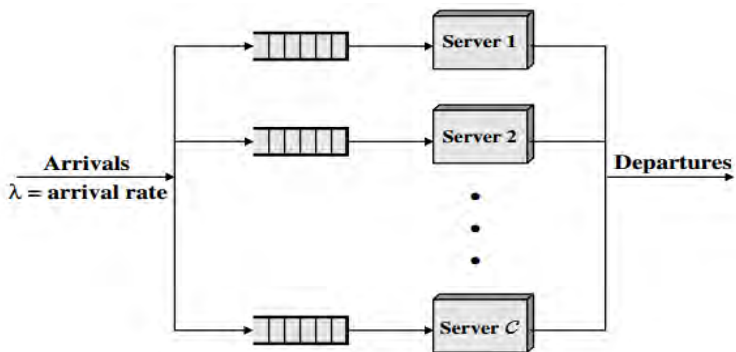
T_w = Rata-rata waktu tunggu

2.3.3 Sistem Antrian *Multi Server*

Sistem antrian banyak server adalah sebuah sistem antrian yang memiliki lebih dari satu pelayanan. Berbeda halnya dengan sistem antrian yang memiliki server tunggal. Sebagian besar antrian merupakan sistem antrian dengan memiliki banyak server. Contohnya adalah pelayanan di teller bank yang memiliki banyak server untuk melayani pelanggan. Dalam sistem antrian *multi server* juga terdapat istilah *resource* yaitu sesuatu yang melakukan pelayanan terhadap pelanggan atau barang. *Resource* dapat berupa orang atau mesin, yang dimana pembagian atau aturan kerjanya dapat ditentukan sesuai kebutuhan [3].



Gambar 2.6 *Multi server* dengan 1 jalur antrian



Gambar 2.7 *Multi server* dengan banyak jalur antrian

Sistem antrian *multi server* memiliki dua tipe seperti gambar diatas. Gambar 2.6 menunjukkan sistem antrian *multi server* yang memiliki satu jalur antrian saja sedangkan Gambar 2.7 menunjukkan sistem antrian *multi server* yang memiliki jalur antrian di setiap servernya. Peforma sistem antrian yang memiliki server lebih dari satu seperti utilisasi server dan rata-rata lama waktu tunggu dapat dihitung dengan formula dibawah [3].

$$\rho = \frac{\lambda}{\mu} \quad (2.3)$$

$$P_0 = \frac{c!(1-\rho)}{(\rho c)^c + c!(1-\rho) \left\{ \sum_{n=0}^{c-1} \frac{1}{n!} (\rho c)^c \right\}} \quad (2.4)$$

$$T_w = \frac{(\rho c)^c}{c!(1-\rho)^2 c \mu} P_0 \quad (2.5)$$

Keterangan:

C = Jumlah server

P_0 = Probabilitas tidak adanya pelanggan dalam sistem

2.3.4 Jaringan Antrian

Banyak sistem yang kompleks dapat dimodelkan oleh jaringan antrian dimana pelanggan menerima layanan dari satu atau lebih server, dimana setiap server memiliki antrian. Jaringan ini dimodelkan dengan *directed graph* yang dimana *node* mewakili sistem antrian dan *arc* antara *node* memiliki bobot yang memiliki probabilitas routing, di mana routing probabilitas dari *node* A ke *node* B adalah probabilitas bahwa seorang pelanggan yang meninggalkan antrian diwakili oleh *node* A berikutnya akan pergi ke antrian yang diwakili oleh *node* B dengan label probabilitas pada *arc* (A, B). Sebuah jaringan antrian dapat berupa antrian terbuka ataupun tertutup.

Jaringan antrian juga disebut sebagai *Markovian network of queues* atau *Markovian queueing network*. Yang juga dapat direpresenasikan dengan notasi M/M/c. Antrian dapat dikatakan sebagai *Markovian network of queues* jika antrian tersebut memiliki karakteristik sebagai berikut. Pertama, memiliki waktu pelayanan yang berbeda di

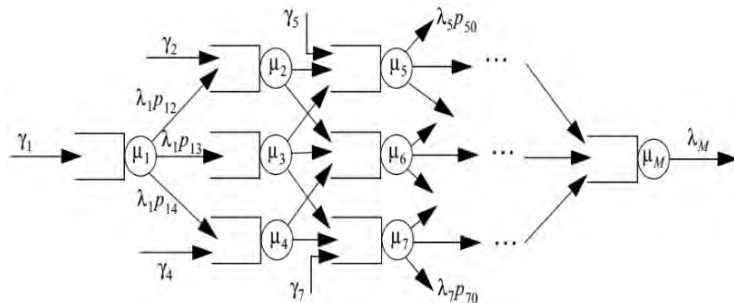
antrian yang berbeda. Kedua, jika antrian tersebut merupakan antrian terbuka, rate kedatangan dari eksternal sistem mengikuti distribusi poisson. Ketiga, transisi antara *node* yang berurutan bersifat independen. Untuk menganalisis *Markovian network of queues*, ada sebuah teori yang penting untuk diperhatikan yaitu *Burke's output theorem* (Burke 1956). Burke telah membuktikan bahwa proses keberangkatan dari antrian M/M/n adalah proses Poisson. Secara khusus, teorema Burke menyatakan bahwa untuk antrian M/M/n dalam kondisi steady state dengan laju kedatangan λ , hasil berikut akan terpenuhi:

- a. Proses keberangkatan adalah proses Poisson dengan laju λ . Jadi, proses keberangkatan secara statistik identik dengan proses kedatangan.
- b. Pada setiap waktu t , jumlah pelanggan di sistem independen dari urutan waktu keberangkatan sebelumnya. Untuk melihat pengaruh dari teorema tersebut secara umum dapat diperhatikan pada jaringan antrian *feed-forward*.

Notasikan γ_i menunjukkan laju kedatangan Poisson dari eksternal menuju antrian- i . Notasikan p_{ij} sebagai proporsi dari user yang telah dilayani pada antrian- i dan melanjutkan ke antrian- j . p_{i0} menunjukkan proporsi user meninggalkan jaringan setelah dilayani pada antrian- i . Jadi, λ_j yang merupakan laju kedatangan user di antrian- j dapat dinyatakan sebagai berikut dan direpresentasikan pada Gambar 2.8 [2].

$$\lambda_1 = \gamma_1 \tag{2.6}$$

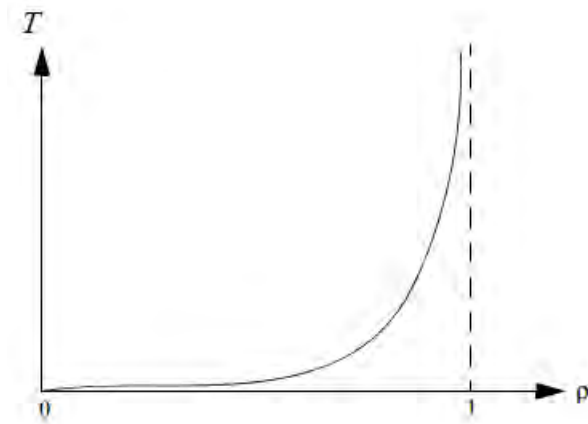
$$\lambda_j = \gamma_j + \sum_{i < j} \lambda_i p_{ij} \tag{2.7}$$



Gambar 2.8 *Feed forward queue* [2]

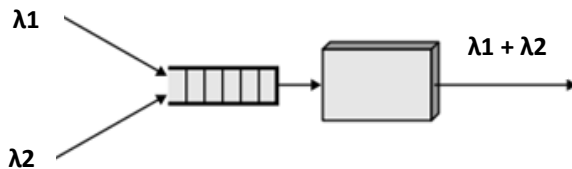
2.3.5 Karakteristik Sistem Antrian

Sistem antrian memiliki berbagai karakteristik seperti yang telah disebutkan pada bab sebelumnya. Dalam subbab ini akan dibahas lebih terkait karakteristik *utilisasi* dan lama waktu pelanggan dalam sistem. Keduanya memiliki hubungan yang berbanding lurus, yang berarti ketika *utilisasi* suatu server naik, maka akan diikuti dengan kenaikan waktu pelanggan dalam sistem. Hubungan antara *utilisasi* dan lama waktu tunggu dapat dilihat pada Gambar 2.9.

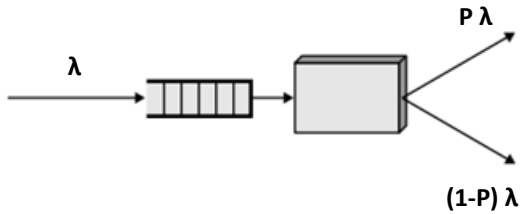


Gambar 2.9 Hubungan antara utilisasi dan waktu tunggu [3]

Rate kedatangan dalam sebuah sistem antrian juga dapat digabung dari beberapa sumber atau biasa disebut dengan agregasi. Demikian pula sebaliknya rate kedatangan dapat dipisah-pisah atau dipartisi ke dalam beberapa server pelayanan. Ilustrasi agregasi dan partisi rate kedatangan dapat dilihat pada Gambar 2.10 dan Gambar 2.11.



Gambar 2.10 Agregasi rate kedatangan



Gambar 2.11 Partisi rate kedatangan

2.4 Optimasi Multi-Tujuan

Optimasi multi-tujuan (*Multi-objective optimization*) merupakan permasalahan optimasi dengan fungsi tujuan lebih dari satu. Di antara fungsi-fungsi tujuan ini sangat mungkin terjadi konflik, misal ketika fungsi tujuan yang satu naik maka fungsi tujuan yang lain sangat mungkin menurun. Dalam dunia nyata banyak sekali permasalahan di dunia nyata yang secara alamiah memiliki banyak tujuan yang ingin untuk di optimasikan.

Bagaimana cara menyelesaikan persoalan optimasi yang memiliki multi-tujuan dengan memiliki konflik. Dalam persoalan optimasi multi-tujuan sangat memungkinkan terjadinya konflik antar tujuan dan merupakan sesuatu yang sulit untuk dihindari. Untuk mengatasi konflik semacam ini digunakanlah konsep yang dikemukakan oleh seorang ilmuwan bernama Vilfredo Pareto di tahun 1896.

Menurut konsep Pareto ini, sekelompok titik berada dalam keadaan optimal atau *maximum ophelimity*, apabila pergeseran sedikit saja akan membuat nilai optimal beberapa fungsi membaik dan pada saat yang sama akan membuat nilai optimal fungsi-fungsi yang lainnya memburuk. Untuk membahas lebih dalam mengenai optimasi multi-tujuan akan dibahas lebih lanjut mengenai solusi *trade-off*, optimasi Pareto beserta teori dominansi, dan sedikit pembahasan mengenai metode yang digunakan untuk menyelesaikan persoalan multi-tujuan sebelum adanya teori optimasi Pareto [4].

2.4.1 Solusi *Trade-Off*

Dalam optimasi dengan tujuan tunggal (*single objective optimization*) hanya bermaksud untuk mencari sebuah solusi optimal. Dengan demikian, algoritma-algoritma penyelesaian optimasi dengan tujuan tunggal, pada prinsipnya hanya mencari satu solusi optimal. Apabila dalam proses optimasi, solusi yang ditemukan setelahnya dan

mempunyai hasil lebih baik maka solusi baru itu adalah solusi yang terbaik.

Untuk optimasi multi-tujuan (*Multi-objective optimization*) akan berbeda situasinya. Apabila terdapat konflik antar tujuan, maka akan dicari solusi –solusi kompromi yang baik (*trade-off solutions*). Artinya tidak satupun dari solusi tersebut yang terbaik untuk setiap fungsi tujuan yang ada. Setiap solusi yang optimal untuk satu tujuan akan membutuhkan kompromi untuk fungsi tujuan yang lain. Dengan demikian tidak terdapat solusi optimal tunggal untuk masalah optimasi multi-tujuan. Sejumlah solusi yang dihasilkan dan semua solusi ini optimal adanya. Bahkan tidak dapat dikatakan tak ada satupun solusi lebih baik dari pada solusi yang lainnya.

Dengan menangani permasalahan multi-tujuan secara benar, tanpa melakukan konversi menjadi satu tujuan, secara prinsip akan diperoleh sekelompok solusi optimal yang disebut *Pareto set* yang dimana solusi itu bersesuaian dengan solusi optimal yang diperoleh melalui konversi dengan berbagai bobot untuk setiap fungsi. Sementara itu apabila tidak terdapat konflik pada sebuah permasalahan multi-tujuan, maka *cardinality* dari *Pareto set* yang dimilikinya adalah satu. Dengan kata lain *Pareto set* yang dihasilkan hanya terdiri dari satu solusi optimal, yang berarti solusi optimal dari setiap fungsi tujuan adalah sama [4].

2.4.2 Optimalitas Pareto dan Dominansi

Saat melakukan optimasi dalam sebuah realita permasalahan tak jarang ditemukan adanya konflik antar tujuan. Konflik yang dimaksud berupa perbaikan optimalitas suatu fungsi atau beberapa fungsi tujuan yang pada saat bersamaan akan memperburuk optimalitas fungsi-fungsi yang lain. Untuk menentukan optimalitas permasalahan multitujuan, dikenal sebuah konsep yang dikemukakan oleh Pareto yang digunakan secara umum. Konsep ini dikenal dengan Optimalitas Pareto.

Optimalitas Pareto adalah konsep yang dikemukakan oleh Pareto. Konsep ini secara sederhana mengatakan bahwa sebuah titik mencapai titik optimalitas apabila tidak ada titik lain yang mampu memperbaiki optimalitas yang sudah tercapai. Dalam bahasa matematis, untuk masalah optimasi multi-tujuan tanpa pembatas [4],

$$\min(f_1(x), f_2(x), \dots, f_k(x)) \quad (2.8)$$

dengan $k \geq 2$ dan $x \in X$, solusi $x^* \in X$ disebut optimal Pareto jika

$$\nexists x \in X, f_i(x) \leq f_i(x^*), \forall i \in \{1, 2, \dots, n\}, \quad (2.9)$$

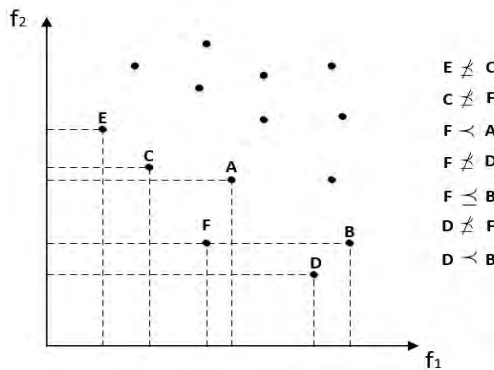
dengan $f_j(x) < f_j(x^*)$ berlaku untuk paling sedikit satu index j .

Optimal Pareto menurut definisi ini dikenal sebagai bentuk lemah (*weak form*). Sementara bentuk tegas (*strict form*) jika yang berlaku adalah tanda “<” menggantikan tanda “ \leq ” pada persamaan 2.9 diatas,

$$\nexists x \in X, f_i(x) < f_i(x^*), \forall i \in \{1, 2, \dots, n\} \quad (2.10)$$

Dengan demikian, optimal Pareto, yang tidak disebut khusus berarti merupakan bentuk tegas atau kuat, yang merupakan bagian dari optimal Pareto yang bersifat lemah (*weak form*). Untuk kasus maksimasi tanda bisa dibalik.

Dalam kasus minimasi, jika $\dot{x}, \ddot{x} \in X$ dan $f_i(\dot{x}) < f_i(\ddot{x})$, maka dapat dikatakan bahwa \ddot{x} didominasi \dot{x} , dan $f_i(\ddot{x})$ didominasi $f_i(\dot{x})$. Dominasi ini bersifat tegas (*strictly domination*), sementara itu solusi \ddot{x} didominasi secara lemah oleh \dot{x} , jika $f_i(\dot{x}) \leq f_i(\ddot{x})$. Apabila solusi \dot{x} mendominasi secara tegas, berarti solusi \dot{x} juga mendominasi secara lemah, namun tidak sebaliknya. Umumnya metode-metode penyelesaian masalah optimasi multi-tujuan mengadopsi konsep dominasi ini untuk mencari solusi-solusi yang tak terdominasi (*non-dominated solutions*).



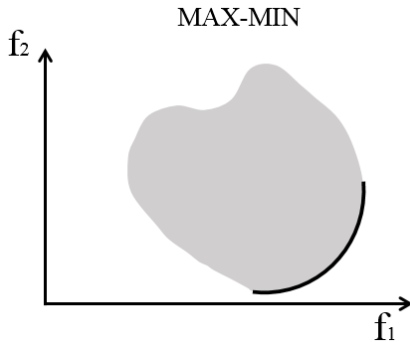
Gambar 2.12 Konsep dominansi

Pada Gambar 2.12 menunjukkan f_1 dan f_2 ingin diminimalkan. Menampilkan sejumlah titik solusi yang terletak antara ruang solusi. Jika solusi A dan F dibandingkan, maka terlihat solusi F memiliki hasil lebih baik dari pada solusi A, dengan demikian inilah yang disebut solusi F mendominasi secara tegas solusi A ($F \prec A$). Jika solusi B dan F dibandingkan maka dapat terlihat bahwa solusi F hanya mendominasi solusi B pada satu fungsi tujuan saja yaitu fungsi f_1 , sementara itu untuk fungsi f_2 nilai solusi F dan B adalah sama. Maka dalam kasus ini dapat dikatakan solusi F mendominasi solusi B secara lemah ($F \prec B$), karena hanya ada satu fungsi tujuan yang nilai optimalnya lebih baik.

Selanjutnya dibandingkan solusi D dan F, terlihat bahwa F lebih baik daripada solusi D untuk fungsi f_1 , namun lebih buruk untuk fungsi f_2 . Begitu juga sebaliknya solusi D terlihat lebih baik daripada solusi F untuk fungsi f_2 , maka dalam kasus ini dapat dikatakan solusi D dan F saling tidak mendominasi. Yang dimana sama halnya dengan solusi E dan C. Maka solusi E, C, F dan D adalah *non-dominated solutions*. Himpunan dari *non-dominated solutions* disebut *non-dominated set*. Himpunan ini memenuhi dua kondisi berikut.

- a. Setiap pasang solusi dalam himpunan ini tidak boleh saling mendominasi
- b. Setiap solusi yang berada diluar himpunan didominasi paling sedikit oleh satu anggota himpunan

Apabila *non-dominated set* yang diperoleh berasal dari keseluruhan ruang pencarian maka himpunan ini dikenal sebagai Pareto set. Dengan demikian seluruh anggota Pareto set adalah yang terbaik dan yang paling optimal sesuai fungsi-fungsi tujuan yang ada. Solusi yang tergabung dalam Pareto set tidak dapat dikatakan ada salah satu solusi yang terbaik dari solusi yang lainnya. Sehingga seluruh solusi yang terdapat di Pareto set merupakan solusi yang terbaik yang ditemukan. Pareto set dapat berbeda-beda karakteristiknya sesuai dengan kombinasi fungsi tujuan maksimasi dan minimasi. Posisi kombinasi fungsi tujuan tersebut juga berpengaruh seperti contoh f_1 minimasi dan f_2 maksimasi akan berbeda himpunan Pareto set dengan f_1 maksimasi dan f_2 minimasi. Karakteristik dari Pareto set berdasarkan fungsi jenis tujuannya dapat diilustrasikan dari Gambar 2.13. Pada gambar tersebut diilustrasikan kasus f_1 maksimasi dan f_2 minimasi.



Gambar 2.13 Solusi visibel sesuai kombinasi fungsi tujuan [8]

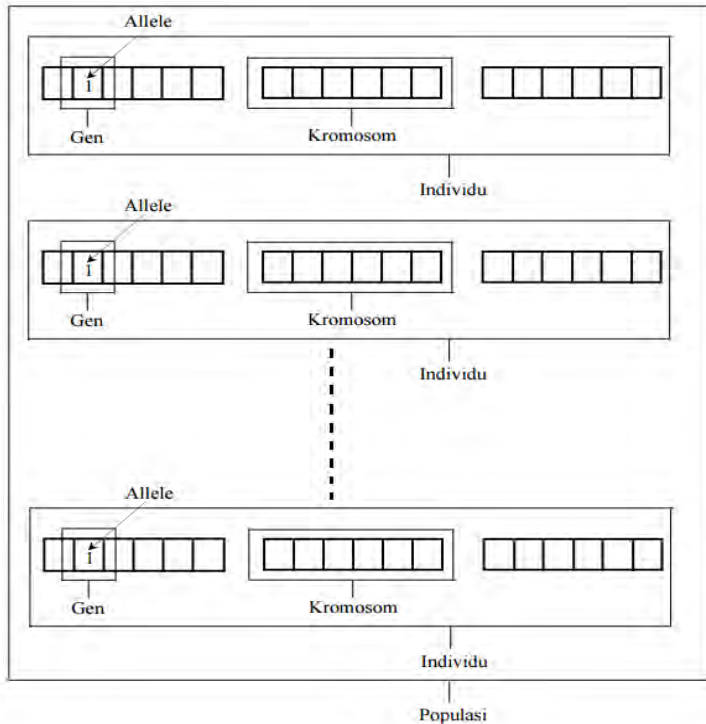
2.5 Genetic Algorithm

Genetic algorithm atau algoritma genetika (GA) masuk dalam kelompok Evolutionary Elgorithm. Genetic algorithm didasarkan pada prinsip-prinsip genetika dan seleksi alam. Elemen-elemen dasar dari genetika alam adalah reproduksi, kawin silang, dan mutasi. Kemudian diadopsi menjadi algoritma komputasi untuk mencari solusi suatu permasalahan optimasi. GA pertama kali dicetuskan oleh John Holland, Profesor University of Michigan pada tahun 1970 dan dikembangkan oleh Goldberg [5].

GA yang dikembangkan oleh Goldberg adalah yang menyatakan bahwa kelangsungan hidup suatu makhluk dipengaruhi aturan “yang kuat adalah yang menang”. Darwin juga menyatakan bahwa kelangsungan tersebut dapat dipertahankan melalui proses reproduksi, *crossover*, dan mutasi dan kemudian diadopsi menjadi algoritma komputasi untuk mencari solusi suatu permasalahan dengan cara yang lebih alamiah.

Dalam Genetic algorithm kromosom adalah satu hal yang penting, satu kromosom atau individu mewakili satu vektor solusi yang akan diproses dengan prosedur-prosedur algoritma genetik. Terkadang representasi dari kromosom dapat ditulis dengan nilai kontinu, dalam hal lain juga dapat ditulis dengan angka biner sehingga harus melalui encoding atau pengkodean lalu melakukan decoding kembali. Hal ini tergantung dengan permasalahan yang dihadapi. Dalam permasalahan optimasi fungsi sering kali menggunakan angka biner, karena dengan angka biner proses kawin silang dan mutase akan lebih banyak variasinya.

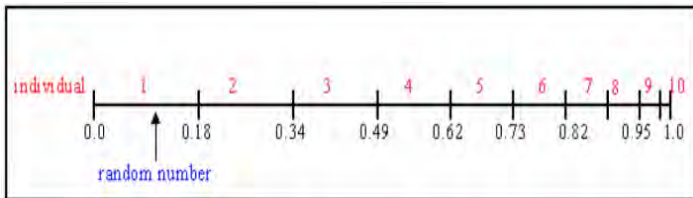
Dalam Genetic algorithm akan dibangkitkan sejumlah kromosom atau individu yang disebut dengan populasi sehingga dengan dibangkitkannya populasi ini, maka akan tersedia banyak pilihan solusi. Populasi yang terjadi pertama kali biasa disebut dengan populasi awal atau inialisasi populasi, seperti pada Gambar 2.14 [6].



Gambar 2.14 Susunan algoritma genetika [5]

Dalam Genetic algorithm dikenal suatu fungsi yang disebut fungsi *fitness*. Dimana fungsi *fitness* ini digunakan untuk mengukur tingkat kebaikan atau kesesuaian suatu solusi yang dicari. Fungsi *fitness* biasanya berhubungan langsung dengan fungsi tujuan. Dalam kondisi tertentu juga bisa dimodifikasi dari fungsi tujuannya, hal ini menyesuaikan dengan tujuan yang diinginkan [7].

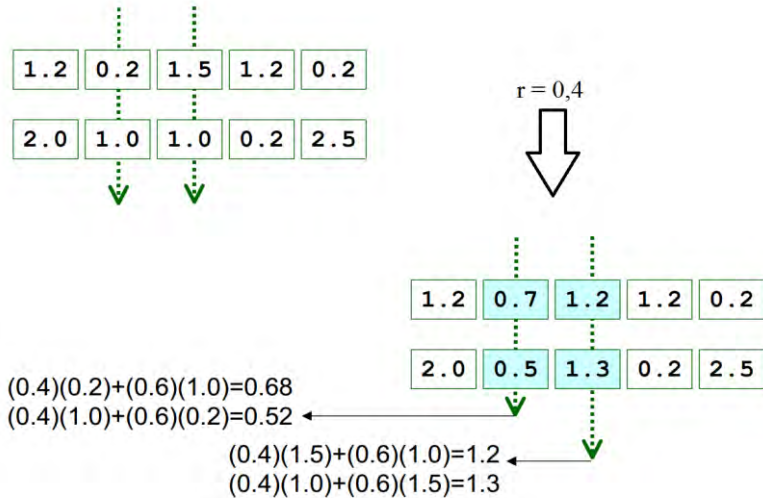
Etilisme merupakan usaha untuk mempertahankan individu atau kromosom terbaik dari populasi yang ada. Sehingga individu terbaik ini tidak terseleksi maupun mengalami kawin silang saat proses genetika selanjutnya. Sehingga individu terbaik ini dapat bertahan di generasi selanjutnya dan sangat mempengaruhi hasil dari Genetic algorithm. Sebelum dilakukannya kawin silang terlebih dahulu dilakukan proses seleksi orang tua yang akan dikawinkan. Seleksi ini dapat dilakukan dengan beberapa metode antara lain, roda lotere, turnamen, dan *stochastic universal selection*. Gambar 2.15 merupakan ilustrasi dari pemilihan induk dengan metode *stochastic universal selection* (SUS). Sebelum melakukan seleksi dengan SUS terlebih dulu dilakukan normalisasi terhadap nilai-nilai *fitness*. Hal ini bertujuan untuk membuat kumulatif nilai *fitness* antara nol sampai 1. Sehingga mempermudah dalam proses perhitungan. Pada umumnya metode *stochastic universal selection* (SUS) ini lebih sering digunakan apabila sedang mengatasi permasalahan yang bilangannya merupakan bilangan kontinyu bukan bilangan biner [7].



Gambar 2.15 *Stochastic Universal Selection* [7]

Setelah terpilih dua individu yang terpilih menjadi orang tua setelah itu akan dilakukan proses kawin silang yang akan menghasilkan individu baru untuk semua populasi, kecuali individu terbaik. Kromosom baru yang disebut dengan *off spring*. Jumlah kromosom dalam populasi yang mengalami *crossover* ditentukan oleh parameter yang disebut dengan *crossover rate*. Gambar dibawah merupakan ilustrasi dari *crossover* dengan menggunakan metode aritmatik yang sebagian banyak sesuai dengan bilangan kontinyu. *Crossover* aritmatik dapat dilakukan dengan formula (2.11) dan diilustrasikan pada Gambar 2.16 [5].

$$\begin{aligned}
 \text{Anak1} &= r * \text{induk1} + (1-r) * \text{induk2} \\
 \text{Anak2} &= r * \text{induk2} + (1-r) * \text{induk1}
 \end{aligned}
 \tag{2.11}$$



Gambar 2.16 Aritmatik *crossover*

Mekanisme perubahan susunan genetika akibat adanya faktor alam disebut dengan mutasi. Jumlah gen dalam populasi yang mengalami mutasi ditentukan oleh parameter yang dinamakan *mutation rate*. Setelah beberapa generasi akan dihasilkan kromosom yang nilainya konvergen pada suatu nilai, di mana merupakan solusi terbaik yang dihasilkan oleh GA terhadap permasalahan yang ingin diselesaikan.

Sebuah kromosom dibentuk dari komponen-komponen penyusun yang disebut sebagai gen dan nilainya dapat berupa bilangan numerik, biner, simbol, ataupun karakter. Sebagian besar kromosom memiliki nilai yang berupa nilai biner atau kontinyu. Penggunaan bilangan tersebut bergantung pada jenis permasalahan yang dihadapi. Penggunaan jenis bilangan pada kromosom tersebut juga menentukan metode yang sesuai untuk proses seleksi dan *crossover*. Kromosom akan berevolusi secara berkelanjutan yang disebut dengan generasi. Dalam tiap generasi kromosom tersebut dievaluasi tingkat keberhasilan nilai solusinya terhadap masalah yang ingin diselesaikan (fungsi objektif) menggunakan ukuran yang disebut dengan *fitness*. Individu yang memiliki nilai *fitness* tinggi akan memiliki peluang lebih besar untuk terpilih lagi pada generasi selanjutnya. Diagram alir GA dapat dilihat pada Gambar 2.17.



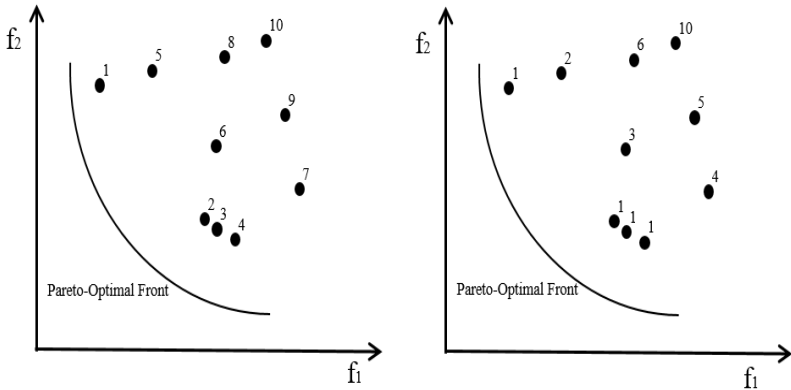
Gambar 2.17 Diagram alir proses algoritma genetika

2.6 *Multi-objective Genetic Algorithm*

Carlos M. Fonseca dan Peter J. Fleming (1993) menemukan model MO-GA. Pendekatan ini terdiri dari sebuah skema di mana ranking dari suatu individu tertentu berhubungan dengan jumlah individu –individu dalam populasi saat ini yang mendominasinya [9]. Dengan demikian individu tak terdominasi diberi ranking satu, sementara individu lainnya diberi ranking berdasar banyaknya solusi yang mendominasinya, katakanlah n yang mendominasi mereka. Selanjutnya individu tersebut menempati ranking $(n+1)$ seperti formula (2.12). Individu yang memiliki ranking terendah yang terpilih untuk membentuk populasi generasi berikutnya [4].

$$r_i = 1 + n_i \tag{2.12}$$

Dalam contoh Gambar 2.18 seluruh *non-dominated solutions* diberi ranking 1, ketika tidak ada solusi yang mendominasinya. Pada Gambar 2.18 terlihat bahwa jika menggunakan ranking pareto seperti pada umumnya maka akan didapat solusi yang mempunyai ranking 1 sampai 10. Namun untuk menentukan ranking dalam MO-GA ini Fonseca dan Fleming sedikit melakukan perubahan. Solusi tersebut diberi ranking sesuai banyaknya individu yang mendominasinya ditambah satu. Pada Gambar 2.18 diperhatikan ada sebuah solusi yang memiliki ranking 6, dengan kata lain ada sejumlah 5 solusi yang mendominasinya. Dengan metode ranking yang di terapkan ini maka akan didapat bahwa tidak semua ranking mempunyai himpunan solusi, seperti yang dapat dilihat pada gambar dibawah tidak terdapat solusi yang memiliki ranking 7,8, dan 9 [8].



Gambar 2.18 Ranking Pareto pada MO-GA [8]

Setelah mendapatkan ranking dari solusi dan jumlah solusi yang berada dalam setiap ranking yang sama tersebut, kemudian dihitung *average fitness* dari setiap ranking solusi. Untuk solusi yang mempunyai ranking yang sama maka nilai dari *average fitness* akan sama pula. Untuk menghitung *average fitness* data dihitung dengan formula dibawah ini.

$$F_i = N - \sum_{k=1}^{r_i-1} \mu(k) - 0.5(\mu(r_i) - 1) \quad (2.13)$$

Keterangan:

F_i = fungsi ke i

$\mu(r_i)$ = jumlah solusi yang berada dalam ranking i

N = banyak solusi

Untuk mempertahankan keragaman di antara *nondominated solutions*, diterapkan *niching* di antara solusi-solusi tiap ranking. *Niche count*-nya dinyatakan sebagai penjumlahan nilai-nilai *sharing function*. Dengan $\mu(r_i)$ menyatakan jumlah solusi dalam ranking r_i . *Niche count* dihitung dengan formula (2.14).

$$nc_i = \sum_{j=1}^{\mu(r_i)} Sh(d_{ij}) \quad (2.14)$$

Sharing function disini menggunakan metrik jarak yang dihitung dengan menggunakan nilai-nilai fungsi tujuan. Kemudian di normalisasi. Normalisasi jarak antara dua solusi i dan j dalam ranking yang sama diberikan sebagai berikut. Dimana $f_k^{(max)}$ dan $f_k^{(min)}$ adalah masing-masing nilai maksimum dan minimum dari fungsi tujuan ke- k . Metrik jarak ini dihitung menggunakan formula (2.15).

$$d_{ij} = \sqrt{\sum_{k=1}^M \left(\frac{f_k^{(i)} - f_k^{(j)}}{f_k^{(max)} - f_k^{(min)}} \right)^2} \quad (2.15)$$

Untuk menghitung *sharing function* itu sendiri merupakan hasil dari perkalian metrik jarak d_{ij} . Dengan menggunakan formula (2.16) dapat dihitung *sharing function*. Yang memiliki syarat $d \leq \sigma_{share}$ d merupakan metrik jarak antara 2 solusi, maka $Sh(d)$ akan bernilai. Sebaliknya jika $d \geq \sigma_{share}$ maka $Sh(d)$ akan bernilai sama dengan nol.

$$Sh(d) = \begin{cases} 1 - \left(\frac{d}{\sigma_{share}} \right)^\alpha & d \leq \sigma_{share} \\ 0 & otherwise \end{cases} \quad (2.16)$$

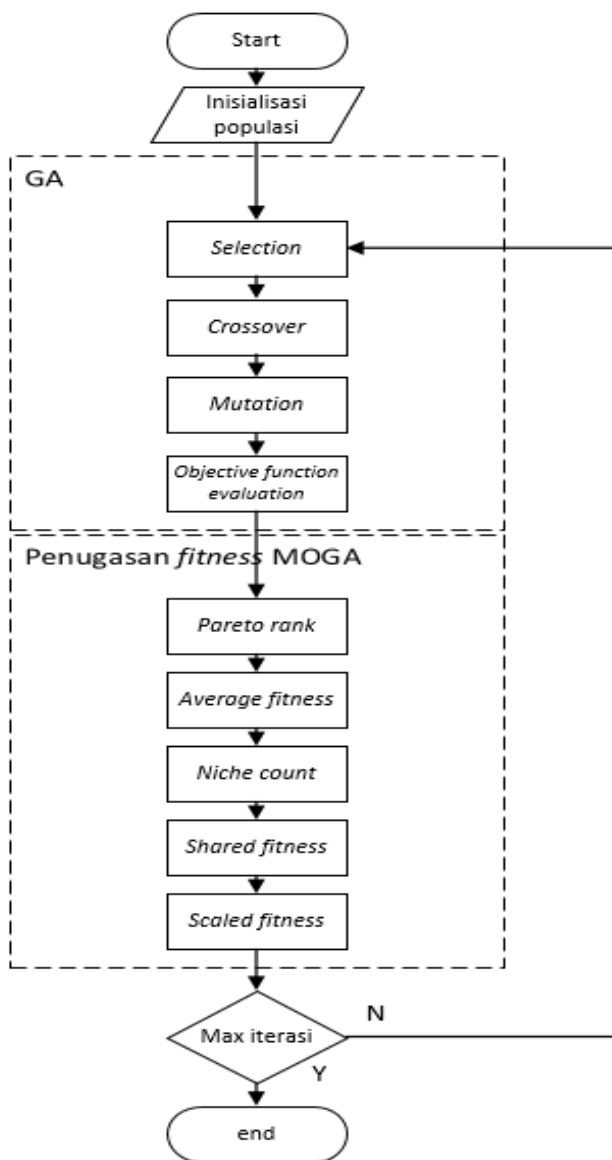
Setelah mendapatkan hasil *Niche count* dari setiap solusi yang ada. Kemudian hasil dari *Niche count* tersebut akan digunakan untuk mencari *Shared fitness*, yang merupakan *Average fitness* dibagi dengan *Niche count* untuk setiap korespondensinya. *Shared fitness* dapat dihitung dengan formula (2.17).

$$F'_j = \frac{F_j}{nc_j} \quad (2.17)$$

Langkah selanjutnya adalah menghitung *Scaled fitness* yang merupakan nilai fitness yang telah di skalarisasi berdasarkan nilai-nilai *Shared fitness* pada solusi ranking yang sama. *Scaled fitness* dapat dihitung dengan formula (2.18).

$$F''_j = \frac{F_j \mu(r)}{\sum_{k=1}^{\mu(r)} F'_k} F'_j \quad (2.18)$$

Dengan beberapa langkah diatas dapat dihitung nilai *fitness* dari setiap solusi yang dihasilkan oleh perhitungan GA. Perbedaan antara GA satu tujuan dan MO-GA adalah terletak pada jumlah hasil solusi optimal yang dikeluarkan. Perbedaan lainnya juga terletak pada penugasan nilai *fitness* yang melalui beberapa langkah yaitu ranking Pareto, *Average fitness*, *Niche count*, *Shared fitness*, dan *Scaled fitness*. Selibuhnya untuk perhitungan yang terkait dengan *selection*, *crossover*, dan *mutation* dapat dikatan tidak ada perbedaan anantara MO-GA dan GA satu tujuan. Diagram alir MO-GA dapat dilihat pada Gambar 2.19.



Gambar 2.19 Diagram alir MO-GA

BAB III

PERANCANGAN SISTEM

Pada bab ini membahas tentang beberapa tahapan dalam merancang alur sistem rujukan, simulasi, permodelan optimisasi dari permasalahan, dan algoritma untuk menyelesaikan permasalahan optimasi distribusi pasien dengan *Multi-objective Genetic Algorithm* (MO-GA). Pada bagian pertama akan dijelaskan mengenai permasalahan yang sedang dihadapi. Kemudian dijelaskan mengenai alur rujukan dari sistem rujukan berjenjang. Setelah dipahami permasalahan dan sistem serta alur rujukan yang ada, langkah selanjutnya adalah merancang model matematis dari permasalahan tersebut. Berikutnya akan dirancang simulasi dengan menggunakan Arena untuk melakukan simulasi kondisi awal dan kondisi yang telah dioptimalkan. Langkah yang terakhir adalah merancang MO-GA untuk mengoptimasikan dua fungsi tujuan yaitu memaksimalkan rata-rata utilisasi dan meminimalkan rata-rata waktu tunggu di poli setiap Rumah sakit.

3.1 Identifikasi Masalah

Masalah yang sedang dihadapi oleh fasilitas pelayanan kesehatan masyarakat terutama di Rumah sakit ialah tidak merata jumlah pasien yang datang ke tiap-tiap Rumah sakit. Hal ini menyebabkan penumpukan pada beberapa Rumah sakit saja, kondisi ini yang membatasi Rumah sakit memberikan pelayanan yang maksimal dan nyaman.

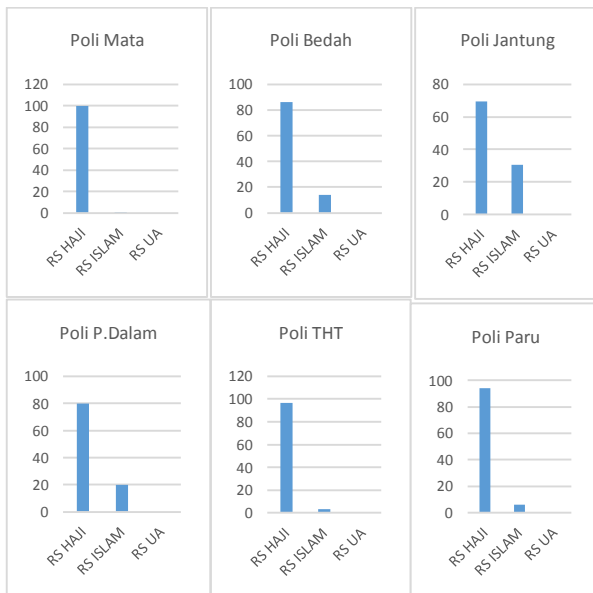
Pada penelitian tugas akhir ini dipilih tiga Rumah sakit yang dapat memberikan pelayanan fasilitas kesehatan tingkat 2 dan tentunya tergabung dalam mitra BPJS Kesehatan, yaitu Rumah sakit Haji, Rumah sakit Islam Jemursari, dan Rumah sakit Universitas Airlangga. Ketiga Rumah sakit tersebut merupakan Rumah sakit bertipe B dan C yang menjadi tujuan rujukan dari sebagian besar fasilitas kesehatan tingkat 1. Adapun sebagai contoh Rumah sakit bertipe A yaitu Rumah sakit DR. Soetomo.

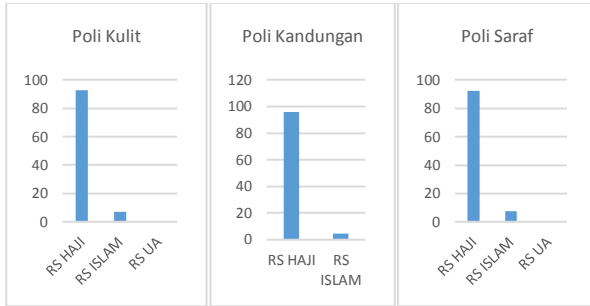
Pertimbangan lain untuk memilih ketiga Rumah sakit tersebut sebagai objek penelitian adalah banyaknya jumlah rujukan pasien yang dirujuk ke Rumah sakit Haji dan Rumah sakit Islam Jemursari. Berdasarkan data dari BPJS Kesehatan tahun 2015 jumlah rujukan ke Rumah sakit Haji sebanyak 128.515 rujukan yang merupakan 15,86% dari keseluruhan rujukan yang dilakukan, sedangkan rujukan ke Rumah

sakit Islam Jemursari sebanyak 46.968 rujukan yang merupakan 5,8% dari keseluruhan rujukan yang dilakukan. Sementara itu Rumah sakit Airlangga berdasarkan data dari BPJS Kesehatan tahun 2015, hanya mendapatkan 24 rujukan. Sehubungan dengan tingkat kesibukan server atau biasa disebut utilisasi. Jika rate kedatangan lebih tinggi daripada rate pelayanan dan jumlah server yang tersedia, maka utilisasi dari server tersebut akan lebih lebih dari satu, hal ini tentunya menandakan bahwa server pelayanan tersebut terlalu sibuk dan akan menyebabkan antrian yang panjang serta waktu tunggu yang lama.

Selain itu faktor wilayah atau tempat dimana Rumah sakit itu berada menjadi pertimbangan ketiga untuk memilih ketiga Rumah sakit tersebut sebagai objek penelitian. Karena seluruh puskesmas di wilayah Surabaya Timur dipilih sebagai objek penelitian yang berjumlah 13.

Dalam penelitian tugas akhir ini akan dipilih poli penyakit yang sering mendapatkan rujukan pasien dari fasilitas kesehatan tingkat 1. Data tersebut dapat diambil dari BPJS Kesehatan. Yang kemudian didapat Sembilan poli penyakit yang banyak mendapatkan rujukan pasien di Surabaya, yaitu poli bedah, penyakit dalam, mata, THT, kulit, paru, kandungan, saraf dan jantung.





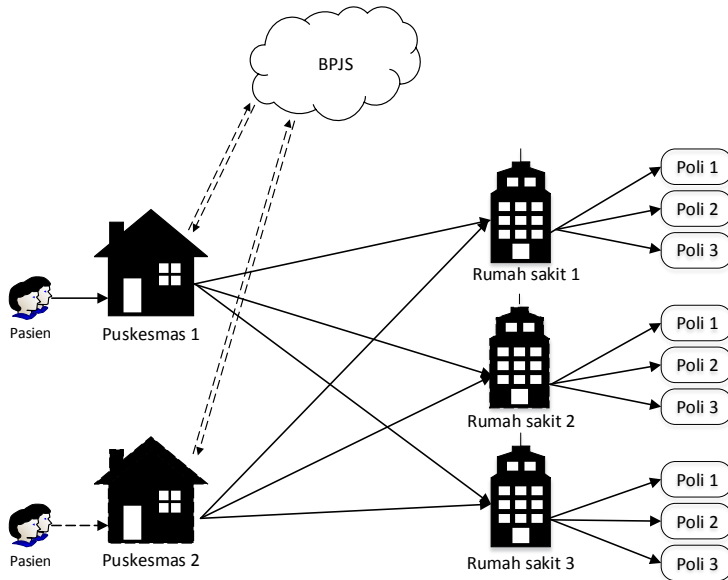
Gambar 3.1 Grafik proporsi rujukan kondisi awal

Melihat proporsi rujukan dari sembilan penyakit se-Surabaya pada Gambar 3.1 dapat disimpulkan bahwa pasien warga Surabaya mendapat rujukan ke Rumah sakit Haji lebih besar daripada ke RS Islam dan RS Universitas Airlangga. Hal ini tentunya membuat keadaan di RS Haji lebih padat daripada dua Rumah sakit lainnya, yang menyebabkan penumpukan pasien. Serta membuat pasien menunggu lama untuk mendapatkan pelayanan kesehatan yang dibutuhkan.

3.2 Deskripsi Sistem Optimisasi Rujukan Pasien

Setelah mengetahui permasalahan yang yang dihadapi, yaitu menumpuknya pasien di salah satu Rumah sakit saja yang disebabkan oleh tidak idealnya proporsi rujukan terhadap 3 rumah sakit tersebut. Kemudian dirancang sebuah sistem rujukan yang akan memperbaiki proporsi rujukan agar pasien tidak menumpuk di salah satu Rumah sakit saja yang dapat menyebabkan nilai utilisasi menjadi buruk yaitu diatas 1 dan tentunya menyebabkan waktu antri yang lama.

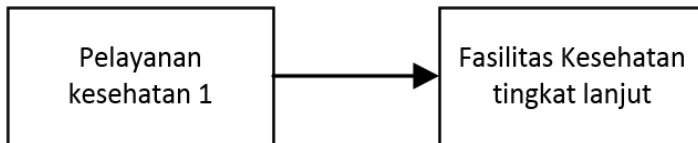
Dalam perancangan sistem optimisasi rujukan pasien BPJS ini. Seperti yang dapat dilihat pada Gambar 3.2 yang berperan menjadi pengendali sistem adalah BPJS, direpresentasikan oleh sebuah *cloud* yang dimana akan memberi rujukan kepada pasien. Awal mulanya pelayanan kesehatan pasien kana berjalan seperti biasa. Pasien BPJS akan mengunjungi Puskesmas dimana ia terdaftar untuk mendapatkan pelayanan kesehatan. Apabila Puskesmas tidak mampu untuk memberi pelayanan yang sesuai kebutuhan pasien. Pihak Puskesmas akan memberikan rujukan terhadap pasien ke Rumah sakit. Disinilah dirancang sebuah sistem yang berguna untuk menentukan Rumah sakit yang akan menjadi tujuan pasien tersebut.



Gambar 3.2 Alur pasien mendapat pelayanan kesehatan

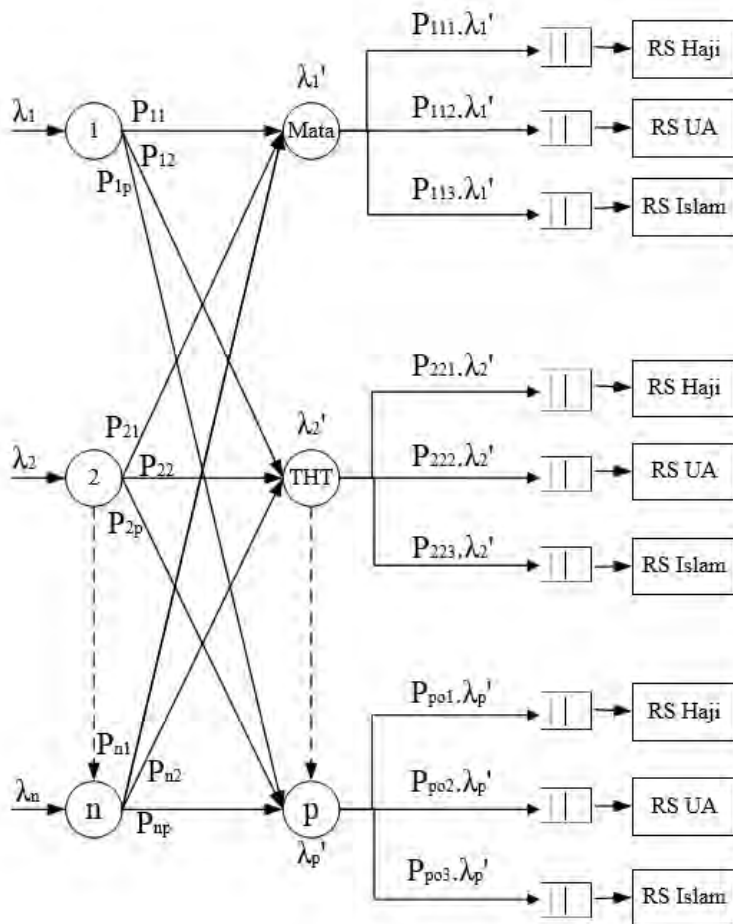
3.3 Perancangan Jaringan Antrian Rujukan Pasien

BPJS menerapkan sistem rujukan yang berjenjang mulai dari pelayanan kesehatan tingkat 1 kemudian tingkat 2 dan yang terakhir ialah pelayanan kesehatan tingkat 3. Pasien peserta BPJS yang sakit haruslah mendatangi pelayanan kesehatan tingkat 1 terlebih dahulu. Pelayanan kesehatan tingkat 1 ini dapat berupa puskesmas, klinik, atau dokter pribadi. Apabila pelayanan kesehatan tingkat 1 ini tidak bisa memberikan pelayanan kesehatan yang dibutuhkan oleh pasien maka pasien akan dirujuk ke pelayanan kesehatan tingkat 2 dan jika tetap tidak mampu pasien akan dirujuk ke pelayanan kesehatan tingkat 3.



Gambar 3.3 Alur rujukan pada sistem rujukan berjenjang BPJS

Namun pada tugas akhir ini sebagai batasan penelitian akan dibatasi pada penelitian alur rujukan dan proporsi rujukan dari fasilitas kesehatan tingkat 1 ke fasilitas kesehatan tingkat 2 atau tingkat 3 saja yang dapat disebut dengan fasilitas kesehatan tingkat lanjut seperti Gambar 3.3. Sehingga dapat diartikan bahwa perancangan model matematis dan optimasi distribusi pasien hanya memperhatikan rujukan vertikal.

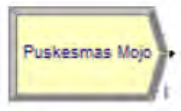


Gambar 3.4 Perancangan jaringan antrian rujukan pasien

Seperti alur Gambar 3.4 terlihat bahwa rate rujukan dari penyakit sejenis akan digabungkan atau diagregasikan terlebih dahulu kemudian dipartisi kembali ke poli penyakit yang sesuai. Untuk itu perlu menghitung agregasi rate rujukan dari setiap penyakit.

3.4 Perancangan Simulasi dengan Arena

Untuk melakukan simulasi sistem antrian di seluruh poli Rumah sakit menggunakan software ARENA. Dalam simulasi ini cukup menggunakan *basic process*. Modul yang digunakan meliputi *create*, *decide*, *process*, dan *dispose*. Perancangan modul *create* (Gambar 3.5) untuk rate kedatangan pasien di setiap puskesmas. Pada Gambar 3.6 pola antar kedatangan (*time between arrivals*) pasien dalam simulasi ini diasumsikan dengan menggunakan distribusi poisson dengan satuan waktu menit. Pasien bisa datang sepanjang waktu sehingga tidak ada pembatasan jumlah kedatangan (*infinite*).

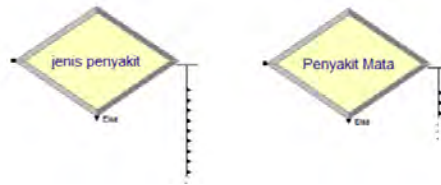


Gambar 3.5 Modul *Create*

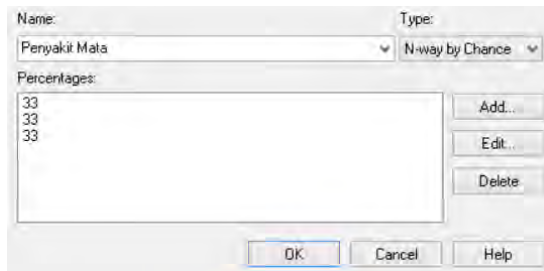
Name:	Entity Type:	
Puskesmas Mojo	Pasien	
Time Between Arrivals		
Type:	Expression:	Units:
Expression	POIS(Mean)	Minutes
Entities per Arrival:	Max Arrivals:	First Creation:
1	Infinite	0.0
OK Cancel Help		

Gambar 3.6 Penentuan parameter modul *Create*

Kemudian pembuatan modul *decide* (Gambar 3.7) yaitu modul yang berguna untuk memecah atau mempartisi rate kedatangan dengan menggunakan probabilitas. Ada dua kegunaan modul *decide* dalam simulasi ini. Yang pertama adalah membagi kedatangan berdasarkan jenis penyakit dan yang kedua adalah untuk membagi kedatangan ke setiap poli Rumah sakit. Tipe *decide* yang digunakan adalah banyak pilihan dengan peluang (*N-way by Chance*) seperti yang terlihat pada Gambar 3.8.



Gambar 3.7 Modul *decide*



Gambar 3.8 Menentukan jenis *decide* *N-way by chance*

Lalu langkah selanjutnya adalah merancang modul *process* (Gambar 3.9) dimana modul ini merupakan representasi dari sebuah server atau poli di Rumah sakit. Pada Gambar 3.10 merupakan penentuan rate pelayanan dari seluruh poli yang ada diasumsikan menggunakan distribusi eksponensial dengan satuan waktu menit. *Action* dari poli ini dibuat *seize delay release* yang berarti pelayanan pasien (entitas) ini akan dilakukan oleh sebuah *resource* dalam kasus ini adalah dokter dan kemudian akan dikeluarkan dari poli apabila telah selesai mendapatkan pelayanan. Hal lain yang perlu diatur adalah *resource* dalam kasus ini adalah dokter, dipilih tipe *set* yang merupakan kumpulan dari *resource* yang dapat diatur cara kerja dari setiap *resource*-nya. Kemudian diatur pada tabel *set* yang dimana akan menentukan jumlah *resource* yang ada di dalam *set* tersebut seperti pada Gambar 3.11, atau dapat dikatakan menentukan jumlah dokter yang terdapat di suatu poli Rumah sakit.



Gambar 3.9 Modul *Process*

Name: Poli Mata RS Haji Type: Standard

Logic:

Action: Seize Delay Release Priority: Medium(2)

Resources:

- Set: Set 1, 1 Cyclical
- <End of list>

Buttons: Add.., Edit.., Delete

Delay Type: Expression Units: Minutes Allocation: Value Added

Expression: EXPD(Mean)

Report Statistics

Buttons: OK, Cancel, Help

Gambar 3.10 Penentuan parameter modul *Process*

Type: Set

Set Name: Set 1 Quantity: 1

Selection Rule: Cyclical Save Attribute:

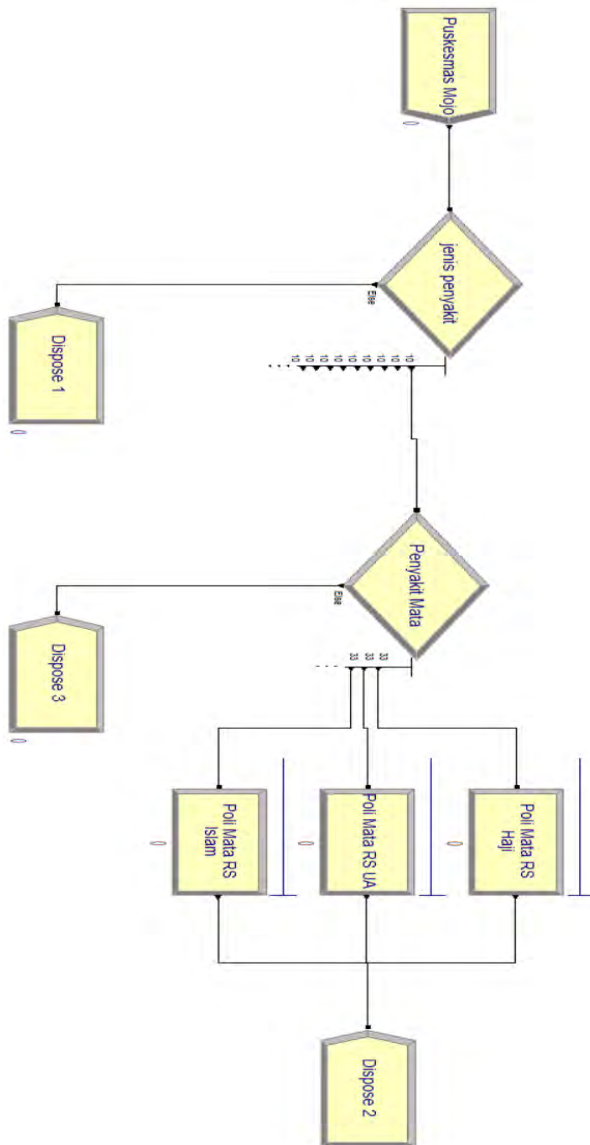
Buttons: OK, Cancel, Help

Set - Basic Process			
Name	Type	Members	
Set 1	Resource	3 rows	

Double-click here to add a new row.

Members	
	Resource Name
1	dokter 1
2	dokter 2
3	dokter 3

Gambar 3.11 Penentuan parameter *resource*



Gambar 3.12 Perancangan model simulasi untuk poli sejenis

3.5 Perancangan Model Optimisasi

Untuk dapat merancang model matematis permasalahan jaringan antrian pada sistem rujukan berjenjang BPJS ini perlu pendefinisian variabel-variabel yang akan digunakan untuk merumuskan model matematis, berikut definisi variabel yang akan digunakan

n = Jumlah puskesmas dan wilayah Surabaya.

m = Jumlah Rumah sakit.

o = Jumlah jenis poli.

p = Jumlah jenis penyakit.

i = Puskesmas ke- i ($i=1,2,\dots,n$).

j = Rumah sakit ke- j ($j=1,2,\dots,m$).

k = Poli ke- k ($k=1,2,\dots,o$)

l = Penyakit ke- l ($l=1,2,\dots,l$)

S = jumlah seluruh poli yang ada di semua Rumah sakit.

3.5.1 Fungsi Objektif 1

Fungsi objektif yang pertama dari permasalahan optimasi distribusi rujukan pasien yang berkaitan langsung dengan sistem antrian adalah memaksimalkan rata-rata utilisasi dari seluruh *resource* yang ada di seluruh poli Rumah sakit. Untuk mencari nilai utilisasi dari sebuah sistem antrian *multi sever*, dengan rate kedatangan yang telah diagregasi dalam setiap jenis penyakit. Berikut formula untuk mencari nilai utilisasi di setiap poli dengan P menyatakan probabilitas $P_{lk1} + P_{lk2} + P_{lk3} = 1$. Untuk setiap l yang sama dan setiap k yang sama.

$$\rho_{kj} = \frac{P_{lkj}\lambda'_l}{c_{kj}\mu_{kj}} \quad (3.1)$$

Sesuai dengan formula *utilisasi* di atas maka dapat dirumuskan fungsi objektif 1 yang dimana memaksimalkan rata-rata *utilisasi*. Yang dimana nilai probabilitas tersebut akan dihitung menggunakan MO-GA untuk mendapatkan proporsi rujukan yang tepat untuk setiap poli sejenis.

$$MAX \quad F1 = \frac{\sum_{l=1}^p \sum_{k=1}^m \sum_{j=1}^o \frac{P_{lkj}\lambda'_l}{c_{kj}\mu_{kj}}}{S} \quad (3.2)$$

3.5.2 Fungsi Objektif 2

Fungsi objektif yang kedua dari permasalahan optimasi distribusi rujukan pasien yang berkaitan langsung dengan sistem antrian adalah meminimalkan rata-rata waktu tunggu pasien di seluruh poli Rumah sakit. Untuk mencari nilai rata-rata waktu tunggu dari sebuah sistem antrian *multi sever* adalah seperti dibawah ini.

$$P_{0kj} = \frac{c_{kj}!(1-\rho_{kj})}{(\rho_{kj}c_{kj})^{c_{kj}} + c_{kj}!(1-\rho_{kj}) \left\{ \sum_{a=0}^{c-1} \frac{1}{a!} (\rho_{kj}c_{kj})^a \right\}} \quad (3.3)$$

$$T_{wkj} = \frac{(\rho_{kj}c_{kj})^{c_{kj}}}{c_{kj}!(1-\rho_{kj})^2 c_{kj}\mu_{kj}} P_{0kj} \quad (3.4)$$

Sesuai dengan formula untuk mencari lama waktu tunggu di atas maka dapat dirumuskan fungsi objektif 2 yang dimana meminimalkan rata-rata waktu tunggu.

$$\begin{aligned} & \text{MIN } F2 \\ & \frac{(\rho_{kj}c_{kj})^{c_{kj}}}{c_{kj}!(1-\rho_{kj})^2 c_{kj}\mu_{kj}} \frac{c_{kj}!(1-\rho_{kj})}{(\rho_{kj}c_{kj})^{c_{kj}} + c_{kj}!(1-\rho_{kj}) \left\{ \sum_{a=0}^{c-1} \frac{1}{a!} (\rho_{kj}c_{kj})^a \right\}} \\ & = \frac{\hspace{10em}}{S} \end{aligned} \quad (3.5)$$

3.5.3 Konstrain Pertidaksamaan 1

Konstrain pertidaksamaan 1 adalah batasan yang mengatur bahwa nilai *utilisasi* dari sebuah server tidak boleh lebih dari satu. Apabila *utilisasi* dari sebuah server lebih dari satu maka dapat dikatakan server tersebut mempunyai kualitas yang buruk. Oleh karena itu rate kedatangan harus lebih kecil daripada rate pelayanan dikali jumlah *resource*.

$$\rho_{kj} < 1 \quad \forall \quad j = 1,2,3 \text{ \& } k = 1,2,3,\dots,o \quad (3.6)$$

3.5.4 Konstrai Persamaan 2

Konstrai persamaan 2 adalah batasan yang mengatur bahwa rate kedatangan pasien dari suatu penyakit harus sama dengan jumlah rate kedatangan yang telah dipartisi ke berbagai poli Rumah sakit untuk setiap penyakit.

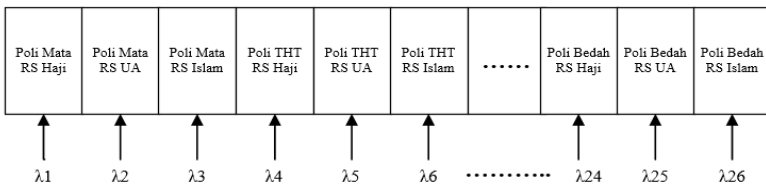
$$\lambda_l = \sum_{k=1}^o \sum_{j=1}^m \lambda_{lkj} \quad \forall l = 1, 2, 3, \dots, p \quad (3.7)$$

3.6 Perancangan MO-GA

Langkah awal yang dilakukan untuk merancang MO-GA ini seperti membuat algoritma genetik pada umumnya seperti inialisasi populasi, seleksi, kawin silang dan mutasi. Setelah algoritma genetik selesai dibuat, kemudian melakukan metode pencarian nilai *fitness* sesuai dengan yang diterapkan pada MO-GA.

Perancangan algoritma genetika diawali dengan mendefinisikan individu yang membawa variabel solusi yang berjumlah 26. Pendefinisian tersebut urut dari poli penyakit mata, THT, paru, kulit, saraf, penyakit dalam, jantung, kandungan dan bedah. Untuk setiap 3 Rumah sakit yaitu RS Haji, RS Universitas Airlangga, dan RS Islam. Ilustrasi pendefinisian individu dapat dilihat pada Gambar 3.13.

Untuk jumlah populasi akan didefinisikan sejumlah 30 individu dengan setiap individunya membawa solusi yang berbeda pula dengan pembangkitan bilangan random kontinyu yang bernilai 0-30.



Gambar 3.13 Perancangan individu algoritma genetika

Kemudian langkah selanjutnya adalah merancang metode seleksi pemilihan individu untuk dikawinkan silang. Metode seleksi yang digunakan dalam tugas akhir ini adalah metode *stochastic universal selection* (SUS). Nilai *fitness* dikumulatikan terlebih dahulu, sehingga

nilai *fitness* yang besar akan memiliki peluang lebih banyak untuk terpilih sebagai *parent*. Namun sebelum itu nilai-nilai *fitness* tersebut dinormalisasi terlebih dahulu agar kumulatif nilai *fitness* tersebut bernilai nol sampai satu. Metode *stochastic universal selection* (SUS) diilustrasikan pada Gambar 3.14.



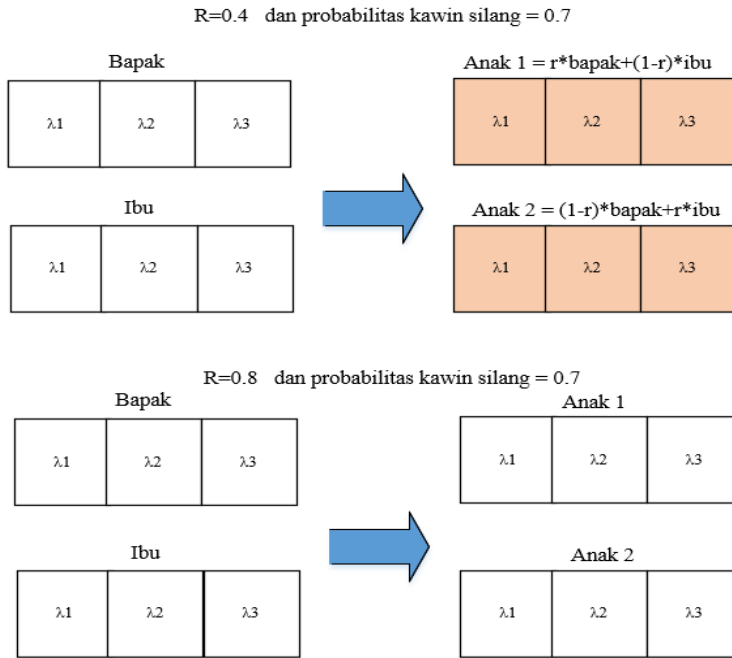
Gambar 3.14 Perancangan metode seleksi dengan SUS

Gambar diatas merupakan ilustrasi SUS. Misalkan dibangkitkan bilangan random. Diasumsikan $r=0.653$, karena bilangan random tersebut terletak antara bilangan 0.62 dan 0.73 sehingga dipilih individu ke-21 untuk menjadi bapak. Kemudian dibangkitkan lagi bilangan random, diasumsikan $r=0.238$, karena bilangan random tersebut terletak antara bilangan 0.18 dan 0.34 sehingga dipilih individu ke-2 untuk menjadi ibu.

Setelah terpilih individu yang akan menjadi *parent* maka akan dilakukan proses kawin silang. Metode kawin silang yang digunakan adalah metode kawin silang aritmatika karena metode ini yang paling sering digunakan untuk menangani permasalahan yang kontinyu. Dengan menentukan probabilitas kawin silang akan didapatkan anak hasil dari perkawinan silang antar dua individu tersebut, yang kemudian anak tersebut akan menggantikan posisi dari bapak dan ibunya. Kawin silang akan terjadi apabila bilangan random yang dibangkitkan kurang dari probabilitas kawin silang yang telah ditentukan ($R < P$. kawin silang). Hasil kawin silang individu atau *offspring* akan menggantikan tempat kedua *parent*, sehingga dapat dikatakan kromosom yang dibawa oleh kedua *parent* akan tergantikan oleh kromosom-kromosom hasil dari kawin silang. Dan apabila tidak terjadi kawin silang ($R > P$. kawin silang) maka kromosom-kromosom kedua *parent* akan tetap seperti semula, seperti pada gambar 3.15.

Setelah itu populasi akan mengalami mutasi, yang banyaknya individu yang di mutasi tergantung probabilitas mutasi. Proses mutasi ini berguna untuk membangkitkan individu-individu yang baru, sehingga

proses kawin silang pada generasi selanjutnya dapat memunculkan individu yang beragam.



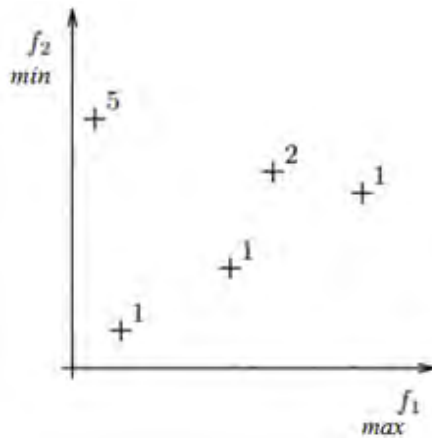
Gambar 3.15 Perancangan *crossover* menggunakan metode aritmatik

Setelah merancang algoritma genetik, kemudian merancang algoritma untuk penugasan nilai *fitness*. Seperti yang telah dijelaskan pada bab sebelumnya, diawali dengan melakukan ranking Pareto terhadap semua individu yang telah melalui proses algoritma genetik. Ranking Pareto ini sedikit diubah konsepnya oleh Fonseca & Fleming pada MO-GA. Jadi dalam metode rankingnya seperti formula dibawah. r_i merupakan ranking yang didapat oleh suatu inividu dengan memperhatikan jumlah individu yang mendominasinya. Dicontohkan pada Tabel 3.1 dan Gambar 3.16.

$$r_i = 1 + n_i \tag{3.8}$$

Tabel 3.1 Contoh ranking Pareto MO-GA

No	F1	F2	Ranking
1	0,4	1,2	1
2	0,38	2,6	5
3	0,5	2,2	1
4	0,45	1,8	1
5	0,47	2,4	2



Gambar 3.16 Contoh ranking Pareto MO-GA

Ada beberapa metode yang digunakan untuk mencari dominansi dari setiap individu. Beberapa diantaranya adalah *Naive and Slow*, *Continously Update*, dan *Kung et al*. Pada perhitungan ini digunakan metode *Naive and Slow* karena fungsi objektif yang digunakan pada sistem antrian ini hanya dua fungsi objektif sehingga dapat menggunakan metode ini dan tidak memakan waktu komputasi yang terlalu lama. Namun apabila fungsi objektif yang digunakan lebih dari dua maka metode *Naive and Slow* menjadi kurang efektif karena waktu komputasinya yang lambat daripada metode lainnya.

Kemudian dari hasil ranking Pareto dari individu-individu tersebut dilakukan penghitungan *average fitness* untuk setiap individunya dengan formula (2.13). Langkah selanjutnya setelah didapatkan nilai *average*

fitness dari setiap individunya adalah menghitung *niche count* dengan formula (2.14). Namun sebelum itu harus terlebih dulu dihitung *euclidian distance* dengan menggunakan formula (2.15). *Euclidian distance* merupakan metrik jarak antar individu. Dalam program MATLAB direpresentasikan dengan sebuah matriks sebesar $N \times N$. Dimana jarak antara $d(i,i) = 0$.

$$\begin{bmatrix} d(1,1) & \cdots & d(1,N) \\ \vdots & \ddots & \vdots \\ d(N,1) & \cdots & d(N,N) \end{bmatrix} \quad (3.9)$$

Setelah mendapatkan jarak antar individu solusi yang telah dihitung dengan formula diatas maka kemudian akan dihitung nilai *sharing function* dari individu dengan formula (2.16). Diasumsikan parameter nilai $\alpha = 1$ dan nilai $\sigma_{share} = 0.5$. Dalam hal ini juga akan direpresentasikan dengan sebuah matriks sebesar $N \times N$. Dimana nilai *sharing function* dari $Sh(i,i) = 1$.

$$\begin{bmatrix} Sh(1,1) & \cdots & Sh(1,N) \\ \vdots & \ddots & \vdots \\ Sh(N,1) & \cdots & Sh(N,N) \end{bmatrix} \quad (3.10)$$

Kemudian dari hasil perhitungan nilai *sharing function* dapat dihitung nilai *niche count* dengan formula (2.14). Perhitungan *niche count* ini dilakukan pada ranking individu yang sama dengan memperhatikan jumlah individu pada ranking yang sama tersebut. Langkah yang terakhir adalah menghitung nilai *shared fitness* dan *scaled fitness*. *Shared fitness* merupakan hasil dari pembagian *average fitness* dengan *niche count* yang berkorespondensi dihitung dengan formula (2.17). Setelah mendapatkan nilai *shared fitness* untuk setiap individu kemudian dihitung nilai *scaled fitness* dari masing-masing individu yang dapat dihitung dengan formula (2.18). Dengan cara ini akan didapatkan nilai *fitness* yang telah diskalarisasi berdasarkan ranking yang didapat.

BAB IV

IMPLEMENTASI DAN ANALISIS

Pada bab ini membahas mengenai implementasi sistem yang telah dirancang sebelumnya, namun sebelum itu dilakukan pengumpulan data-data yang dibutuhkan untuk melakukan implementasi sistem, data-data yang dibutuhkan seperti rate rujukan pasien, rate pelayanan dan jumlah dokter. Sebelum melakukan optimisasi pada sistem antrian rujukan pasien BPJS ini, akan disimulasikan terlebih dahulu kondisi awal dari sistem antrian pasien tersebut menggunakan *software* ARENA. Hal ini bertujuan untuk mengetahui bagaimana performa awal sistem antrian yang belum dioptimalkan. Kemudian dilakukan pengujian terhadap MO-GA yang telah dirancang pada bab sebelumnya untuk menyelesaikan permasalahan optimasi distribusi pasien rujukan berjenjang BPJS. Pengujian pada MO-GA diuji sebanyak 9 kali dengan iterasi yang berbeda dan probabilitas *crossover* yang berbeda sehingga akan mengeluarkan hasil yang berbeda pula. Algoritma tersebut akan mengeluarkan proporsi rujukan untuk menghasilkan nilai fungsi objektif yang optimal dari sistem antrian tersebut dan akan ditampilkan juga solusi *trade-off* yang merepresentasikan hubungan antara fungsi objektif 1 dan 2. Dari kesembilan pengujian algoritma akan diambil sebuah solusi yang dapat mengeluarkan nilai fungsi *fitness* 2 yang paling kecil. Setelah itu proporsi rujukan yang telah didapat akan disimulasikan dua titik solusi dengan program ARENA. Setelah melakukan simulasi dengan proporsi rujukan yang telah dihitung agar mencapai nilai fungsi objektif yang optimal, kemudian dilakukan analisa hasil dari simulasi kondisi awal sebelum dioptimalkan dan simulasi kondisi yang telah dioptimalkan.

4.1 Pengumpulan dan Pengolahan Data

Untuk menerapkan rancangan sistem yang telah dibuat, diperlukan sejumlah data diantara lain rate rujukan dari Puskesmas dan per wilayah Surabaya. Proporsi terjadinya penyakit di setiap Puskesmas dan per wilayah Surabaya, proporsi terjadinya penyakit ini merupakan banyaknya kejadian dari sembilan penyakit yang telah ditentukan sebelumnya. Selanjutnya adalah data rate pelayanan dari setiap poli penyakit di ketiga Rumah sakit. Untuk mempermudah perhitungan rate rujukan akan dihitung terlebih dahulu agregasi rate rujukan setiap penyakit sejenis. Hal ini dilakukan agar sesuai dengan konsep sistem yang telah dirancang.

Tabel 4.1 Rate Kedatangan (jam) dari puskesmas dan wilayah

Puskesmas	λ
Gading	2,3
Gunung Anyar	1,32
Kalijudan	0,82
Kalirungkut	1,79
Keputih	1,02
Klampis Ngasem	1,08
Medokan Ayu	1,62
Menur	1,51
Mojo	2,45
Mulyorejo	1,33
Pacar Keling	2,27
Rangkah	2,28
Tenggilis	1,39
Surabaya Pusat	20,41
Surabaya Barat	24,1
Surabaya Selatan	27,21
Surabaya Utara	35,61
Surabaya Timur	18,34

Tabel 4.1 merupakan rate rujukan (λ) dari Puskesmas dan per wilayah Surabaya sesuai dengan skema Gambar 3.4. Untuk melihat jumlah penyakit yang terjadi di setiap puskesmas dan wilayah Surabaya. Maka perlu diketahui juga proporsi terjadinya penyakit (P_{np}) yang terjadi di setiap fasilitas kesehatan 1 se-Surabaya yang direpresentasikan pada Tabel 4.2 sesuai dengan skema Gambar 3.4.

Tabel 4.2 Proporsi penyakit setiap Puskesmas (%)

Penyakit	Mata	THT	Paru	Kulit	Saraf	P. Dalam	Jantung	Kandungan	Bedah
Gading	26,9	8,6	4,4	5,1	0,5	32,6	1,5	3,1	16,5
G. Anyar	13,8	11,3	3,2	1,8	0,8	26	1,7	1,6	38,2
Kalijudan	16,1	7,9	3,3	2,8	1,1	14,4	2,2	1,6	50,6
Kalirungkut	17,1	6,2	4,3	4	0,6	25,9	1,9	1,6	38
Keputih	20,1	9,6	6,3	4,7	0,7	15,7	0,9	0,8	40,8
Klampis	17,3	7	3,8	1,6	0,6	22,5	1,9	2,1	43,2
Medokan	17,1	7,2	3,2	2	0,8	15,6	1,3	1,8	50,8
Menur	15,2	6,2	4,1	1,8	1	25,5	0,9	2,2	42,6
Mojo	12,1	5,4	2,1	2,2	0,2	23,2	0,6	16,5	33,5
Mulyorejo	25,8	8,7	2,7	4,2	0,4	26,6	0,9	2	28
P. Keling	26,6	4,2	2,8	3,8	0,6	18,3	1,2	3,8	37,5
Rangkah	9,7	3,1	1,9	1,8	0,3	19,3	2,5	1	60,2
Tenggilis	15,6	4,1	5,3	2	0,8	43,2	1,7	2,7	24,6
Surabaya Pusat	31,9	5	2,9	2,5	0,7	29,1	1,8	2,2	24
Surabaya Barat	12,2	9,7	3,4	4,2	1,2	22,1	1,9	2,5	42,8
Surabaya Selatan	8,6	11,4	3,5	2,9	0,8	21,4	2,1	2,3	47
Surabaya Timur	19,5	9,5	3,7	1,6	0,7	19,8	1,9	1,7	41,6
Surabaya Utara	4,2	4,1	5	1,3	1,3	16,7	1,6	1,4	64,5

Setelah melihat proporsi penyakit pada Tabel 4.2, maka perlu untuk menjumlah atau mengagregasi rate rujukan setiap penyakit dan memperhatikan rate pelayanan dari setiap poli Rumah sakit tersebut dan

jumlah dokter yang dapat melayani di setiap poli Rumah sakit. Rate pelayanan dari setiap poli yang sama di Rumah sakit yang berbeda terdapat perbedaan rate pelayanan dan jumlah dokter yang berbeda. Hal ini sesuai dengan skema Gambar 3.4.

Tabel 4.3 Agregasi rate kedatangan (per jam) setiap penyakit

Penyakit	λ'
Mata	20,85
THT	11,12
Paru	5,58
Kulit	3,7
Saraf	1,38
P. Dalam	31,62
Jantung	3,37
Kandungan	3,15
Bedah	66,72

Tabel 4.4 Rate pelayanan (per jam) dan jumlah dokter tiap poli

Poli	RS Haji		RS Islam		RS UA	
	μ	c	μ	c	μ	c
Mata	9,62	2	9,57	1	4,89	2
THT	10,64	2	11,43	1	4,22	2
Paru	7,96	1	8,74	1	4,1	3
Kulit	9,48	2	10,4	1	6,21	2
Saraf	12,48	2	10,64	1	2,54	2
P. Dalam	13,09	2	13,77	2	3,64	3
Jantung	13,92	2	13,33	1	6,85	2
Kandungan	7,8	1	8,32	1	-	-
Bedah	6,42	4	7,86	2	4,21	8

4.2 Simulasi Kondisi Awal

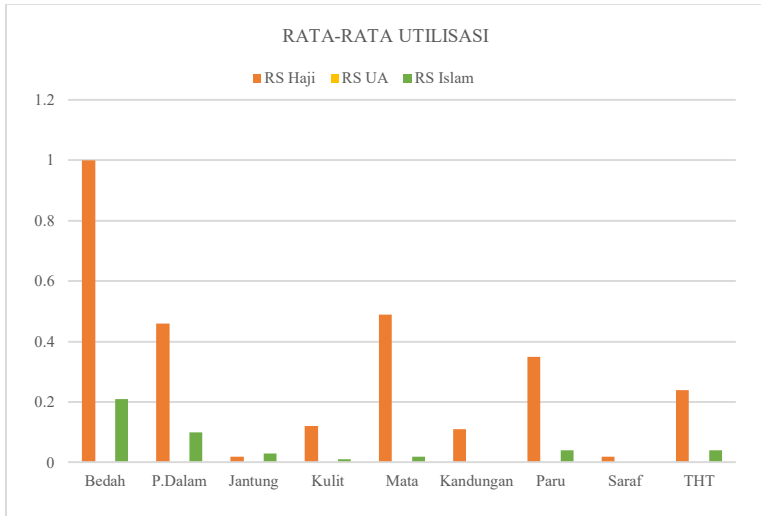
Setelah mengetahui data-data yang diperlukan, kemudian mengimplementasikan data-data tersebut kedalam model sistem jaringan antrian yang telah dirancang pada bab sebelumnya mengacu pada skema Gambar 3.4. Data-data rate rujukan pasien dan rate pelayanan beserta jumlah dokter yang ada di setiap poli penyakit di ketiga Rumah sakit tersebut akan menjadi parameter yang akan dijadikan parameter pada modul *create* mengacu pada Gambar 3.5 sebagai representasi rate kedatangan, modul *decide* mengacu pada Gambar 3.8 sebagai representasi proporsi penyakit dan rujukan, dan modul *process* mengacu pada Gambar 3.10 sebagai representasi dari rate pelayanan poli dan jumlah dokter.

Simulasi kondisi awal merupakan salah satu hal yang perlu dilakukan untuk mengetahui peforma awal sistem antrian sebelum dioptimalkan. Proporsi disini berfungsi sebagai input atau variabel yang dapat diubah-ubah untuk mendapatkan nilai fungsi objektif yang optimal Dengan proporsi rujukan pada kondisi awal dan hasil simulasi ARENA sebagai berikut.

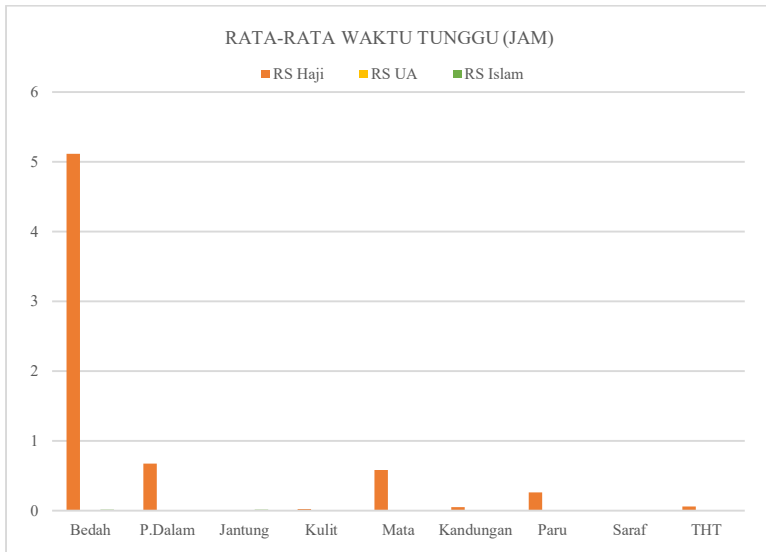
Tabel 4.5 Proporsi rujukan kondisi awal

Poli	Proporsi (%)		
	RS Haji	RS UA	RS Islam
Mata	99,59	0	0,41
THT	96,83	0	3,17
Paru	94,12	0	5,88
Kulit	92,94	0	7,06
Saraf	92,31	0	7,69
P.Dalam	79,94	0	20
Jantung	69,62	0	30,38
Kandungan	95,71	-	4,29
Bedah	86,13	0	13,85

Dengan proporsi rujukan (P_{kj}) kondisi awal pada Tabel 4.5, maka dihasilkan output yang akan dianalisa yaitu berupa nilai rata-rata utilisasi dan nilai rata-rata waktu tunggu seperti Gambar 4.1 dan 4.2.

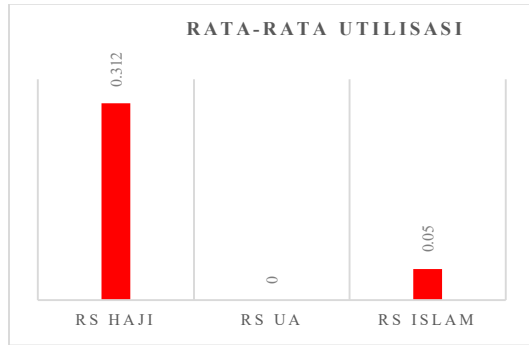


Gambar 4.1 Utilisasi setiap poli di setiap Rumah sakit kondisi awal

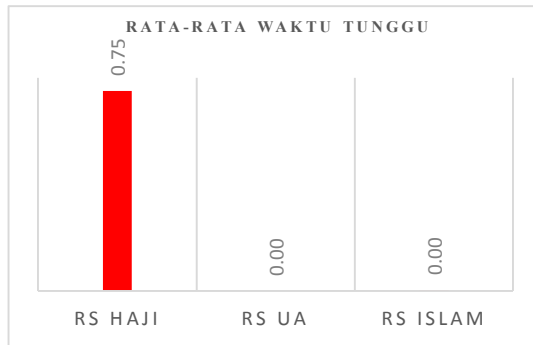


Gambar 4.2 Waktu tunggu setiap poli di setiap Rumah sakit kondisi awal

Dengan hasil pada Gambar 4.1 dan 4.2 dapat juga diketahui nilai rata-rata utilisasi dan rata-rata waktu tunggu untuk setiap Rumah sakit yang direpresentasikan oleh grafik pada Gambar 4.3 dan 4.4.



Gambar 4.3 Utilisasi kondisi awal setiap RS



Gambar 4.4 Waktu tunggu kondisi awal tiap RS

Dapat dilihat pada hasil simulasi di atas, bahwa proporsi rujukan pada kondisi awal menyebabkan sistem antrian tidak optimal. Dikarenakan nilai utilisasi dari poli penyakit bedah di Rumah sakit Haji sangat tinggi melebihi satu jika dihitung secara deterministik. Sehingga waktu tunggu di poli penyakit tersebut menjadi sangat tinggi. Hal ini yang membuat performa sistem antrian menjadi buruk karena ada server yang *overload*.

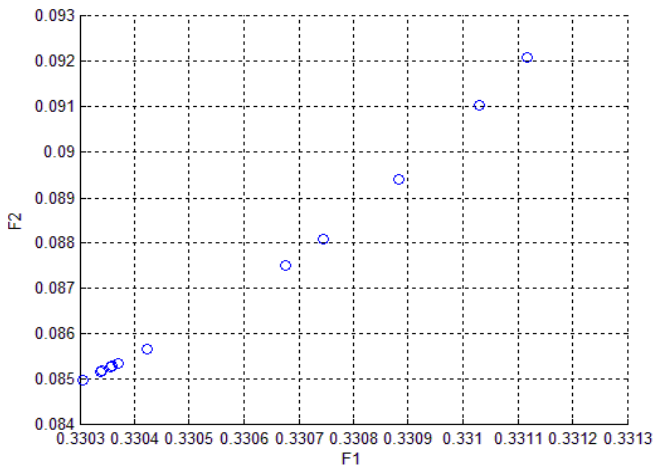
Dan proporsi rujukan ke Rumah sakit Haji sangat tinggi jika dibandingkan dengan Rumah sakit yang lain. Hal ini tentu menyebabkan nilai rata-rata utilisasi dari Rumah sakit haji berbeda jauh dengan Rumah sakit lainnya yaitu sebesar 0,312. Sehubungan dengan nilai rata-rata utilisasi Rumah sakit Haji yang tinggi maka nilai rata-rata waktu tunggu juga tinggi yaitu sebesar 0,75 jam. Kondisi seperti ini perlu untuk diperbaiki, hal yang perlu diperbaiki adalah proporsi rujukan ke tiga rumah sakit tersebut, yang akan dihitung menggunakan MO-GA.

4.3 Penerapan MO-GA

4.3.1 Pengujian MO-GA

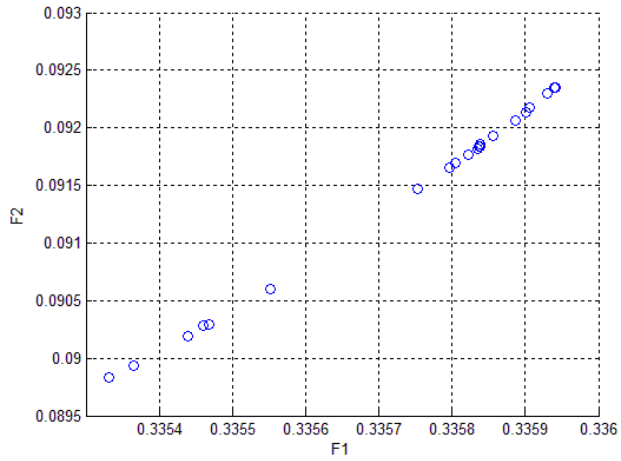
Dalam pengujian MO-GA ini akan diuji sebanyak sembilan kali dengan jumlah iterasi yang berbeda yaitu 100, 300, dan 500. Dengan mengubah probabilitas *crossover* yaitu 0,5 ; 0,7 ; dan 0,9. Dengan probabilitas mutasi sebesar 0,1 dan jumlah populasi sebanyak 30 serta variabel yang dihitung sebanyak 26. Kemudian akan dibandingkan hasil perhitungan algoritma tersebut dan yang menghasilkan nilai fungsi *fitness* 2 paling baik akan dipilih untuk disimulasikan pada program ARENA.

- a. Probabilitas *crossover* = 0,5 dan iterasi = 100



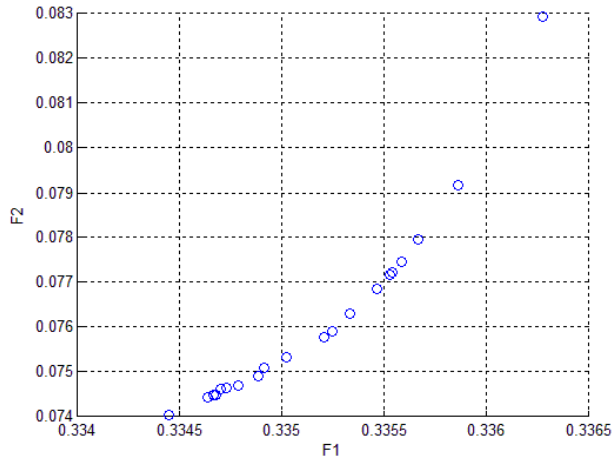
Gambar 4.5 Probabilitas *crossover* = 0,5 dan iterasi = 100

b. Probabilitas *crossover* = 0,5 dan iterasi = 300



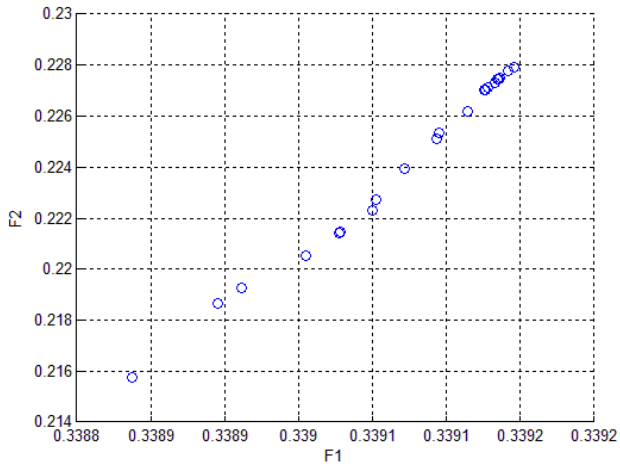
Gambar 4.6 Probabilitas *crossover* = 0,5 dan iterasi = 300

c. Probabilitas *crossover* = 0,5 dan iterasi = 500



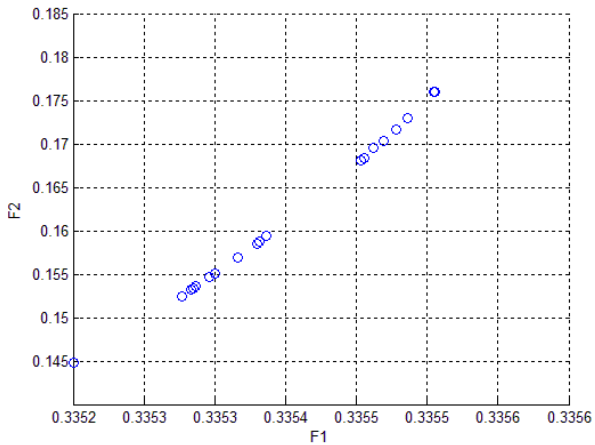
Gambar 4.7 Probabilitas *crossover* = 0,5 dan iterasi = 500

d. Probabilitas *crossover* = 0,7 dan iterasi = 100



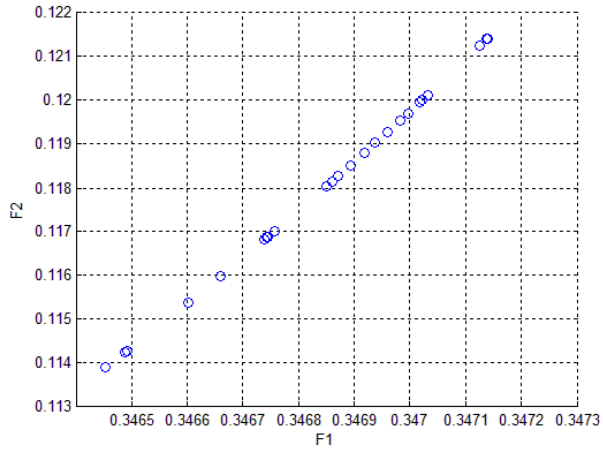
Gambar 4.8 Probabilitas *crossover* = 0,7 dan iterasi = 100

e. Probabilitas *crossover* = 0,7 dan iterasi = 300



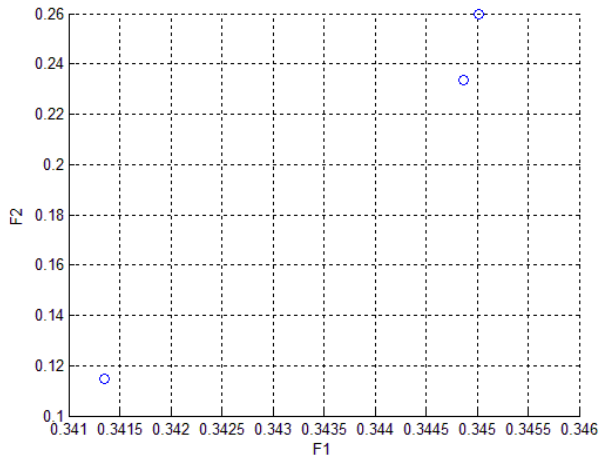
Gambar 4.9 Probabilitas *crossover* = 0,7 dan iterasi = 300

f. Probabilitas *crossover* = 0,7 dan iterasi = 500



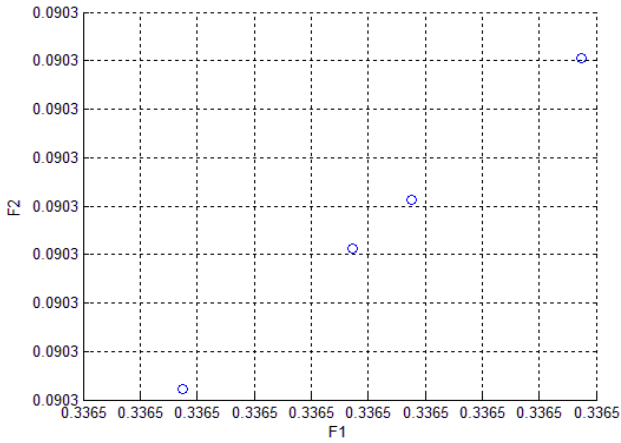
Gambar 4.10 Probabilitas *crossover* = 0,7 dan iterasi = 500

g. Probabilitas *crossover* = 0,9 dan iterasi = 100



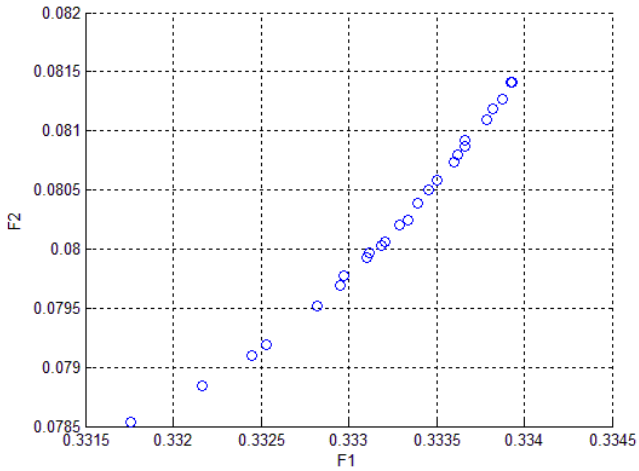
Gambar 4.11 Probabilitas *crossover* = 0,9 dan iterasi = 100

h. Probabilitas *crossover* = 0,9 dan iterasi = 300



Gambar 4.12 Probabilitas *crossover* = 0,9 dan iterasi = 300

i. Probabilitas *crossover* = 0,9 dan iterasi = 500



Gambar 4.13 Probabilitas *crossover* = 0,9 dan iterasi = 500

Setelah melakukan sembilan kali pengujian MO-GA dengan probabilitas *crossover* dan iterasi yang berbeda selalu mengeluarkan hasil yang berbeda pula. Seperti yang dapat dilihat pada hasil pengujian dengan probabilitas *crossover* yang semakin kecil pada umumnya dapat menemukan hasil solusi yang optimal dimana dapat menemukan nilai fungsi *fitness* 2 terkecil dan nilai fungsi *fitness* 1 terbesar dalam Pareto set. Hal ini dikarenakan probabilitas *crossover* yang kecil akan semakin kecil pula peluang individu-individu terbaik yang melakukan kawin silang. Sehingga nilai variabel yang telah ditemukan tidak teracak kembali.

Telihat pada percobaan dengan probabilitas *crossover* 0,5 dengan iterasi 100 terlihat bahwa *non-dominated solutions* yang ditemukan yaitu 26 solusi. Pada percobaan dengan iterasi 300 terlihat bahwa *non-dominated solutions* yang ditemukan yaitu 28 solusi serta ada perbaikan untuk nilai fungsi *fitness* 2 yang semakin minimal. Pada percobaan dengan iterasi 500 terlihat bahwa *non-dominated solutions* yang ditemukan semakin banyak yaitu 21 solusi. Walaupun pada percobaan dengan iterasi 500 ini menemukan jumlah *non-dominated solutions* yang lebih sedikit namun dapat menemukan nilai fungsi *fitness* 2 yang paling minimal pada percobaan dengan probabilitas *crossover* 0,5.

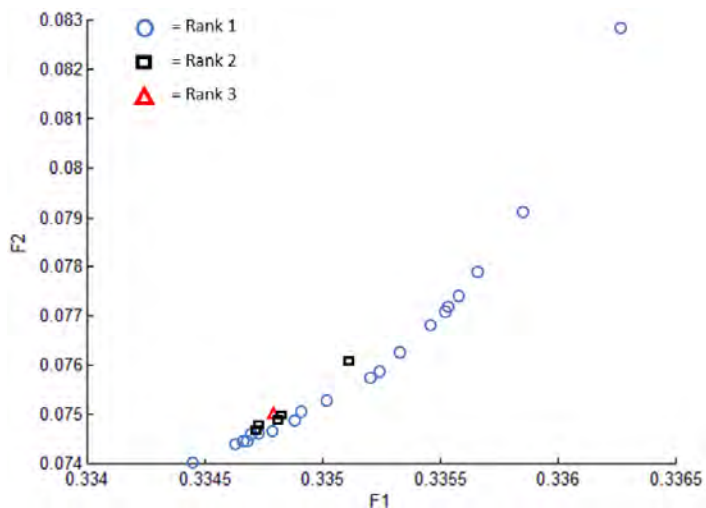
Pada percobaan dengan probabilitas *crossover* 0,7 dengan iterasi 100 terlihat bahwa *non-dominated solutions* yang ditemukan yaitu 22 solusi. Pada percobaan dengan iterasi 300 terlihat bahwa *non-dominated solutions* yang ditemukan yaitu 24 solusi serta ada perbaikan untuk nilai fungsi *fitness* 2 yang semakin minimal. Pada percobaan dengan iterasi 500 terlihat bahwa *non-dominated solutions* yang ditemukan semakin banyak yaitu 25 solusi. Dan pada percobaan dengan iterasi 500 ini dapat menemukan nilai fungsi *fitness* 2 yang paling minimal pada percobaan dengan probabilitas *crossover* 0,7.

Pada percobaan dengan probabilitas *crossover* 0,9 dengan iterasi 100 terlihat bahwa *non-dominated solutions* yang ditemukan sedikit yaitu 3 solusi. Pada percobaan dengan iterasi 300 terlihat bahwa *non-dominated solutions* yang ditemukan sedikit yaitu 4 solusi namun ada perbaikan untuk nilai fungsi *fitness* 2 yang semakin minimal. Pada percobaan dengan iterasi 500 terlihat bahwa *non-dominated solutions* yang ditemukan semakin banyak yaitu 25 solusi. Dan pada percobaan dengan iterasi 500 ini dapat menemukan nilai fungsi *fitness* 2 yang paling minimal pada percobaan dengan probabilitas *crossover* 0,9.

Tabel 4.6 Perbandingan hasil pengujian

Probabilitas <i>crossover</i>	Iterasi	F1	F2
0.5	100	0,3303	0,085
	300	0,3353	0,089
	500	0,3344	0,074
0.7	100	0,33888	0,215
	300	0,3352	0,145
	500	0,34645	0,1139
0.9	100	0,3414	0,118
	300	0,3365	0,09
	500	0,3339	0,0785

Tabel 4.8 menampilkan hasil-hasil pengujian MO-GA untuk probabilitas 0,5; 0,7 dan 0,9. Untuk setiap iterasi 100, 300 dan 500. Hasil pada tabel 4.8 menampilkan nilai fungsi *fitness* 2 yang paling minimal, karena untuk permasalahan ini dicari sebuah hasil pengujian yang dapat menemukan nilai fungsi *fitness* 2 paling minimal. Walaupun tidak semua percobaan dengan iterasi yang lebih banyak akan menghasilkan solusi yang baik. Namun pada umumnya iterasi yang semakin banyak dapat menemukan hasil solusi yang lebih baik daripada percobaan dengan iterasi yang sedikit. Pada percobaan dengan probabilitas *crossover* sebesar 0,5 dan pada iterasi ke-500 MO-GA dapat menemukan hasil fungsi *fitness* yang baik yaitu dapat meminimalkan fungsi *fitness* 2 seminimal mungkin jika dibandingkan dengan percobaan dengan probabilitas *crossover* dan iterasi yang lain. Percobaan dengan probabilitas *crossover* 0,5 dengan iterasi sebanyak 500 dapat menemukan fungsi *fitness* 1 sebesar 0,3363 dan fungsi *fitness* 2 sebesar 0,0829. Dan juga menemukan fungsi *fitness* 1 sebesar 0,3344 dan nilai fungsi *fitness* 2 sebesar 0,074. Nilai fungsi *fitness* tersebut dapat dilihat pada Tabel 4.9 individu nomor 16 dan individu nomor 22.



Gambar 4.14 Ranking Pareto

Tabel 4.7 Hasil ranking Pareto pada percobaan C

Solusi	F1	F2	R	Solusi	F1	F2	R
1	0,3348	0,074963	3	16	0,336276	0,082937	1
2	0,334669	0,074488	1	17	0,335584	0,077471	1
3	0,335665	0,077959	1	18	0,335226	0,075971	2
4	0,334788	0,074698	1	19	0,334911	0,075099	1
5	0,335861	0,079157	1	20	0,334669	0,074488	1
6	0,334853	0,074967	2	21	0,334634	0,074429	1
7	0,334730	0,074639	1	22	0,334442	0,074018	1
8	0,334834	0,074918	2	23	0,335211	0,075785	1
9	0,334723	0,074689	2	24	0,335238	0,075917	1
10	0,3347	0,074614	1	25	-13,9316	14,30523	29
11	0,335538	0,077221	1	26	0,335334	0,076304	1
12	0,33553	0,077171	1	27	0,334738	0,074721	2
13	0,334880	0,074888	1	28	-4,42592	4,77973	28
14	0,334673	0,074490	1	29	0,335465	0,07686	1
15	0,335015	0,075321	1	30	-32,47332	32,8187	30

4.3.2 Penentuan Variabel Keputusan

Solusi *trade-off* yang dipilih adalah pada percobaan dengan probabilitas *crossover* 0,5 dan iterasi 500 yang dimana dapat menemukan nilai *fitness* 2 yang terkecil daripada percobaan lainnya. Kemudian dipilih dua titik optimal yang akan digunakan sebagai variabel proporsi rujukan, yaitu solusi nomor 16 dan 22. Hal ini bertujuan untuk melihat perbedaan hasil antara solusi yang menghasilkan nilai rata-rata utilisasi paling maksimum dan nilai rata-rata waktu tunggu paling minimum. Variabel rate kedatangan pada Tabel 4.10 dan 4.11 melambangkan $P_{tkj}\lambda'_j$ pada skema Gambar 3.4.

Tabel 4.8 Rate kedatangan solusi optimal 1

Poli	Rate kedatangan (jam)		
	RS Haji	RS UA	RS Islam
Mata	6,36	7,27	7,22
THT	3,79	2,15	5,19
Paru	2,09	1,38	2,1
Kulit	1,08	1,43	1,19
Saraf	0,3	0,57	0,5
P.Dalam	10,73	13,17	7,72
Jantung	0,63	1,28	1,47
Kandungaan	1,72	-	1,42
Bedah	22,47	13,15	31,09

Tabel 4.9 Rate kedatangan solusi optimal 2

Poli	Rate kedatangan (jam)		
	RS Haji	RS UA	RS Islam
Mata	6,27	7,4	7,18
THT	3,94	2,39	4,79
Paru	2,1	1,43	2,05
Kulit	1,1	1,38	1,22
Saraf	0,35	0,52	0,5
P.Dalam	9,79	13,59	8,24
Jantung	0,67	1,28	1,42
Kandungaan	1,73	-	1,41
Bedah	22,02	14,24	30,46

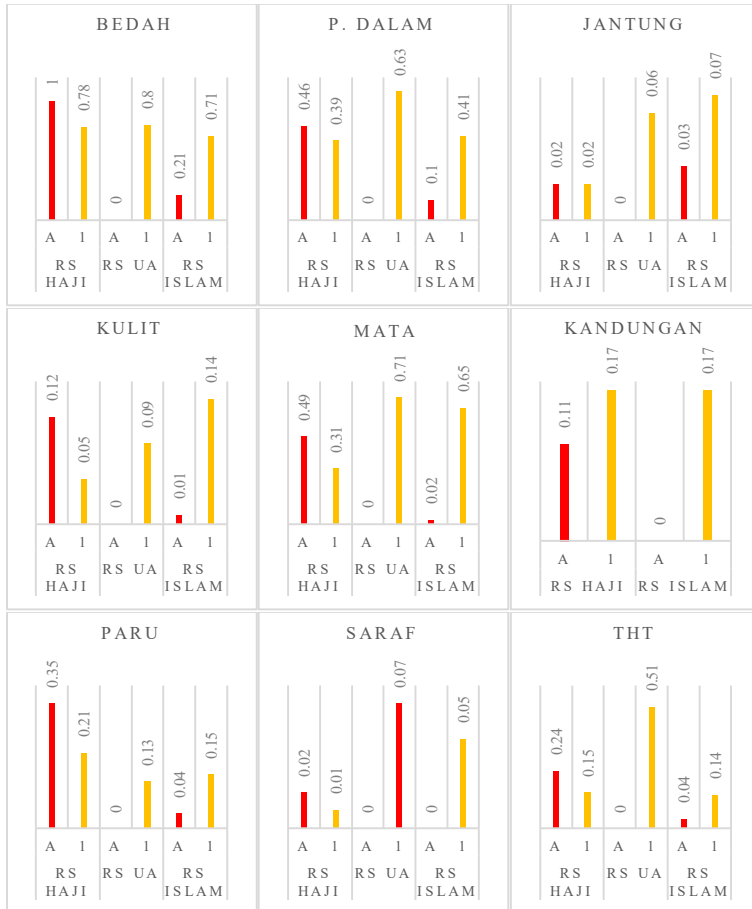
4.3 Simulasi dengan Solusi Optimal 1

Simulasi dengan solusi optimal 1 merupakan pengujian simulasi dengan program ARENA untuk menguji solusi optimal 1. Solusi optimal 1 merupakan solusi dengan nilai *fitness* 2 terkecil yaitu meminimalkan rata-rata waktu tunggu yang dimana nilai rata-rata utilisasinya juga lebih rendah daripada solusi optimal 2. Simulasi ini dilakukan dengan menggunakan solusi yang dibawa oleh individu nomor 22 pada Tabel 4.7 dan memperhatikan rate kedatangan pada Tabel 4.8 maka secara detail proporsi rujukan dari tiap penyakit ke setiap poli sejenis pada 3 Rumah sakit ditunjukkan pada Tabel 4.10.

Tabel 4.10 Proporsi rujukan optimal 1

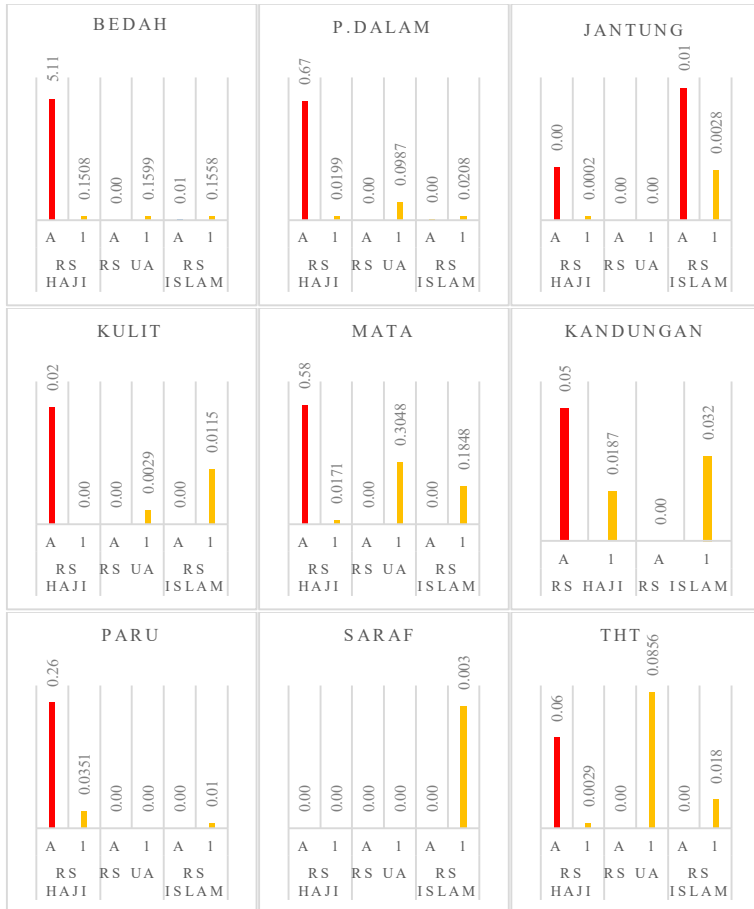
Poli	Proporsi (%)		
	RS Haji	RS UA	RS Islam
Mata	30,61	34,98	34,41
THT	33,86	19,97	46,17
Paru	37,55	24,53	37,91
Kulit	29,08	38,58	32,34
Saraf	22,23	41,39	36,38
P.Dalam	33,94	41,72	24,34
Jantung	18,83	37,73	43,44
Kandungan	55,12	-	44,88
Bedah	33,83	19,58	46,58

Dengan proporsi rujukan (P_{ikj}) seperti yang dapat dilihat pada Tabel 4.10 menghasilkan rate kedatangan seperti Tabel 4.8. Setelah disimulasikan menghasilkan nilai rata-rata utilisasi dan nilai rata-rata waktu tunggu untuk setiap poli di setiap Rumah sakit beserta perbandingan dengan kondisi awal seperti grafik Gambar 4.15.



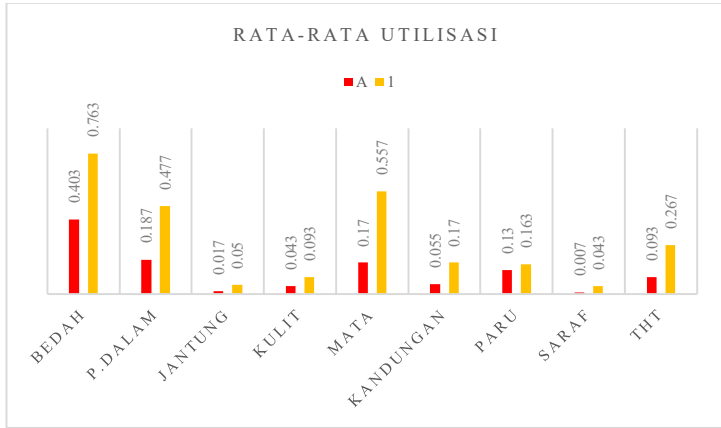
Gambar 4.15 Perbandingan utilisasi setiap poli (1)

Terlihat pada grafik Gambar 4.15 nilai rata-rata utilisasi pada hampir seluruh poli di Rumah sakit Haji menurun. Hal ini dikarenakan terjadi perpindahan pasien yang awalnya menumpuk pada Rumah sakit Haji saja. Pada kondisi optimal 1 pasien telah didistribusikan ke Rumah sakit Islam dan Rumah sakit Universitas Airlangga. Kondisi tersebut tentu mengakibatkan peningkatan nilai rata-rata utilisasi di seluruh poli Rumah sakit Islam dan Rumah sakit Universitas Airlangga.

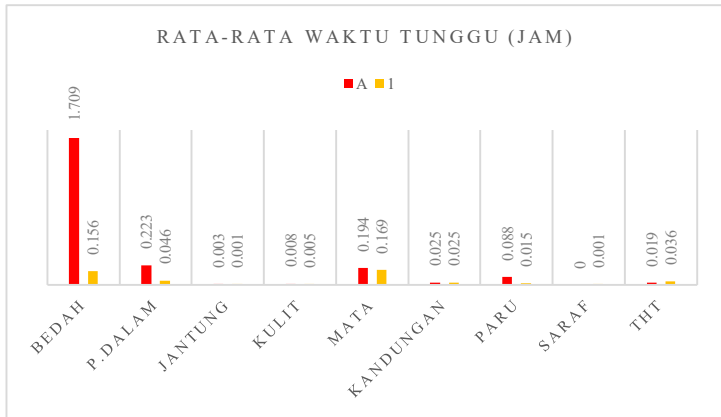


Gambar 4.16 Perbandingan waktu tunggu setiap poli (1)

Terlihat pada grafik Gambar 4.16 nilai rata-rata waktu tunggu pada hampir seluruh poli di Rumah sakit Haji menurun. Hal ini dikarenakan terjadi penurunan nilai utilisasi pada Rumah sakit Haji, sehingga rata-rata waktu tunggu pasien juga menurun. Kondisi ini berkebalikan dengan Rumah sakit Islam dan Rumah sakit Universitas Airlangga yang waktu tenggunya cenderung naik diakibatkan oleh meningkatnya utilisasi.

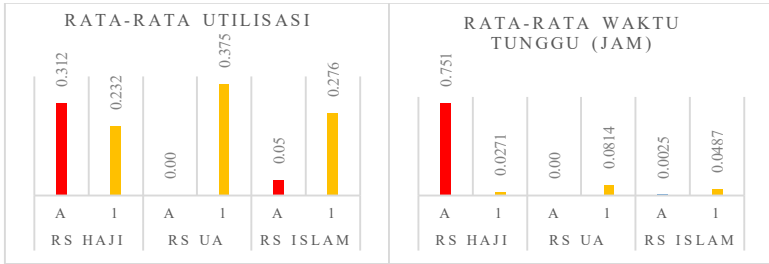


Gambar 4.17 Perbandingan utilisasi poli sejenis di semua RS (1)



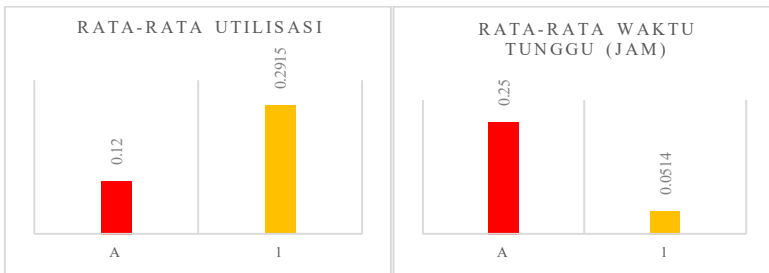
Gambar 4.18 Perbandingan waktu tunggu poli sejenis di semua RS (1)

Gambar 4.17 dan 4.18 merupakan rata-rata utilisasi dan rata waktu tunggu di poli sejenis di semua Rumah sakit. Dapat dilihat bahwa kondisi optimal 1 lebih baik dari kondisi awal, nilai rata-rata utilisasi meningkat untuk seluruh poli sejenis di semua Rumah sakit disebabkan oleh persebaran pasien yang semakin merata. Hal ini juga menurunkan rata-rata waktu tunggu di sebagian besar poli sejenis di semua Rumah sakit.



Gambar 4.19 Perbandingan utilisasi & waktu tunggu setiap RS (1)

Pada grafik Gambar 4.19 merepresentasikan perbandingan nilai rata-rata utilisasi dan nilai rata-rata waktu tunggu untuk setiap Rumah sakit. Terlihat nilai rata-rata utilisasi di Rumah sakit Haji menurun. dikarenakan terjadi perpindahan pasien dari Rumah sakit Haji ke Rumah sakit Islam dan Rumah sakit Universitas Airlangga. Perpindahan pasien tersebut tentu meningkatkan rata-rata utilisasi di Rumah sakit Islam dan Rumah sakit Universitas Airlangga. Namun dengan perpindahan pasien dari Rumah sakit Haji ke Rumah sakit Islam dan Rumah sakit Universitas Airlangga, menyebabkan meningkatnya rata-rata waktu tunggu di kedua Rumah sakit tersebut. Dan sebaliknya menurunkan rata-rata waktu tunggu di Rumah sakit Haji. Perbandingan nilai rata-rata utilisasi dan rata-rata waktu tunggu untuk seluruh poli di semua Rumah sakit dapat dilihat pada Gambar 4.20. Terlihat nilai rata-rata utilisasi mengalami perbaikan daripada kondisi awal, disebabkan oleh persebaran pasien yang semakin merata sehingga tidak menyebabkan *overload* di poli Rumah sakit tertentu. Dan untuk nilai rata-rata waktu juga terlihat semakin baik, hal ini dikarenakan pasien tidak menumpuk di poli Rumah sakit tertentu.



Gambar 4.20 Perbandingan utilisasi & waktu tunggu keseluruhan (1)

4.4 Simulasi dengan Solusi Optimal 2

Simulasi dengan solusi optimal 2 merupakan pengujian simulasi dengan program ARENA untuk menguji solusi optimal 2. Solusi optimal 2 merupakan solusi dengan nilai *fitness* 1 terbesar yaitu memaksimalkan rata-rata utilisasi yang dimana nilai rata-rata waktu tunggu juga lebih tinggi daripada solusi optimal 1. Simulasi ini menggunakan solusi yang dibawa oleh individu nomor 16 pada Tabel 4.7 dan memperhatikan rate kedatangan pada Tabel 4.9. Secara detail proporsi rujukan dari tiap penyakit ke setiap poli sejenis pada 3 Rumah sakit dapat dilihat pada Tabel 4.11.

Tabel 4.11 Proporsi rujukan optimal 2

Poli	Proporsi (%)		
	RS Haji	RS UA	RS Islam
Mata	30,08	35,5	34,41
THT	35,44	21,51	43,06
Paru	37,71	25,59	36,7
Kulit	29,74	37,27	32,99
Saraf	25,56	41,32	36,48
P.Dalam	33,92	41,68	24,4
Jantung	18,67	37,84	43,49
Kandungan	54,78	-	45,22
Bedah	33,7	19,7	46,6

Dengan proporsi rujukan (P_{ikj}) seperti yang dapat dilihat pada Tabel 4.11 menghasilkan rate kedatangan seperti Tabel 4.9. Setelah disimulasikan menghasilkan nilai rata-rata utilisasi dan nilai rata-rata waktu tunggu untuk setiap poli di setiap Rumah sakit beserta perbandingan dengan kondisi awal seperti grafik Gambar 4.21.



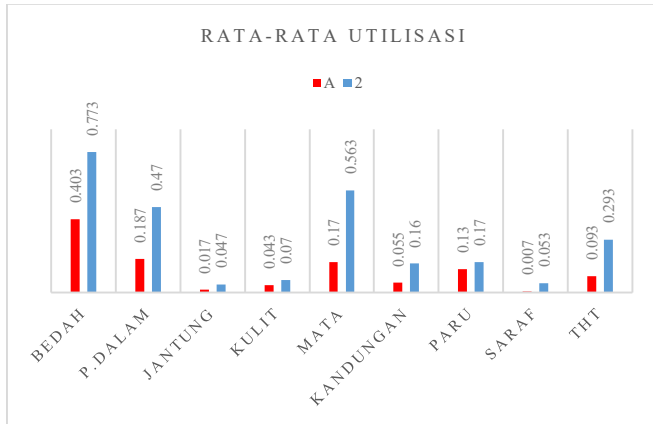
Gambar 4.21 Perbandingan utilisasi setiap poli (2)

Terlihat pada grafik Gambar 4.21 nilai rata-rata utilisasi pada hampir seluruh poli di Rumah sakit Haji menurun. Hal ini dikarenakan terjadi perpindahan pasien yang awalnya menumpuk pada Rumah sakit Haji saja. Pada kondisi optimal 2 pasien telah didistribusikan ke Rumah sakit Islam dan Rumah sakit Universitas Airlangga. Kondisi tersebut tentu mengakibatkan peningkatan nilai rata-rata utilisasi di seluruh poli Rumah sakit Islam dan Rumah sakit Universitas Airlangga.

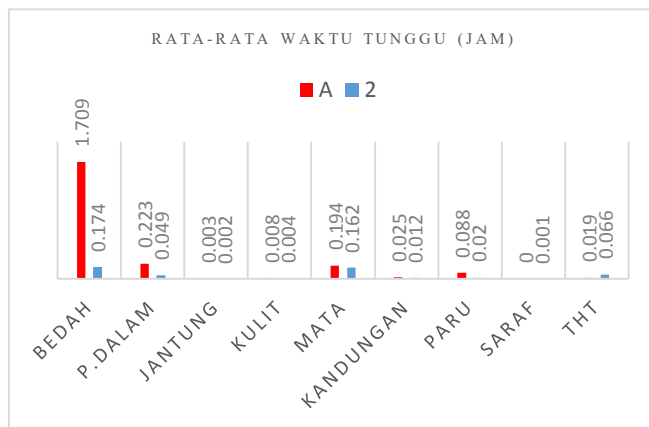


Gambar 4.22 Perbandingan waktu tunggu setiap poli (2)

Terlihat pada grafik Gambar 4.22 nilai rata-rata waktu tunggu pada hampir seluruh poli di Rumah sakit Haji menurun. Hal ini dikarenakan terjadi penurunan nilai utilisasi pada Rumah sakit Haji, sehingga rata-rata waktu tunggu pasien juga menurun. Kondisi ini berkebalikan dengan Rumah sakit Islam dan Rumah sakit Universitas Airlangga yang waktu tunggunya naik diakibatkan oleh meningkatnya utilisasi.

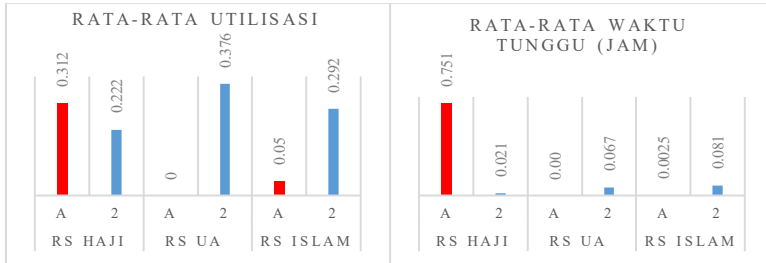


Gambar 4.23 Perbandingan utilisasi poli sejenis di semua RS (2)



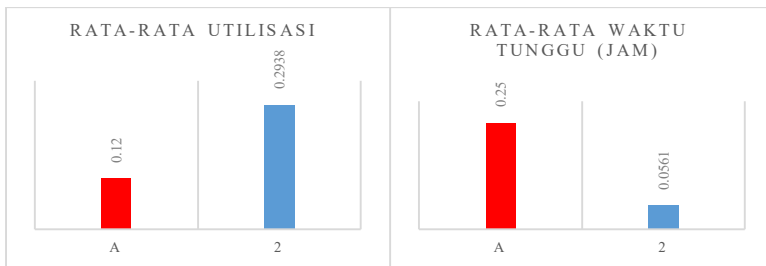
Gambar 4.24 Perbandingan waktu tunggu poli sejenis di semua RS (2)

Gambar 4.23 dan 4.24 merupakan rata-rata utilisasi dan rata waktu tunggu di setiap poli di semua Rumah sakit. Dapat dilihat bahwa kondisi optimal 2 lebih baik dari kondisi awal, nilai rata-rata utilisasi meningkat untuk seluruh poli sejenis di semua Rumah sakit disebabkan oleh persebaran pasien yang semakin merata. Hal ini juga menurunkan rata-rata waktu tunggu di sebagian besar poli sejenis di semua Rumah sakit



Gambar 4.25 Perbandingan utilisasi & waktu tunggu setiap RS (2)

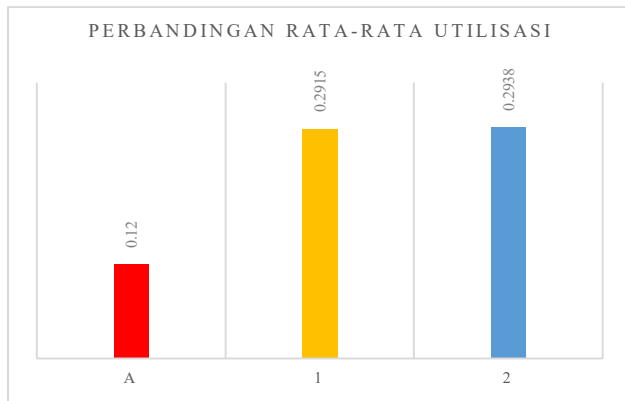
Pada grafik Gambar 4.25 merepresentasikan perbandingan nilai rata-rata utilisasi dan nilai rata-rata waktu tunggu untuk setiap Rumah sakit. Terlihat nilai rata-rata utilisasi di Rumah sakit Haji menurun. dikarenakan terjadi perpindahan pasien dari Rumah sakit Haji ke Rumah sakit Islam dan Rumah sakit Universitas Airlangga. Perpindahan pasien tersebut tentu meningkatkan rata-rata utilisasi di Rumah sakit Islam dan Rumah sakit Universitas Airlangga. Namun dengan perpindahan pasien dari Rumah sakit Haji ke Rumah sakit Islam dan Rumah sakit Universitas Airlangga, menyebabkan meningkatnya rata-rata waktu tunggu di kedua Rumah sakit tersebut. Dan sebaliknya menurunkan rata-rata waktu tunggu di Rumah sakit Haji. Perbandingan nilai rata-rata utilisasi dan rata-rata waktu tunggu untuk seluruh poli di semua Rumah sakit dapat dilihat pada Gambar 4.26. Terlihat nilai rata-rata utilisasi mengalami perbaikan daripada kondisi awal, disebabkan oleh persebaran pasien yang semakin merata sehingga tidak menyebabkan *overload* di poli Rumah sakit tertentu. Dan untuk nilai rata-rata waktu juga terlihat semakin baik, hal ini dikarenakan pasien tidak menumpuk di poli Rumah sakit tertentu.



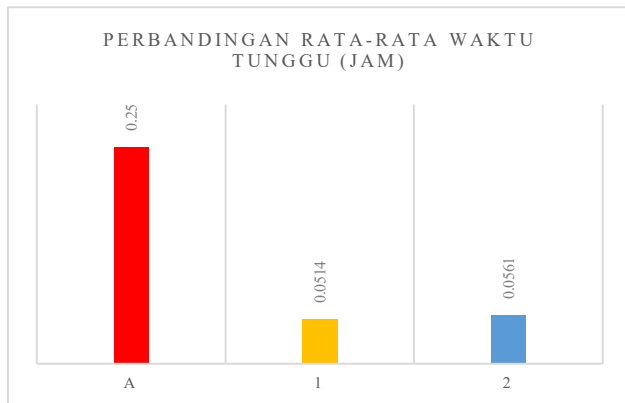
Gambar 4.26 Perbandingan utilisasi & waktu tunggu keseluruhan (2)

4.5 Perbandingan Hasil

Setelah mengetahui hasil nilai rata-rata utilisasi dan rata-rata waktu tunggu pada kondisi awal, optimal 1 dan optimal 2. Kemudian akan dibandingkan hasil-hasil antara kondisi tersebut, terutama melihat kondisi optimal 1 dan kondisi optimal 2. Perbandingan tersebut dapat dilihat pada grafik Gambar 4.27 dan 4.28 yang merupakan nilai rata-rata utilisasi dan rata-rata waktu tunggu untuk seluruh poli di semua Rumah sakit .

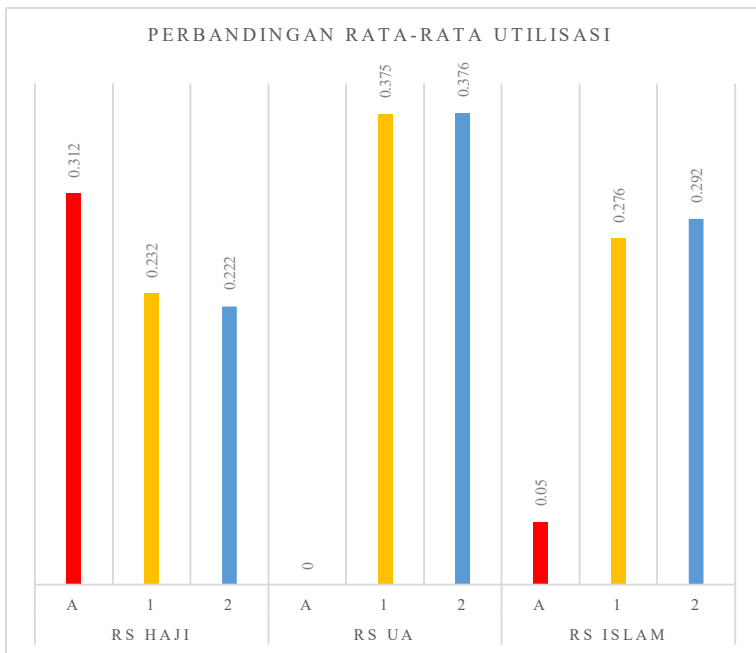


Gambar 4.27 Perbandingan utilisasi seluruh poli semua RS (3)

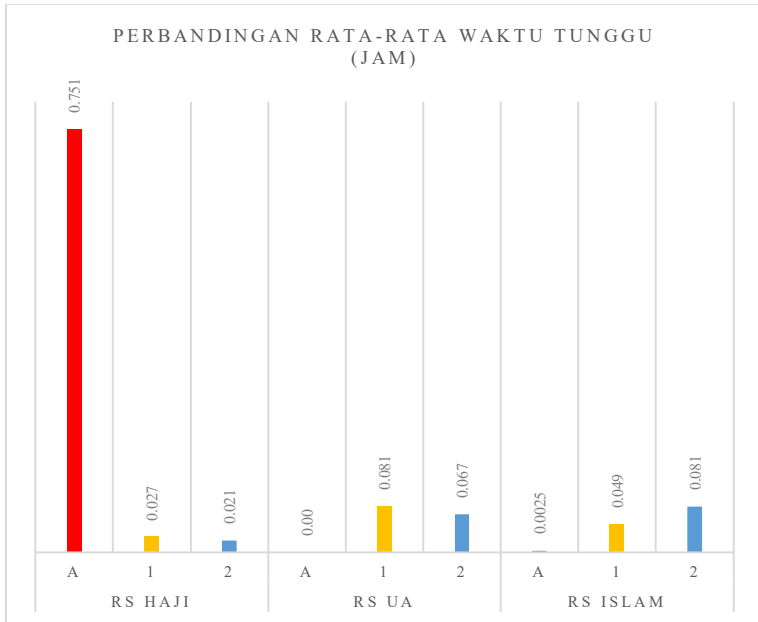


Gambar 4.28 Perbandingan waktu tunggu seluruh poli semua RS (3)

Dapat dilihat pada hasil perbandingan di atas membuktikan solusi *trade-off* yang dihasilkan oleh MO-GA. Bahwa memaksimalkan nilai rata-rata utilisasi akan mengorbankan nilai rata-rata waktu tunggu yang tidak dapat diminimalkan. Dan pada kondisi sebaliknya ketika ingin meminimalkan rata-rata waktu tunggu maka akan mengurangi nilai rata-rata utilisasi. Kondisi seperti ini yang disebut dengan solusi *trade-off* dimana ketika suatu nilai berubah maka akan mempengaruhi nilai solusi yang bisa menjadi minimal atau menjadi maksimal, kondisi ini membenarkan semua solusi yang didapat dan saling memberi pemakluman pada solusi-solusi yang lain yang terhimpun dalam Pareto set. Namun pada kasus ini hasil yang dikeluarkan tidak terlalu berbeda atau bisa disebut relatif sama antara setiap solusinya. Untuk mengetahui perbandingan nilai rata-rata utilisasi dan rata-rata waktu tunggu di setiap Rumah sakit, direpresentasikan dengan Gambar 4.29 dan 4.30.



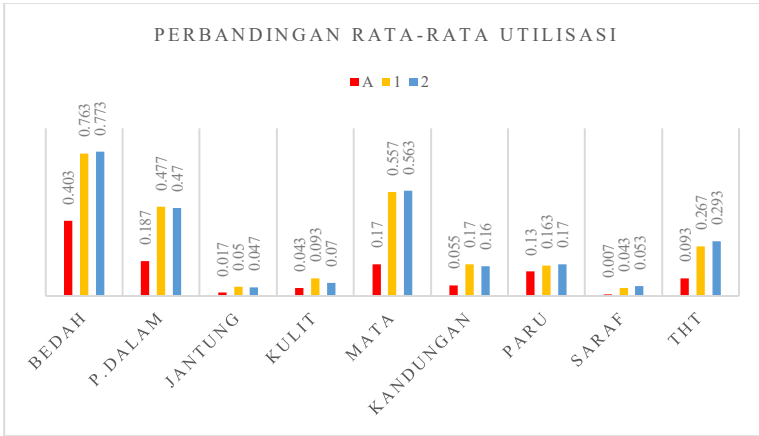
Gambar 4.29 Perbandingan utilisasi setiap RS (3)



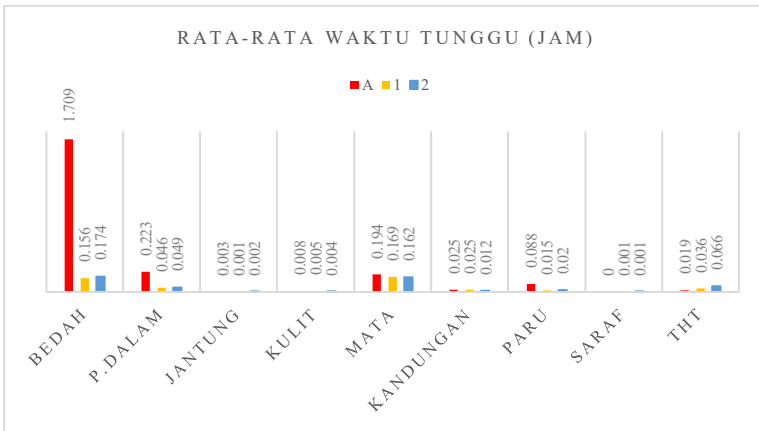
Gambar 4.30 Perbandingan waktu tunggu setiap RS (3)

Dapat dilihat bahwa jika dibandingkan dengan kondisi awal nilai rata-rata utilisasi dan perbandingan rata-rata waktu tunggu hasil yang diperoleh dengan MO-GA mampu menghasilkan solusi yang optimal. Terjadi penurunan nilai rata-rata utilisasi dan rata-rata waktu tunggu di Rumah sakit Haji, hal ini disebabkan oleh menurunnya rate kedatangan pasien ke Rumah sakit tersebut. Yang dimana pasien didistribusikan ke Rumah sakit Universitas Airlangga dan Rumah sakit Islam. Sehingga pasien tidak menumpuk lagi di Rumah sakit Haji, serta dapat memaksimalkan nilai rata-rata utilisasi di Rumah sakit Universitas Airlangga dan Rumah sakit Islam walaupun menyebabkan naiknya nilai rata-rata waktu tunggu di kedua Rumah sakit tersebut.

Perbandingan nilai rata-rata utilisasi kondisi optimal 1 dan kondisi optimal 2 juga membuktikan bahwa kondisi optimal 2 dapat memaksimalkan nilai rata-rata utilisasi daripada kondisi optimal 1. Namun terjadi sedikit penurunan nilai rata-rata utilisasi kondisi optimal 2 di Rumah sakit Haji, jika dibandingkan dengan kondisi optimal 1.



Gambar 4.31 Perbandingan utilisasi poli sejenis di semua RS (3)



Gambar 4.32 Perbandingan waktu tunggu poli sejenis di semua RS (3)

Perbandingan nilai rata-rata utilisasi dan rata-rata waktu tunggu untuk poli sejenis di semua Rumah sakit dapat dilihat pada Gambar 4.31 dan 4.32. Yang dimana juga dibuktikan bahwa kondisi optimal 2 dapat memaksimalkan rata-rata utilisasi daripada kondisi optimal 1. Rata-rata utilisasi relatif meningkat disemua jenis poli, meski terlihat terjadi penurunan di beberapa jenis poli.

Lampiran A

Script MATLAB program MO-GA

A1. Data jumlah server dan waktu pelayanan

```
c=[2 2 1 2 2 1 1 3 1 2 2 1 2 2 1 2 3 2 2 2 1 1 1 4 8 2];  
u=[9.62 4.89 9.57 10.64 4.22 11.43 7.96 4.1 8.74 9.84 6.21 10.4 12.48  
2.54 10.64 13.1 3.64 13.7 13.92 6.85 13.3 7.8 8.32 6.42 4.21 7.86];
```

A2. Parameter GA

```
M=30; %jumlah populasi  
N=26; %jumlah variabel  
rb=0; %batas bawah rb  
ra=30; %batas bawah ra  
maxit=500;  
p.silang=0.9;  
p.mut=0.1;  
rg=20; %konstanta pinalti
```

A3. Inisialisasi populasi

```
for i=1:M  
    pop(i,:)=rb+(ra-rb)*rand(1,N);  
end
```

A4. Evaluasi Fungsi F1

```
[mm,nn]=size(c);  
[oo,pp]=size(u);  
[qq,rr]=size(pop);  
f=[];  
for i=1:nn  
    for j=1:pp  
        for k=1:rr  
            if i==j && j==k;  
                f=[f (pop(:,k)./(c(1,i)*u(1,j)))];  
            end  
        end  
    end  
end  
end
```



```

[ss,tt]=size(f);

%menjumlahkan utilisasi tiap baris
f1=[];
for i=1:ss
    jum=0;
    for j=1:tt
        jum=jum+f(i,j);
    end
    f1=[f1;jum];
end
%rata-rata utilisasi
f1=f1./N;
%constrain utilisai <1

uz=[];
for i=1:qq
    loss=zeros(1,1);
    for j=1:rr
        loss=[loss (f(i,j))-1];
    end
    maxc=max(loss);
    uz=[uz;maxc];
end
uz=uz*rg;
f1=f1-uz;

```

A5. Evaluasi Fungsi F2

```

ff=[];
for i=1:nn
    for j=1:pp
        for k=1:rr
            if i==j && j==k;
                ff= [ff (pop(:,k)/(c(1,i)*u(1,j)))]];
            end
        end
    end
end
end

```

```

[uu,vv]=size(ff);
%men cari qt
qt=[];
for i=1:uu
    qwert=[];
    for j=1:vv
        qwe = ((ff(i,j)*c(1,j))^c(1,j))/(factorial(c(1,j))*((1-
ff(i,j))^2)*c(1,j)*u(1,j));
        qwert = [qwert qwe];
    end
    qt=[qt ;qwert];
end

%men cari Po tiap baris tiap kolom
Po=[];
for i=1:uu
    zxcv=[];
    for j=1:vv
        sigma=0;
        for n=0:c(1,j)-1
            sigma= sigma +((1/factorial(n))*((ff(i,j)*c(1,j)))^n);
        end
        zxc = (factorial(c(1,j))*(1-
ff(i,j)))/(((ff(i,j)*c(1,j))^c(1,j)))+(factorial(c(1,j))*(1-ff(i,j))*sigma);
        zxcv=[zxcv zxc];
    end
    Po=[Po;zxcv];
end

Qt=qt.*Po;
[as,sd]=size(Qt);

f2=[];
for i=1:as
    jumlah=0;
    for j=1:sd
        jumlah= jumlah+Qt(i,j);
    end
end

```

```

    f2=[f2;jumlah];
end
f2=f2./N;

```

```

f2=f2+uz;

```

A6. Ranking Pareto

```

s=[f1 f2];
s=[s zeros(M,6)];

```

```

%pareto rank fonseca&fleming
for i=1:M;
    for j=1:M;
        if s(i,1) < s(j,1) && s(i,2) > s(j,2); %max min
            s(i,3)=s(i,3)+1;
        end
    end
    s(i,3)=s(i,3)+1;
end

```

A7. Penugasan fitness

```

%average fitness
for i=1:M;
    for j=1:M;
        if s(j,3)==s(i,3);
            s(i,4)=s(i,4)+1;
        end
    end
end

```

```

xx=zeros(M,1);
for i=1:M
    for j=1:M
        if s(j,3)==i
            xx(i,1)=xx(i,1)+1;
        end
    end
end

```

```

for i=1:M
    s(i,5)=M;
    for j=1:(s(i,3)-1)
        s(i,5)= s(i,5) - xx(j,1);
    end
    s(i,5) = s(i,5) - 0.5*(s(i,4)-1);
end

%euclidian distance
d=zeros(M,M);
for i=1:M
    for j=1:M
        d(i,j)=sqrt(((s(i,1)-s(j,1))/0.9)^2+((s(i,2)-s(j,2))/9)^2);
    end
end

%sharing function
sh=zeros(M,M);
for i=1:M
    for j=1:M
        if d(i,j)<=0.5;
            sh(i,j)=1-(d(i,j)/0.5);
        end
    end
end

%niche count
for i=1:M
    for j=1:M
        if s(i,3)==s(j,3)
            s(i,6)=s(i,6)+sh(i,j);
        end
    end
end

%shared fitness
for i=1:M
    s(i,7)=s(i,5)/s(i,6);
end

```

```

%scaled fitness
for i=1:M
    sumf=0;
    for j=1:M
        if s(i,3)==s(j,3)
            sumf=sumf+s(j,7);
        end
    end
    s(i,8)=s(i,5)*s(i,4)*s(i,7)/sumf;
end

```

```

FitnV=s(:,8);
maxF1=[];
minF2=[];

```

A8. Algoritma Genetik

```

for gen=1:maxit
%===== geneticoperator =====
%buat pop kosong baru
pop_s=[];

%~~~~~SUS selection~~~~~
Nsel=2;
for i=iterasimulai:2:M
% Identify the population size (Nind)
[Nind,ans] = size(FitnV);

% Perform stochastic universal sampling
cumfit = cumsum(FitnV);
trials = cumfit(Nind) / Nsel * (rand + (0:Nsel-1)');
Mf = cumfit(:, ones(1, Nsel));
Mt = trials(:, ones(1, Nind))';
[NewChrIx, ans] = find(Mt < Mf & [zeros(1, Nsel); Mf(1:Nind-1, :)] <=
Mt);

[ans, shuf] = sort(rand(Nsel, 1));
NewChrIx = NewChrIx(shuf) ;

```

```

%pilih bapak dan ibu
bapak=pop((NewChrIx(1)),:);
ibu=pop((NewChrIx(2)),:);

%~~~~~crossover~~~~~
r=rand;
anak1=(r*bapak)+((1-r)*ibu);
anak2=((1-r)*bapak)+(r*ibu);

%probabilitas kawin silang
if (r<p.silang)
    pop_s(i,:)=anak1;
    pop_s(i+1,:)=anak2;
else
    pop_s(i,:)=bapak;
    pop_s(i+1,:)=ibu;
end
end

%~~~~~Mutasi~~~~~
%prob mutasi
for i=1:(p.mut*M)
    kk=ceil(rand*M);
    pop_s(kk,:)=rand(1,N).*(ra-rb)+rb;
end
%-----

```

--halaman ini sengaja dikosongkan--

LAMPIRAN B

Perhitungan agregasi rate kedatangan setiap penyakit

Puskesmas	λ (per jam)	Mata (%)	λ' (per jam)
Gading	2.3004	26.86084	0.618
Gunung Anyar	1.32192	13.8089	0.183
Kalirungkut	1.7928	16.05667	0.288
Keputih	1.02168	17.10407	0.175
Klampisngasem	1.08	20.12436	0.217
Medokan Ayu	1.62	17.27463	0.28
Menur	1.512	17.09531	0.258
Mojo	2.4516	15.17403	0.372
Mulyorejo	1.3392	12.06786	0.162
Pacarkeling	2.27448	25.83384	0.588
Rangkah	2.28312	26.64032	0.608
Tenggilis	1.3932	9.746835	0.136
Kalijudan	0.8208	15.55891	0.128
Surabaya Pusat	20.418581	31.86275	6.506
Surabaya Barat	24.104927	12.17807	2.936
Surabaya Selatan	27.212609	8.561887	2.33
Surabaya Timur	18.344555	19.5134	3.58
Surabaya Utara	35.619434	4.178495	1.488
Agregasi Rate Kedatangan			20.851

Puskesmas	λ (per jam)	THT	λ' (per jam)
Gading	2.3004	8.629989	0.199
Gunung Anyar	1.32192	11.32199	0.15
Kalirungkut	1.7928	7.949626	0.143
Keputih	1.02168	6.153846	0.063
Klampisngasem	1.08	9.55342	0.103
Medokan Ayu	1.62	6.960168	0.113
Menur	1.512	7.186082	0.109
Mojo	2.4516	6.179116	0.151
Mulyorejo	1.3392	5.377721	0.072
Pacarkeling	2.27448	8.671922	0.197
Rangkah	2.28312	4.189723	0.096
Tenggilis	1.3932	3.122363	0.044
Kalijudan	0.8208	4.07855	0.033
Surabaya Pusat	20.41858	4.991997	1.019
Surabaya Barat	24.10493	9.730801	2.346
Surabaya Selatan	27.21261	11.37967	3.097
Surabaya Timur	18.34455	9.505623	1.744
Surabaya Utara	35.61943	4.054294	1.444
Agregasi Rate Kedatangan			11.121

Puskesmas	λ (per jam)	Paru (%)	λ' (per jam)
Gading	2.3004	4.422869	0.102
Gunung Anyar	1.32192	3.206806	0.042
Kalirungkut	1.7928	3.266431	0.059
Keputih	1.02168	4.343891	0.044
Klampusngasem	1.08	6.274731	0.068
Medokan Ayu	1.62	3.815514	0.062
Menur	1.512	3.214826	0.049
Mojo	2.4516	4.106375	0.101
Mulyorejo	1.3392	2.144686	0.029
Pacarkeling	2.27448	2.728927	0.062
Rangkah	2.28312	2.84585	0.065
Tenggilis	1.3932	1.940928	0.027
Kalijudan	0.8208	5.287009	0.043
Surabaya Pusat	20.41858	2.881152	0.588
Surabaya Barat	24.10493	3.449481	0.831
Surabaya Selatan	27.21261	3.494051	0.951
Surabaya Timur	18.34455	3.717849	0.682
Surabaya Utara	35.61943	4.985806	1.776
Agregasi Rate Kedatangan			5.58

Puskesmas	λ (per jam)	Kulit (%)	λ' (per jam)
Gading	2.3004	5.070119	0.117
Gunung Anyar	1.32192	1.832461	0.024
Kalirungkut	1.7928	2.83353	0.051
Keputih	1.02168	3.9819	0.041
Klampusngasem	1.08	4.691916	0.051
Medokan Ayu	1.62	1.593291	0.026
Menur	1.512	2.004539	0.03
Mojo	2.4516	1.759875	0.043
Mulyorejo	1.3392	2.176697	0.029
Pacarkeling	2.27448	4.184354	0.095
Rangkah	2.28312	3.794466	0.087
Tenggilis	1.3932	1.814346	0.025
Kalijudan	0.8208	1.963746	0.016
Surabaya Pusat	20.41858	2.45098	0.5
Surabaya Barat	24.10493	4.236103	1.021
Surabaya Selatan	27.21261	2.901273	0.79
Surabaya Timur	18.34455	1.570125	0.288
Surabaya Utara	35.61943	1.321859	0.471
Agregasi Rate Kedatangan			3.704

Puskesmas	λ (per jam)	P. Dalam (%)	λ' (per jam)
Gading	2.3004	32.57821	0.749
Gunung Anyar	1.32192	25.98168	0.343
Kalirungkut	1.7928	14.36442	0.258
Keputih	1.02168	25.88235	0.264
Klampisngasem	1.08	15.65856	0.169
Medokan Ayu	1.62	22.51572	0.365
Menur	1.512	15.58245	0.236
Mojo	2.4516	25.53774	0.626
Mulyorejo	1.3392	23.17542	0.31
Pacarkeling	2.27448	26.6222	0.606
Rangkah	2.28312	18.26087	0.417
Tenggilis	1.3932	19.2827	0.269
Kalijudan	0.8208	43.20242	0.355
Surabaya Pusat	20.41858	29.07163	5.936
Surabaya Barat	24.10493	22.0895	5.325
Surabaya Selatan	27.21261	21.42768	5.831
Surabaya Timur	18.34455	19.77038	3.627
Surabaya Utara	35.61943	16.65188	5.931
Agregasi Rate Kedatangan			31.616

Puskesmas	λ (per jam)	Jantung (%)	λ' (per jam)
Gading	2.3004	1.510248	0.035
Gunung Anyar	1.32192	1.701571	0.022
Kalirungkut	1.7928	2.164502	0.039
Keputih	1.02168	1.900452	0.019
Klampusngasem	1.08	0.904466	0.01
Medokan Ayu	1.62	1.928721	0.031
Menur	1.512	1.323752	0.02
Mojo	2.4516	0.899492	0.022
Mulyorejo	1.3392	0.576184	0.008
Pacarkeling	2.27448	0.909642	0.021
Rangkah	2.28312	1.185771	0.027
Tenggilis	1.3932	2.489451	0.035
Kalijudan	0.8208	1.661631	0.014
Surabaya Pusat	20.41858	2.45098	0.5
Surabaya Barat	24.10493	4.236103	1.021
Surabaya Selatan	27.21261	2.901273	0.79
Surabaya Timur	18.34455	1.570125	0.288
Surabaya Utara	35.61943	1.321859	0.471
Agregasi Rate Kedatangan			3.372

Puskesmas	λ (per jam)	Kandungan (%)	λ' (per jam)
Gading	2.3004	3.128371	0.072
Gunung Anyar	1.32192	1.636126	0.022
Kalirungkut	1.7928	1.613538	0.029
Keputih	1.02168	1.628959	0.017
Klampisngasem	1.08	0.847937	0.009
Medokan Ayu	1.62	2.054507	0.033
Menur	1.512	1.815431	0.027
Mojo	2.4516	2.150958	0.053
Mulyorejo	1.3392	16.48528	0.221
Pacarkeling	2.27448	2.001213	0.046
Rangkah	2.28312	3.794466	0.087
Tenggilis	1.3932	1.012658	0.014
Kalijudan	0.8208	2.719033	0.022
Surabaya Pusat	20.41858	2.220888	0.453
Surabaya Barat	24.10493	2.511362	0.605
Surabaya Selatan	27.21261	2.316844	0.63
Surabaya Timur	18.34455	1.69979	0.312
Surabaya Utara	35.61943	1.392832	0.496
Agregasi Rate Kedatangan			3.148

Puskesmas	λ (per jam)	Bedah (%)	λ' (per jam)
Gading	2.3004	16.50485	0.38
Gunung Anyar	1.32192	38.2199	0.505
Kalirungkut	1.7928	50.61	0.907
Keputih	1.02168	38.00905	0.388
Klampusngasem	1.08	40.75749	0.44
Medokan Ayu	1.62	43.22851	0.7
Menur	1.512	50.79425	0.768
Mojo	2.4516	42.62808	1.045
Mulyorejo	1.3392	33.4507	0.448
Pacarkeling	2.27448	28.01698	0.637
Rangkah	2.28312	37.54941	0.857
Tenggilis	1.3932	60.16878	0.838
Kalijudan	0.8208	24.62236	0.202
Surabaya Pusat	20.41858	23.96959	4.894
Surabaya Barat	24.10493	42.78639	10.314
Surabaya Selatan	27.21261	47.01315	12.794
Surabaya Timur	18.34455	41.57295	7.626
Surabaya Utara	35.61943	64.51384	22.979
Agregasi Rate Kedatangan			66.724

Puskesmas	λ (per jam)	Saraf (%)	λ' (per jam)
Gading	2.3004	0.539374	0.012408
Gunung Anyar	1.32192	0.78534	0.010382
Kalirungkut	1.7928	1.062574	0.01905
Keputih	1.02168	0.633484	0.006472
Klampusngasem	1.08	0.678349	0.007326
Medokan Ayu	1.62	0.587002	0.009509
Menur	1.512	0.832073	0.012581
Mojo	2.4516	1.016817	0.024928
Mulyorejo	1.3392	0.224072	0.003001
Pacarkeling	2.27448	0.4245	0.009655
Rangkah	2.28312	0.632411	0.014439
Tenggilis	1.3932	0.295359	0.004115
Kalijudan	0.8208	0.755287	0.006199
Surabaya Pusat	20.41858	0.720288	0.147073
Surabaya Barat	24.10493	1.165365	0.28091
Surabaya Selatan	27.21261	0.776456	0.211294
Surabaya Timur	18.34455	0.726124	0.133204
Surabaya Utara	35.61943	1.304116	0.464519
Agregasi Rate Kedatangan			1.377065

--halaman ini sengaja dikosongkan--

LAMPIRAN C

Waktu kedatangan pasien setiap poli Rumah Sakit.

Poli Mata RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:00	9:06	6
2	9:03	9:08	5
3	9:06	9:12	6
4	9:08	9:16	8
5	9:12	9:21	9
6	9:16	9:22	6
7	9:21	9:25	4
8	9:22	9:27	5
9	9:25	9:30	5
10	9:27	9:34	7
11	9:30	9:38	8
12	9:34	9:40	6
13	9:38	9:42	4
14	9:40	9:43	3
15	9:42	9:48	6
16	9:43	9:51	9

Poli THT RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:00	10:04	4
2	10:02	10:09	7
3	10:04	10:12	8
4	10:09	10:16	7
5	10:12	10:17	5
6	10:16	10:19	3
7	10:17	10:23	6
8	10:19	10:22	3
9	10:22	10:31	9
10	10:23	10:26	3
11	10:26	10:30	4

12	10:30	10:36	6
13	10:31	10:35	4
14	10:35	10:41	6
15	10:36	10:39	3
16	10:39	10:47	8

Poli Paru RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:12	8:21	9
2	8:21	8:29	8
3	8:29	8:34	5
4	8:34	8:38	4
5	8:38	8:45	7
6	8:45	8:55	10
7	8:55	9:06	11
8	9:06	9:11	5
9	9:11	9:16	5
10	9:16	9:23	7
11	9:23	9:30	7

Poli Kulit RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:34	9:39	5
2	9:35	9:41	6
3	9:39	9:45	6
4	9:41	9:44	3
5	9:44	9:49	5
6	9:45	9:58	13
7	9:49	9:54	5
8	9:54	9:58	4
9	9:58	10:00	2
10	9:58	10:03	5
11	10:00	10:05	5
12	10:03	10:14	11
13	10:05	10:12	7

Poli Saraf RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:15	10:21	6
2	10:17	10:20	3
3	10:20	10:32	12
4	10:21	10:25	4
5	10:25	10:30	5
6	10:30	10:35	5
7	10:32	10:40	8
8	10:35	10:39	4
9	10:40	10:49	9
10	10:39	10:43	4
11	10:43	10:48	5
12	10:48	10:52	4
13	10:49	10:54	5

Poli Penyakit Dalam RS Haji			
1	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:55	10:59	4
2	10:57	11:01	4
3	10:59	11:06	7
4	11:01	11:06	5
5	11:06	11:08	2
6	11:06	11:11	5
7	11:08	11:11	3
8	11:11	11:15	4
9	11:11	11:15	4
10	11:15	11:19	5
11	11:15	11:20	4
12	11:19	11:22	3
13	11:20	11:27	7
14	11:22	11:28	6

Poli Jantung RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:48	10:57	9
2	10:50	10:56	6
3	10:56	11:01	5
4	10:57	11:02	5
5	11:01	11:05	4
6	11:02	11:05	3
7	11:05	11:09	5
8	11:05	11:10	4
9	11:09	11:13	4
10	11:10	11:17	7
11	11:13	11:18	5
12	11:17	11:20	3
13	11:18	11:22	4
14	11:20	11:24	4
15	11:22	11:26	4

Poli Kandungan RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:24	8:28	4
2	8:28	8:33	5
3	8:33	8:40	7
4	8:40	8:45	5
5	8:45	8:52	7
6	8:52	9:01	9
7	9:01	9:14	13
8	9:14	9:23	9
9	9:23	9:29	6
10	9:29	9:37	8
11	9:37	9:44	7

Poli Bedah RS Haji			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:06	9:17	11
2	9:07	9:16	9
3	9:09	9:13	4
4	9:13	9:20	7
5	9:16	9:26	10
6	9:17	9:25	8
7	9:20	9:26	6
8	9:25	9:33	8
9	9:26	9:35	9
10	9:26	9:33	7
11	9:33	9:37	4
12	9:33	9:40	7
13	9:35	9:44	9
14	9:37	9:47	10

Poli Mata RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:32	8:44	12
2	8:44	8:47	3
3	8:47	8:54	7
4	8:54	9:00	6
5	9:00	9:03	3
6	9:03	9:08	5
7	9:08	9:10	2
8	9:10	9:16	6
9	9:16	9:30	14
10	9:30	9:40	10
11	9:40	9:50	10
12	9:50	9:55	5
13	9:55	9:57	2
14	9:57	10:00	3
15	10:00	10:04	4
16	10:04	10:18	14

Poli THT RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:26	10:29	3
2	10:29	10:38	9
3	10:38	10:43	5
4	10:43	10:48	5
5	10:48	10:56	8
6	10:56	11:01	5
7	11:01	11:08	7
8	11:08	11:11	3
9	11:11	11:15	4
10	11:15	11:19	4
11	11:19	11:24	5
12	11:24	11:32	8
13	11:32	11:37	5
14	11:37	11:42	5
15	11:42	11:52	10
16	11:52	11:54	2
17	11:54	12:06	12

Poli Paru RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	13:03	13:13	10
2	13:13	13:20	7
3	13:20	13:26	6
4	13:26	13:38	12
5	13:38	13:47	9
6	13:47	13:52	5
7	13:52	13:57	5
8	13:57	14:04	7
9	14:04	14:10	6
10	14:10	14:22	12
11	14:22	14:31	9
12	14:31	14:40	9
13	14:40	14:50	10
14	14:50	14:54	4
15	14:54	15:00	6

Poli Kulit RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:14	8:17	3
2	8:17	8:23	6
3	8:23	8:28	5
4	8:28	8:38	10
5	8:38	8:42	4
6	8:42	8:46	6
7	8:46	9:00	14
8	9:00	9:05	5
9	9:05	9:12	7
10	9:12	9:17	5
11	9:17	9:23	6
12	9:23	9:30	7
13	9:30	9:38	8
14	9:38	9:42	4
15	9:42	9:47	5

Poli Saraf RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:58	10:02	4
2	10:02	10:09	7
3	10:09	10:13	4
4	10:13	10:22	9
5	10:22	10:27	5
6	10:27	10:35	8
7	10:35	10:40	5
8	10:40	10:43	3
9	10:43	10:48	5
10	10:48	10:54	6
11	10:54	11:02	8
12	11:02	11:06	4
13	11:06	11:09	3
14	11:09	11:12	3
15	11:12	11:16	4
16	11:16	11:19	3
17	11:19	11:23	4

Poli Penyakit Dalam RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	11:31	11:36	5
2	11:33	11:36	3
3	11:36	11:40	4
4	11:36	11:41	5
5	11:40	11:45	5
6	11:41	11:48	7
7	11:45	11:53	8
8	11:48	11:57	9
9	11:53	11:58	5
10	13:05	13:09	4
11	13:05	13:10	5
12	13:09	13:13	4
13	13:10	13:13	3
14	13:13	13:17	4
15	13:13	13:18	5
16	13:17	13:20	3
17	13:18	13:22	4
18	13:20	13:24	4
19	13:22	13:26	4

Poli Jantung RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	13:35	13:40	5
2	13:40	13:45	5
3	13:45	13:48	3
4	13:48	13:54	6
5	13:54	13:57	3
6	13:57	14:00	3
7	14:00	14:03	3
8	14:03	14:09	6

9	14:09	14:16	7
10	14:16	14:19	3
11	14:19	14:23	4
12	14:23	14:29	6
13	14:29	14:36	7
14	14:36	14:43	7
15	14:43	14:51	8
16	14:51	14:54	3
17	14:54	14:58	4
18	14:58	15:03	3
19	15:03	15:06	3

Poli Kandungan RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:38	9:42	4
2	9:42	9:50	8
3	9:50	9:56	6
4	9:56	10:03	7
5	10:03	10:09	6
6	10:09	10:18	9
7	10:18	10:29	11
8	10:29	10:37	8
9	10:37	10:48	11
10	10:48	10:55	7
11	10:55	11:04	9
12	11:04	11:08	4
13	11:08	11:30	12

Poli Bedah RS Islam			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:26	9:35	9
2	9:17	9:23	6

3	9:23	9:33	10
4	9:35	9:41	6
5	9:33	9:41	8
6	9:41	9:50	9
7	9:41	9:47	6
8	9:50	9:59	9
9	9:47	10:01	14
10	9:59	10:06	7
11	10:01	10:10	9
12	10:06	10:18	12
13	10:10	10:19	9
14	10:18	10:30	12
15	10:20	10:33	13

Poli Mata UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:00	8:07	7
2	8:04	8:15	11
3	8:07	8:23	16
4	8:15	8:31	16
5	8:23	8:38	15
6	8:31	8:43	12
7	8:38	8:46	8

Poli THT UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:53	9:13	20
2	9:05	9:23	18
3	9:13	9:24	11
4	9:23	9:37	14
5	9:24	9:39	15
6	9:37	9:46	9
7	9:39	9:53	13
8	9:46	10:01	15
9	9:53	10:05	12

Poli Paru UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	8:15	8:25	10
2	8:17	8:34	17
3	8:20	8:34	14
4	8:25	8:39	14
5	8:34	8:50	16
6	8:34	8:45	11
7	8:39	9:04	15
8	8:45	9:05	20

Poli Kulit UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	9:12	9:20	8
2	9:16	9:22	6
3	9:20	9:31	11
4	9:22	9:36	14
5	9:31	9:46	15
6	9:36	9:45	9
7	9:45	9:52	7
8	9:46	10:01	15
9	9:52	9:56	4
10	9:56	10:06	10

Poli Saraf UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:15	10:38	23
2	10:23	10:49	26
3	10:38	11:01	23
4	10:49	11:08	19
5	11:01	11:25	24
6	11:08	11:33	25

Poli Penyakit Dalam UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	10:12	10:24	12
2	10:16	10:37	21
3	10:21	10:38	17
4	10:24	10:37	13
5	10:37	10:52	15
6	10:37	10:51	14
7	10:38	11:01	23
8	10:51	11:08	17

Poli Jantung UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	11:12	11:20	8
2	11:17	11:27	10
3	11:20	11:29	9
4	11:27	11:36	9
5	11:29	11:35	6
6	11:35	11:44	9
7	11:36	11:48	12
8	11:44	11:50	6
9	11:48	11:56	8
10	11:50	12:00	10
11	11:56	12:07	11
12	12:00	12:13	13

Poli Bedah UA			
No	Jam Masuk	Jam Keluar	Waktu Pelayanan
1	11:34	11:51	17
2	11:36	11:47	11
3	11:39	11:49	10
4	11:40	11:59	19
5	11:41	11:54	13
6	11:41	11:57	16
7	11:42	11:56	14

Lampiran D

Poli	Utilisasi											
	RS Haji			RS UA			RS Islam			Rata-Rata		
	A	1	2	A	1	2	A	1	2	A	1	2
Bedah	1	0.78	0.74	0	0.8	0.77	0.21	0.71	0.81	0.403	0.763	0.773
P.Dalam	0.46	0.39	0.34	0	0.63	0.66	0.1	0.41	0.41	0.187	0.477	0.47
Jantung	0.02	0.02	0.01	0	0.06	0.07	0.03	0.07	0.06	0.017	0.05	0.047
Kulit	0.12	0.05	0.04	0	0.09	0.07	0.01	0.14	0.1	0.043	0.093	0.07
Mata	0.49	0.31	0.3	0	0.71	0.68	0.02	0.65	0.71	0.17	0.557	0.563
Kandungan	0.11	0.17	0.18	-	-	-	0	0.17	0.14	0.055	0.17	0.16
Paru	0.35	0.21	0.21	0	0.13	0.13	0.04	0.15	0.17	0.13	0.163	0.17
Saraf	0.02	0.01	0.01	0	0.07	0.11	0	0.05	0.04	0.007	0.043	0.053
THT	0.24	0.15	0.17	0	0.51	0.52	0.04	0.14	0.19	0.093	0.267	0.293
Rata-Rata	0.312	0.232	0.222	0	0.375	0.376	0.05	0.276	0.292			
Rata-Rata keseluruhan Poli di Seluruh RS										0.12	0.2915	0.2938

Poli	Rata-rata waktu tunggu											
	RS Haji			RS UA			RS Islam			Rata-Rata		
	A	1	2	A	1	2	A	1	2	A	1	2
Bedah	5.11	0.1508	0.1267	0.00	0.1599	0.0792	0.01	0.1558	0.3167	1.709	0.156	0.174
P.Dalam	0.67	0.0199	0.0104	0.00	0.0987	0.1196	0.00	0.0208	0.0171	0.223	0.046	0.049
Jantung	0.00	0.0002	0	0.00	0	0	0.01	0.0028	0.0046	0.003	0.001	0.002
Kulit	0.02	0	0.0006	0.00	0.0029	0.0006	0.00	0.0115	0.0114	0.008	0.005	0.004
Mata	0.58	0.0171	0.0096	0.00	0.3048	0.1605	0.00	0.1848	0.3173	0.194	0.169	0.162
Kandungan	0.05	0.0187	0.0112	-	-	-	0.00	0.032	0.0123	0.025	0.025	0.012
Paru	0.26	0.0351	0.0307	0.00	0	0	0.00	0.01	0.0278	0.088	0.015	0.02
Saraf	0.00	0	0	0.00	0	0.0032	0.00	0.003	0.0007	0	0.001	0.001
THT	0.06	0.0029	0.0023	0.00	0.0856	0.1724	0.00	0.018	0.0236	0.019	0.036	0.066
Rata-Rata	0.751	0.027	0.021	0.00	0.081	0.067	0.0025	0.049	0.081			
Rata-Rata keseluruhan Poli di Seluruh RS										0.25	0.0514	0.0561

BAB V

PENUTUP

Hasil dari perancangan dan penelitian Tugas Akhir dirangkum dan dirumuskan kesimpulan. Kesimpulan ini menerangkan hasil dari pengujian dan simulasi yang telah dilaksanakan.

Selama proses perancangan dan penelitian, terdapat banyak kendala yang dihadapi. Kendala tersebut telah penulis rangkum dan dirumuskan dalam bentuk saran untuk penyempurnaan dan penelitian lebih lanjut

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis, dapat diperoleh kesimpulan sebagai berikut:

- a. MO-GA dapat mengoptimalkan nilai rata-rata utilisasi dan rata-rata waktu tunggu pada sistem antrian pasien. Rata-rata utilisasi yang semula sebesar 0,12 menjadi 0,2915 untuk solusi optimal 1 dan 0,2938 untuk solusi optimal 2. Rata-rata waktu tunggu yang semula sebesar 0,25 jam menjadi 0,0514 jam untuk solusi optimal 1 dan 0,0561 jam untuk solusi optimal 2.
- b. Dalam permasalahan distribusi pasien BPJS ini terdapat dua tujuan yang ingin dioptimalkan namun saling tidak mengoptimalkan satu sama lain. Dengan kata lain untuk mendapatkan nilai optimal fungsi tujuan 1 maka harus mengorbankan nilai optimal fungsi tujuan 2. Meskipun dalam kasus ini nilai optimalnya relatif sama.
- c. MO-GA yang diuji dalam penelitian tugas akhir ini dapat menurunkan rata-rata waktu tunggu pada solusi optimal 1 yaitu 0,0514 jam dibandingkan hasil dari solusi optimal 2 yang bernilai 0,0561 jam. Namun untuk nilai rata-rata utilisasi lebih baik nilainya pada solusi optimal 2 yaitu 0,2938 dibandingkan hasil dari solusi optimal 1 yang bernilai 0,2915.

5.2 Saran

Dari hasil penelitian yang dilakukan, untuk pengembangan berikutnya, disarankan beberapa hal berikut ini:

- a. Pemodelan sistem antrian di poli Rumah sakit yang lebih mendalam akan membantu untuk mengembangkan penelitian.

- b. Perlunya memperhatikan pasien umum yang datang ke poli rumah sakit.
- c. Penelitian selanjutnya diharapkan menggunakan objek yang lebih banyak (Rumah sakit).
- d. Penelitian kedepan diharapkan dapat merealisasikan perancangan optimisasi sistem rujukan ini dengan membuat *software*.
- e. Penggunaan MO-GA memungkinkan untuk menemukan nilai yang lebih optimal dengan merubah parameter *crossover*, *mutation*, dan jumlah populasi.

DAFTAR PUSTAKA

- [1] BPJS Kesehatan. <http://bpjs-kesehatan.go.id>. Diakses tanggal 25 Februari 2016.
- [2] C.I. Oliver, “*Fundamental of Stochastic Network*”, Massachusetts, Wiley, 2011.
- [3] Stallings, Wiliam, “*Queueing Analysis*”. 2000.
- [4] Santosa, Budi dan Paul Willy, “*Metoda Metaheuristik Konsep dan Implementasi*”, Surabaya, Guna Widya, 2011.
- [5] Achmad Basuki, “*Algoritma Genetika*”, Politeknik Elektronika Negeri Surabaya PENS-ITS, Surabaya, 2003.
- [6] David E, and Goldberg, “*Genetic Algorithms in Search, Optimization and Machine Learning*”, Addison-Wesley, Reading, Massachusetts, 1989.
- [7] Pencheva, Tania. Krassimir Atanassov. Anthony Shannon, “*Modelling of a Stochastic Universal Sampling Selection Operator in Genetic Algorithms Using Generalized Nets*” ,Tenth Int. Workshop on Generalized Nets. 2009.
- [8] Deb K, “*Multi-objective optimization using evolutionary algorithms*”, Chichester, UK, Wiley, 2001.
- [9] C. M. Fonseca and P. J. Fleming (1993), “*Genetic Algorithms for Multi-Objective Optimization: Formulation, Discussion and Generalization*”, Proceedings of the 5th International Conference on Genetic Algorithms, pp. 416-423
- [10] Rjiurata,Tadahiko. Hisao Ishibuchi, “*MOGA: Multi-Objective Genetic Algorithms*”, IEEE. 1995.
- [11] Zitzler, Eckart. Lothar Thiele, “*Multiobjective Optimization Using Evolutionary Algorithms*”. Parallel Problem Solving from Nature PPSNV.Amsterdam,The Netherlands. 1998.
- [12] Ekpenyong and Emmanuel John, “*Analysis of Multi-Server Single Queue System with Multiple Phases*”, Pak.j.stat.oper.res. Vol.VII No.2. 2011.
- [13] Fitri, Wahyi Ba’dal, “*Sistem Informasi Pemilihan Rumah Sakit Prioritas di Area Surabaya Berbasis Sistem Pendukung Keputusan Menggunakan Metode Analytical Hierarchy Process*”, Tugas Akhir, Institut Teknologi Sepuluh Nopember. Surabaya. 2015.
- [14] Qiu, Yunzhe dkk. “*A Simulation Based GA for Multi-objective Optimization in Patient Flow Distribution*”, IEEE International Conference, 2014.

RIWAYAT PENULIS



Aulia Vica Rachman Hernanda yang biasanya dipanggil Vica lahir pada tanggal 23 Oktober 1994. Vica merupakan anak pertama dari tiga bersaudara. Lulus dari SDN Airlangga V Surabaya, kemudian melanjutkan sekolah di SMPN 1 Surabaya dan lulus pada tahun 2009. Kemudian melanjutkan sekolah di SMAN 5 Surabaya dan lulus pada tahun 2012. Setelah lulus SMA penulis memilih untuk melanjutkan pendidikan di Teknik Elektro Institut Teknologi Sepuluh Nopember pada tahun 2012. Penulis juga aktif dalam kegiatan mahasiswa, seperti Himatekro dan pernah menjadi sekretaris departemen PSDM. Memasuki tahun ketiga kuliah yaitu pada tahun 2014 penulis memutuskan untuk mengambil bidang studi Teknik Sistem Pengaturan. Dan pada tahun terakhir yaitu pada tahun 2015 penulis menjadi asisten Laboratorium Teknik Sistem. Pada bulan Juni 2016 penulis mengikuti seminar dan ujian Tugas Akhir sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik Elektro dari Institut Teknologi Sepuluh Nopember Surabaya.

E-mail : vicarachman@gmail.com

HP : 085730149301