

OPTIMASI GIZI PADA ANAK PANTI ASUHAN DENGAN IMMUNE GENETIC ALGORITHM

Iqbal Dhea Furqon Tauhid¹⁾, Imam Cholissodin²⁾, M. Tanzil Furqon³⁾

Program Studi Teknik Informatika, Jurusan Teknik Informatika

Fakultas Ilmu Komputer

Universitas Brawijaya

Jl. Veteran, Malang 65145, Indonesia

Email: ¹iqbal.dft@gmail.com, ²imamcs@ub.ac.id, ³m.tanzil.furqon@ub.ac.id

Abstrak

Gizi merupakan zat yang dibutuhkan manusia bagi pertumbuhan dan kesehatannya. Meskipun begitu asupan gizi yang kurang tepat juga dapat mengakibatkan berbagai permasalahan yang mengganggu kesehatan. Sayangnya masih dapat dijumpai permasalahan gizi di berbagai tempat, salah satunya di panti asuhan. *Immune Genetic Algorithm* merupakan algoritma genetika yang dioptimasi dengan konsep imunitas dalam biologi dengan tujuan meningkatkan kemampuan algoritma genetika dalam pencarian optimum lokal sementara tetap menjaga kemampuannya dalam pencarian optimum global. Pencarian solusi rekomendasi menu makanan didapat dari mengkombinasikan kromosom yang diproses dengan operator genetika (*crossover* dan mutasi) yang kemudian dioptimasi dengan metode vaksinasi lalu diseleksi dengan proses seleksi imun. Hasil dari pengujian menunjukkan nilai *fitness* terbaik didapatkan dengan parameter: jumlah populasi awal = 70, probabilitas *crossover* = 0,7, probabilitas mutasi = 0,7, konstanta *affinity* = 0,8, konstanta *similarity* = 0,7, serta *death factor* = 0,7. Algoritma ini dapat memberikan rekomendasi menu makanan yang dapat memenuhi kebutuhan gizi tiap hari untuk satu populasi, hanya saja biaya dari menu makanan yang dihasilkan masih cukup tinggi.

Kata kunci: gizi, rekomendasi menu makanan, panti asuhan, algoritma genetika, *immune genetic algorithm*, seleksi imun

Abstract

Nutrition is a substance that human need for growth and health. However, lack of proper nutrition could lead to various problems that interfere with health. Unfortunately nutritional problem still could be found in various places, one of them is in orphanage. Immune Genetic Algorithm is genetic algorithm that optimized with the concept of immunity in biology to improving the ability of genetic algorithm in search of local optimum while maintain its ability in search of global optimum. The search of menu recommendation is obtained from combination of chromosome which processed with genetic operator (crossover and mutation) and then optimized with vaccination method and then selected by immune selection process. The result of testing showing the best fitness value obtained with parameters: initial population number = 70, crossover probability = 0.7, mutation probability = 0.7, affinity constant = 0.8, similarity constant = 0.7, and death factor = 0.7. This algorithm could provide food recommendation which can meet the daily nutritional needs for a population, however the cost the resulting menu is still quite high.

Keywords: nutrition, food recommendation, orphanage, genetic algorithm, *immune genetic algorithm*, immune selection

1. PENDAHULUAN

1.1. Latar Belakang

Gizi merupakan zat yang dibutuhkan manusia bagi pertumbuhan dan kesehatannya, meskipun begitu asupan gizi yang kurang tepat juga dapat mengakibatkan berbagai permasalahan yang mengganggu kesehatan. Berdasarkan penelitian oleh (William, 2010) pada salah satu panti asuhan di Medan, dijumpai adanya penghuni yang menderita malnutrisi ringan sebesar 14.4%, dan overweight sebanyak 8.7%. Hal ini lebih mempertegas hasil temuan dari penelitian yang dilakukan oleh Save the Children dan Kementerian Sosial pada tahun 2007 yang dikutip oleh (Indonesia, 2011), yang salah satu hasil temuannya menunjukkan bahwa kurangnya pengetahuan pengurus panti tentang situasi anak serta pengasuhan yang seharusnya diterima anak.

Selain itu hasil penelitian tersebut juga menggambarkan tentang buruknya kualitas makanan yang diterima oleh anak-anak panti asuhan.

Immune Genetic Algorithm merupakan algoritma genetika yang dioptimasi dengan konsep imunitas dalam biologi dengan tujuan meningkatkan kemampuan algoritma genetika dalam pencarian optimum lokal sementara tetap menjaga kemampuan algoritma genetika dalam pencarian optimum global (Wang, et al., 1998). Penggunaan *Immune Genetic Algorithm* dalam penelitian ini didasarkan pada penelitian yang dilakukan oleh (Juan, et al., 2009) yang berjudul "Research on Application of IGA (*Immune Genetic Algorithm*) to the Solution of Course-Timetabling Problem", penelitian menunjukkan bahwa *Immune Genetic Algorithm* dapat mengurangi fenomena degenerasi atau

turunnya kualitas keturunan, serta dapat meningkatkan kecepatan konvergensi dalam setiap iterasi algoritma genetika. Senada dengan penelitian tersebut, (Jin & Fan, 2013) dengan judulnya “A New Immune Genetic Algorithm for 0-1 Knapsack Problem” juga memberi kesimpulan yang sama.

Dengan dilakukannya penelitian ini, diharapkan hasil dari penelitian ini dapat membantu untuk diterapkannya pola makan bergizi seimbang bagi penghuni panti asuhan, yakni dengan memberikan beberapa opsi contoh menu makanan yang dapat digunakan sehari-hari.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan sebelumnya, dapat ditarik rumusan masalah sebagai berikut :

1. Bagaimana menerapkan Immune Genetic Algorithm untuk mengoptimasi gizi pada anak panti asuhan?
2. Bagaimana pengaruh parameter Immune Genetic Algorithm terhadap hasil optimasi gizi untuk anak panti asuhan?

1.3. Batasan Masalah

untuk membatasi penelitian sehingga tidak melenceng terlalu jauh dari tujuan awal maka diberikan batasan masalah sebagai berikut :

1. Algoritma yang digunakan untuk optimasi ialah Immune Genetic Algorithm.
2. Tidak melakukan perbandingan dengan metode lain.
3. Optimasi gizi berpatokan pada kebutuhan energi anak sesuai jenis kelamin dan rentang usia tertentu sesuai dengan tabel Angka Kecukupan Gizi.
4. Data anak didapatkan dari panti asuhan Nurul Abyadh.
5. Harga yang digunakan adalah harga umum pada kota Malang.

1.4. Tujuan

Berdasarkan latar belakang dan rumusan masalah yang telah dijabarkan sebelumnya dapat disimpulkan tujuan dari penelitian ini ialah :

1. Menerapkan Immune Genetic Algorithm untuk mengoptimasi gizi untuk anak panti asuhan.
2. Mengetahui pengaruh parameter Immune Genetic Algorithm terhadap hasil optimasi gizi untuk anak panti asuhan.

2. LANDASAN TEORI

2.1 Panti Asuhan

Panti asuhan atau yang juga disebut dengan Lembaga Kesejahteraan Sosial Anak merupakan lembaga-lembaga kesejahteraan sosial yang dibentuk oleh pemerintah maupun masyarakat yang melaksanakan pengasuhan anak. Lembaga ini berperan dalam mendukung pengasuhan anak oleh keluarga serta memberi pelayanan bagi anak yang membutuhkan pengasuhan alternatif. Penempatan anak di lembaga seperti ini merupakan pilihan terakhir dari pengasuhan alternatif untuk anak yang

tidak dapat diasuh dalam keluarga inti, keluarga besar, kerabat, maupun keluarga pengganti (Indonesia, 2011).

2.2 Gizi

Istilah gizi mulai dikenal pada tahun 1950-an di Indonesia. Kata gizi yang merupakan terjemahan dari kata dalam bahasa Inggris “*nutrition*” berasal dari bahasa arab “*ghidza*” yang dalam dialek mesir dibaca “*gizi*”, sementara kata *nutrition* sendiri diterjemahkan langsung menjadi kata nutrisi yang memiliki arti serupa. Zat gizi diperlukan tubuh untuk memperoleh energi yang dapat digunakan untuk menjaga kelangsungan fungsi tubuh, tumbuh dan berkembang, serta melakukan aktifitas. Zat-zat gizi yang merupakan sumber energi utama bagi tubuh ialah karbohidrat, lemak, dan protein. (Muchtadi, 2014).

2.3 Angka Kecukupan Gizi

Menurut Peraturan Menteri Kesehatan nomor 75 tahun 2013, Angka Kecukupan Gizi (AKG) merupakan suatu kecukupan rata-rata zat gizi setiap hari bagi semua orang menurut golongan umur, jenis kelamin, ukuran tubuh, serta aktifitas tubuh untuk mencapai derajat kesehatan yang optimal. Fungsi dari AKG itu sendiri dapat digunakan dalam penilaian kadar kecukupan gizi, acuan penyusunan makanan sehari-hari, acuan perhitungan perencanaan penyediaan pangan tingkat regional maupun nasional, acuan pendidikan gizi, dan acuan label pangan yang mencantumkan informasi nilai gizi. (Indonesia, 2013).

2.4 Algoritma Genetika

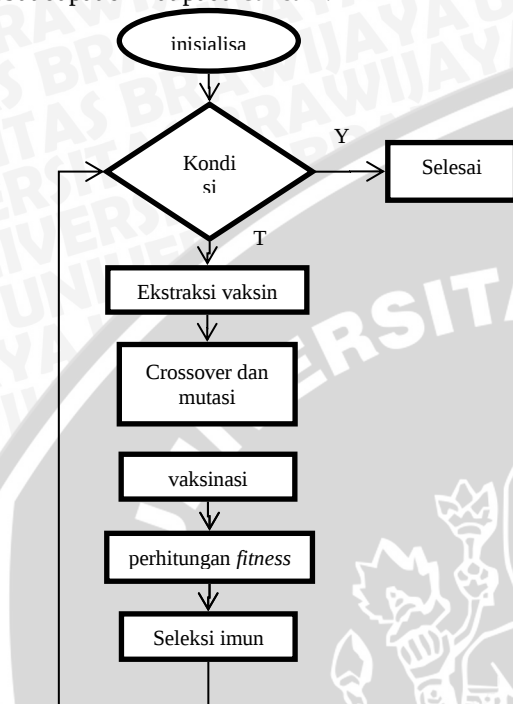
Tujuan utama dari algoritma genetika ialah untuk memaksimalkan hasil dari kandidat solusi dalam suatu populasi terhadap fungsi biaya dari domain problem. Strategi utama algoritma ini yaitu dengan berulang kali mengganti kombinasi solusi dengan metode rekombinasi dan mutasi pada mekanisme genetis dari suatu populasi kandidat solusi dengan mempertimbangkan fungsi biaya atau fungsi *fitness*. Secara umum algoritma ini diawali dengan inisialisasi populasi awal, kemudian evaluasi populasi tersebut untuk mendapatkan individu terbaik dalam satu populasi. Lakukan perulangan hingga mencapai kondisi selesai yang diinginkan untuk mendapatkan individu individu yang lebih baik dengan menggunakan metode *crossover* dan mutasi (Brownlee, 2011).

2.5 Immune Genetic Algorithm

Immune Genetic Algorithm (IGA) merupakan algoritma genetika yang dioptimasi dengan konsep imunitas dalam biologi dengan tujuan meningkatkan kemampuan algoritma genetika dalam pencarian optimum lokal sementara tetap menjaga kemampuan algoritma genetika dalam pencarian optimum global (Wang et al., 1998).

IGA dapat ditambahkan dengan operasi vaksinasi untuk meningkatkan probabilitas dalam mendapatkan nilai *fitness* yang lebih tinggi (Lei & Licheng, 2000) serta mencegah jatuhnya populasi

dalam konvergensi lokal (Yan-cong et al., 2011). Lebih lanjut, (Jin & Fan, 2013) mengoptimisasi metode ekstraksi vaksin dengan menambah informasi lokasi vaksin tersebut untuk lebih mempercepat konvergensi dan mencegah konvergensi prematur, sehingga flowchart algoritma tersebut dapat dilihat pada Gambar 2.1



Gambar 2.1 Flowchart Immune Genetic Algorithm

2.6 Inisialisasi

Generasi populasi awal dilakukan dengan membuat antibodi antibodi baru sebanyak jumlah populasi awal yang diinginkan. Setiap antibodi direpresentasikan dengan kromosom yang menyatakan menu makanan. Tiap kromosom terdiri atas beberapa fragmen, dimana tiap fragmen memiliki panjang 4 gen yang merepresentasikan menu makanan dalam satu kali makan baik pagi, siang, maupun malam yang meliputi makanan pokok, sayur, lauk, dan minuman. Setiap tiga fragmen yang berurutan merupakan menu makanan dalam satu hari dimana fragmen pertama untuk makan pagi, fragmen kedua untuk makan siang, dan fragmen ketiga untuk makan malam.

Hari	1												2											
	Pagi				Siang				Malam				Pagi				Siang				Malam			
Menu	P	S	L	M	P	S	L	M	P	S	L	M	P	S	L	M	P	S	L	M	P	S	L	M
A1	3	8	8	1	3	4	1	1	3	10	7	4	3	4	2	2	2	10	5	3	3	7	2	2
A2	3	4	9	2	1	10	4	1	1	8	6	4	1	10	4	4	2	10	8	1	3	7	9	4

Gambar 2.2 Contoh Representasi Kromosom

2.7 Ekstraksi Vaksin

Proses ekstraksi vaksin dilakukan dalam dua tahap, yaitu ekstraksi vaksin M1 dan ekstraksi vaksin M2, yang keduanya masing masing akan menjadi satu vaksin utuh, sehingga dalam satu kali ekstraksi vaksin akan dihasilkan dua buah vaksin. Tiap vaksin yang dihasilkan akan disimpan dalam

immune memory untuk menggantikan vaksin yang memiliki fitness lebih rendah.

Pemilihan kandidat antibodi didasarkan pada antibodi dengan nilai *fitness* tertinggi dalam populasi saat itu. Dari antibodi terpilih akan diproses untuk ekstraksi baik untuk ekstraksi M1 maupun M2.

1. Ekstraksi M1

Langkah langkah ekstraksi M1 dapat diuraikan sebagai berikut,

- Ambil satu fragmen pada antibodi terpilih dengan metode roulette wheel
- Pasangkan fragmen terbaik tersebut dengan satu fragmen lainnya, dan hitung nilai *fitness* masing masing pasangan
- Ambil satu pasangan terbaik sebagai vaksin M1, simpan ke dalam memori vaksin

2. Ekstraksi M2

Langkah langkah ekstraksi tahap kedua dapat dijabarkan sebagai berikut,

- Cari nilai panjang fragmen dengan persamaan (2-1).

$$B_i = 2 + l \quad (2-1)$$

$$l = \begin{cases} 0, & r \leq 0.6 \\ 1, & 0.6 < r \leq 0.85 \\ 2, & 0.85 < r \leq 0.95 \\ 3, & 0.95 < r \end{cases}$$

- Pasangkan tiap fragmen dengan panjang pasangan B_i , dan hitung *fitness*-nya
- Ambil satu pasangan terbaik sebagai vaksin M2, simpan ke dalam memori vaksin

2.8 Crossover dan Mutasi

Metode *crossover* yang digunakan untuk penelitian ini ialah metode *one-point crossover*, yaitu memilih satu titik dalam kromosom sebagai *cut point*, lalu melakukan pertukaran gen antar kromosom mulai dari titik tersebut hingga sampai pada gen yang paling akhir.

A	2	3	4	9	2	1	0	4	1	1	8	6	4	1	0	4	4	2	0	8	1	3	7	9	4
A	4	2	8	7	3	2	0	1	1	3	2	2	2	1	5	7	4	1	4	5	1	1	1	2	2
O	1	3	4	9	2	1	0	4	1	1	2	2	2	1	5	7	4	1	4	5	1	1	1	2	2
O	1	3	4	9	2	1	0	4	1	1	2	2	2	1	5	7	4	1	4	5	1	1	1	2	2
2	2	8	7	3	2	0	1	1	3	8	6	4	1	0	4	4	2	0	8	1	3	7	9	4	

Gambar 2.3 Contoh Proses Crossover

Sedangkan metode mutasi yang dipilih untuk penelitian ini ialah *uniform mutation*. Proses mutasi dilakukan dengan memilih titik-titik mutasi, kemudian melakukan perubahan gen pada titik-titik tersebut sesuai dengan *range* gen yang mungkin dapat masuk kedalam posisi tersebut.

A						1																1		
4	2	8	7	3	2	0	1	1	3	2	2	2	1	5	7	4	1	4	5	1	1	1	2	2

O						1																1		
1	2	8	7	2	2	0	1	1	3	2	2	2	1	5	7	4	1	4	5	1	1	1	2	2

Gambar 2.4 Contoh Proses Mutasi

2.9 Vaksinasi

Antibodi yang akan divaksinasi dipilih secara random, begitupula dengan vaksin yang diambil dari *immune memory*. Antibodi terpilih akan diubah gennya sesuai dengan gen yang ada pada vaksin terpilih.

Untuk menjaga kualitas antibodi, jika hasil vaksinasi mengalami peningkatan nilai *fitness*, maka hasil vaksinasi akan menggantikan antibodi tersebut, sebaliknya jika nilai *fitness* hasil vaksinasi lebih rendah dari antibodi terpilih, maka hasil vaksinasi akan dibuang.

A1	3	8	8	1	3	4	1	1	3	10	7	4	3	4	2	2	2	10	5	3	3	7	2	2
V1	1	9	5	2	2	4	5	2																
A	1	9	5	2	2	4	5	2	3	0	7	4	3	4	2	2	2	0	5	3	3	7	2	2

Gambar 2.5 Contoh Proses Vaksinasi

2.10 Perhitungan Fitness

Tiap antibodi yang dihasilkan akan dihitung nilai *fitness*-nya sebagai tolak ukur seberapa baik antibodi tersebut terhadap antigen atau permasalahan. Nilai *fitness* tiap antibodi didapatkan dari jumlah penalti gizi dengan nilai variasi dan harga makanan.

Nilai fitness dapat dihitung dengan persamaan (2-2)

$$f(x) = \frac{1}{P+1} + \frac{1}{L+1} + \frac{1}{K+1} + \frac{1000}{B+1} + \frac{1}{S+1}$$

Dimana :

P = total selisih protein

L = total selisih lemak

K = total selisih karbohidrat

B = total selisih biaya

S = jumlah nilai kesamaan variasi kromosom

2.11 Seleksi Imun

Keberlangsungan hidup suatu antibodi ditentukan tak hanya oleh nilai *fitness* saja, tetapi juga memperhitungkan kesamaan antar antibodi yang ada pada populasi. Berikut langkah langkah untuk seleksi imun,

1. Mencari nilai kedekatan antar antibodi. Untuk mencari nilai kedekatan (*affinity*) antar antibodi, dapat digunakan persamaan (2-3)

$$A_{ij} = \frac{1}{1 + \sqrt{\sum_{v=1}^m (i_v - j_v)^2}} \quad (2-3)$$

Dimana A_{ij} merupakan nilai kedekatan antar antibodi i dan j , dan m merupakan banyak antibodi dalam satu populasi

2. Dari nilai kedekatan tersebut, dapat dicari nilai kesamaan (*similarity*) dengan persamaan (2-4)

$$C_i = \frac{\text{banyak antibodi yang nilai kedekatannya terhadap antibodi } i \text{ lebih besar daripada } \phi}{N} \quad (2-4)$$

Dimana N merupakan banyak antibodi dalam satu populasi, dan ϕ ialah konstanta *similarity*

3. Dengan mengombinasikan antara nilai *fitness* antibodi dengan nilai kesamaan antibodi, sehingga didapatkan probabilitas seleksi yang tak hanya memperhitungkan nilai *fitness* semata, tetapi juga mempertimbangkan kesamaan antar antibodi dalam populasi. Probabilitas seleksi dapat dihitung dengan persamaan (2-5)

$$P = \mu P_f + (1 - \mu) P_d \quad (2-5)$$

Dimana μ adalah konstanta *affinity*, P_f merupakan probabilitas berdasarkan nilai *fitness* yang dapat dicari dengan persamaan (2-6)

$$P_f = \frac{f_i}{\sum_{i=1}^N f_i} \quad (2-6)$$

Sedangkan P_d merupakan probabilitas berdasarkan nilai kesamaan yang dapat dicari dengan persamaan (2-7)

$$P_d = \frac{C_i}{\sum_{i=1}^N C_i} \quad (2-7)$$

Dimana N merupakan jumlah populasi, f_i adalah nilai *fitness* antibodi, dan C_i adalah nilai *similarity* antibodi.

4. Menghitung banyak antibodi yang akan bertahan ke generasi selanjutnya. Banyaknya antibodi yang akan bertahan ke generasi selanjutnya dapat dicari dengan menggunakan persamaan (2-8)

$$N_r = N - N_d = N(1 - \sigma PD) \quad (2-8)$$

Dimana N_r ialah jumlah antibodi yang akan bertahan ke generasi selanjutnya, N merupakan jumlah populasi saat ini, N_d adalah jumlah antibodi yang akan mati, σ adalah konstanta *death factor*, dan nilai PD dapat dicari dengan persamaan (2-9)

$$PD = \frac{f_{avg}}{f_{max}} \quad (2-9)$$

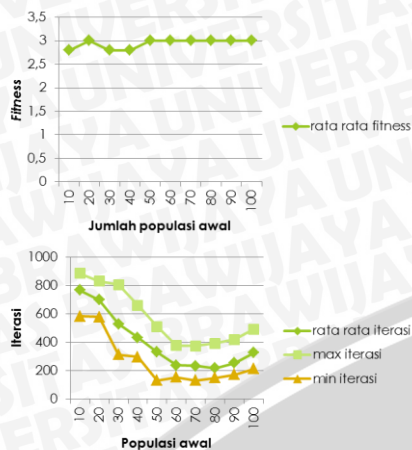
Dimana f_{avg} merupakan rata rata nilai *fitness* dalam populasi, dan f_{max} merupakan nilai *fitness* tertinggi dalam populasi

5. Proses seleksi dengan mengurutkan antibodi berdasarkan nilai probabilitas bertahan hidup yang telah dihitung. Diambil sebanyak N_r antibodi yang memiliki probabilitas paling tinggi sebagai kandidat untuk maju ke generasi selanjutnya

3. PENGUJIAN DAN ANALISIS

Pengujian dilakukan untuk mengetahui pengaruh parameter-parameter dari immune genetic algorithm terhadap hasil keluaran. Pengujian ini meliputi, pengujian jumlah populasi awal, probabilitas crossover, probabilitas mutasi, konstanta *affinity*, konstanta *similarity*, serta pengujian konstanta *death factor*. Pengujian dilakukan sebanyak 10 kali untuk setiap parameter yang diujikan. Setiap kali pengujian akan dicatat hasil *fitness* dan kecepatan konvergensinya.

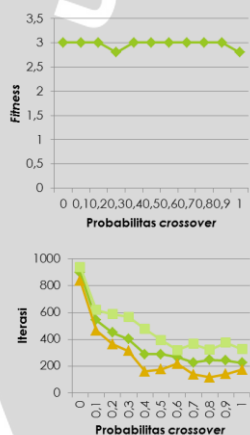
Hasil pengujian jumlah populasi awal dapat dilihat pada Gambar 3.1



Gambar 3.1 Hasil pengujian jumlah populasi awal

Hasil dari pengujian populasi awal menunjukkan bahwa semakin banyak jumlah populasi, akan semakin cepat terjadi konvergensi, tetapi nilai fitness lebih stabil pada jumlah popuasi awal yang kecil. Parameter terbaik diambil pada titik 70.

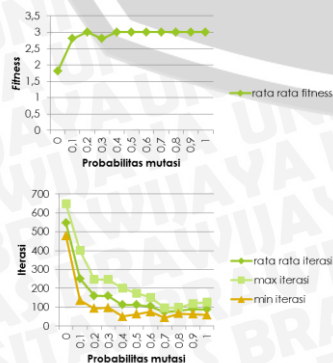
Hasil pengujian probabilitas crossover dapat dilihat pada Gambar 3.2



Gambar 3.2 Hasil pengujian probabilitas crossover

Pada pengujian probabilitas *crossover* juga menunjukkan semakin tinggi probabilitas, maka akan cenderung lebih cepat terjadi konvergensi. Parameter terbaik dipilih pada nilai 0,7

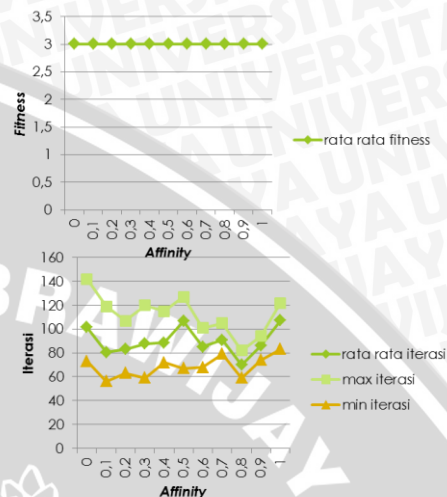
Hasil dari pengujian probabilitas mutasi dapat dilihat pada Gambar 3.3



Gambar 3.3 Hasil pengujian probabilitas mutasi

Dari pengujian probabilitas mutasi, terlihat kecepatan konvergensi dan nilai fitness cenderung membaik seiring semakin besarnya nilai probabilitas. Nilai yang paling optimal berada pada 0,7

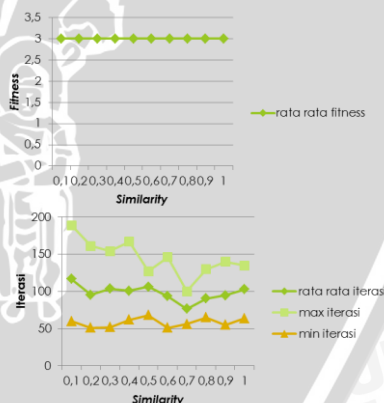
Hasil dari pengujian konstanta *affinity* dapat dilihat pada Gambar 3.4



Gambar 3.4 Hasil pengujian konstanta affinity

Konstanta *affinity* tidak terlalu berpengaruh pada nilai fitness, melainkan lebih berpengaruh pada kecepatan konvergensi. Nilai terbaik untuk konvergensi didapatkan pada titik 0,8

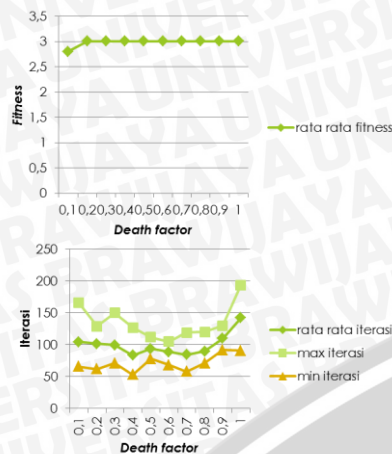
Hasil dari pengujian konstanta *similarity* dapat dilihat pada Gambar 3.5



Gambar 3.5 Hasil pengujian konstanta similarity

Nilai *similarity* tidak terlalu berpengaruh pada nilai *fitness*. Nilai terbaik berada pada konstanta 0,7

Sedangkan untuk hasil pengujian konstanta death factor dapat dilihat pada Gambar 3.6



Gambar 3.6 Hasil pengujian kontanta death factor

Hasil pengujian menunjukkan bahwa death factor tidak terlalu berpengaruh pada hasil nilai fitness. Konvergensi juga terlihat cukup stabil. Dipilih nilai 0,7 sebagai nilai paling optimal sebagai konstanta *death factor*

Dari hasil pengujian, didapatkan parameter paling optimal yaitu sebagai berikut,

1. Jumlah populasi awal = 70
2. probabilitas crossover = 0,7
3. probabilitas mutasi = 0,7
4. konstanta affinity = 0,8
5. konstanta similarity = 0,7
6. konstanta death factor = 0,7

Dengan parameter tersebut, didapatkan contoh hasil menu untuk 2 hari seperti yang ditampilkan pada Gambar 3.7

Hari 1			
Nasi putih	tumis kangkung	telur goreng	teh
Nasi Goreng	mie telur	usus ayam goreng	air putih
Nasi Kuning	sayur sop	tahu goreng	air putih
Biaya : Rp. 288.800			
Protein : 1945,60 kecukupan : 1,01			
Lemak : 2119,45 kecukupan : 0,91			
Karbohidrat : 8823,41 kecukupan : 0,93			
Hari 2			
Nasi Pecel	sayur asem	telur dadar	teh
Nasi putih	mie goreng sawi	pindang goreng	teh
Nasi Pecel	sayur bayam	dadar jagung	susu
Biaya : Rp. 302.100			
Protein : 2033,38 kecukupan : 1,06			
Lemak : 2497,74 kecukupan : 1,08			
Karbohidrat : 8570,60 kecukupan : 0,90			

Gambar 3.7 Contoh hasil menu

Jika dibandingkan dengan data menu dari panti asuhan pada Gambar 3.8

Hari 1			
Nasi Pecel	-	Tahu goreng	Teh
Nasi	Tumis kacang panjang	Tahu goreng	Sirup
Nasi	Mie Goreng Sawi	Pindang goreng	Susu
Biaya : Rp. 209.000			
Protein : 2633,4 kecukupan : 1,37			
Lemak : 1710,38 kecukupan : 0,73			
Karbohidrat : 8230,87 kecukupan : 0,86			
Hari 2			
Nasi	Tumis kangkung	Tempe goreng	Teh
Nasi	Sayur sop	Telur dadar	Air putih
Nasi	Rawon	Tempe goreng	Air putih
Biaya Rp. 214.700			
Protein : 1187,88 kecukupan : 0,61			
Lemak : 1174,96 kecukupan : 0,50			
karbohidrat : 6230,86 kecukupan : 0,65			

Gambar 3.8 Menu dari panti asuhan

Terlihat menu hasil dari algoritma ini dapat memberikan kecukupan gizi yang lebih baik daripada menu dari panti asuhan (diatas 90% kebutuhan gizi menurut AKG). Akan tetapi, meskipun nilai gizinya telah mencukupi kebutuhan anak sesuai dengan AKG, tetapi biaya yang dihasilkan dari algoritma ini masih cukup tinggi jika dibandingkan dengan menu aslinya (mendekati atau bahkan lebih dari Rp. 300.000 per hari jika dibandingkan menu dari panti asuhan yang hanya berkisar pada Rp. 200.000).

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

1. *Immune Genetic Algorithm* dapat digunakan untuk mengoptimasi gizi pada anak panti asuhan dengan cara merepresentasikan solusi yang berupa menu makanan dan biayanya sebagai kromosom yang akan diproses oleh algoritma. Satu kromosom akan merepresentasikan menu santapan dalam bentuk deretan nomor indeks makanan sehingga membentuk satu menu makanan untuk beberapa hari sekaligus, lengkap untuk makan pagi, siang, maupun malam dengan masing-masing mengandung makanan berupa makanan pokok, sayur, lauk, serta minuman. Kromosom terbaik dilihat dari pemenuhan gizi, kombinasi menu, dan biaya yang paling minimal, akan dijadikan sebagai rekomendasi menu makanan untuk optimasi gizi pada anak panti asuhan.
2. Hasil dari pengujian menunjukkan bahwa semakin banyak jumlah populasi, akan semakin cepat terjadi konvergensi, tetapi nilai *fitness* terbaik didapatkan pada jumlah populasi awal 70. Pada pengujian probabilitas *crossover* juga menunjukkan semakin tinggi probabilitas, maka akan cenderung lebih cepat terjadi konvergensi, hanya saja jika nilainya terlalu besar akan berakibat pada semakin banyaknya terjadi konvergensi prematur. Nilai paling optimal untuk probabilitas *crossover* berada pada nilai 0,7. Dari pengujian probabilitas mutasi, terlihat kecepatan konvergensi dan nilai *fitness*

cenderung membaik seiring semakin besarnya nilai probabilitas. Nilai paling optimal untuk probabilitas mutasi ialah 0,7. Konstanta *affinity* tidak terlalu berpengaruh pada nilai *fitness*, melainkan lebih berpengaruh pada kecepatan konvergensi. Konvergensi terbaik terjadi pada nilai konstanta *affinity* 0,8. Untuk konstanta *similarity* menunjukkan nilai *fitness* yang stabil pada nilai 0,7. Sedangkan untuk konstanta *death factor*, hasil pengujian menunjukkan bahwa *death factor* tidak terlalu berpengaruh pada hasil nilai *fitness*. Konvergensi juga terlihat cukup stabil, nilai paling optimal berada pada 0,7. Dengan nilai parameter tersebut, hasil dari algoritma menunjukkan jika menu yang dihasilkan sudah dapat memberi kecukupan gizi yang baik, hanya saja biaya yang dihasilkan masih cukup tinggi.

4.2 Saran

1. Pada perhitungan nilai *fitness*, dapat dilakukan perbaikan untuk lebih menyeimbangkan antara nilai gizi dengan harga atau biaya, sehingga dapat tetap memberikan rekomendasi menu yang dapat memenuhi gizi dengan biaya yang lebih minimal.
2. Pada proses seleksi, dapat dilakukan perbaikan agar perubahan jumlah populasi tidak terlalu jauh dari jumlah populasi awal, sehingga meminimalkan kemungkinan program tidak berjalan dengan baik.
3. Penelitian ini lebih mengutamakan kecukupan gizi harian untuk satu populasi, diharapkan untuk penelitian selanjutnya lebih memperhatikan kebutuhan gizi tiap individu tiap kali makan sehingga menu yang dihasilkan dapat lebih optimal untuk kebutuhan harian, maupun kebutuhan tiap kali makan baik untuk keseluruhan populasi maupun tiap individu.
4. Data menu makanan yang digunakan sebaiknya lebih lengkap dan bervariasi agar sistem memiliki lebih banyak pilihan, sehingga diharapkan hasil keluaran menu lebih optimal.
5. Lebih memperhatikan kombinasi menu makanan agar dapat menghasilkan kombinasi menu makanan yang lebih realistis dengan tetap memberi kecukupan gizi yang baik serta biaya yang minimum.

5. DAFTAR PUSTAKA

Brownlee, J., 2011. *Clever Algorithms: Nature-Inspired Programming Recipes*. First penyunt. s.l.:LuLu.

Indonesia, 2011. *Peraturan Menteri Sosial No. 3 Tentang Standar Nasional Pengasuhan Anak Untuk Lembaga Kesejahteraan Sosial Anak*, Jakarta: s.n.

Indonesia, 2013. *Peraturan Menteri Kesehatan Republik Indonesia Nomor 75 Tahun 2013 Tentang Angka Kecukupan Gizi yang Dianjurkan bagi Bangsa Indonesia*, Jakarta: s.n.

Jin, Z. & Fan, H., 2013. A New Immune Genetic Algorithm for 0-1 Knapsack Problem. *Sixth International Symposium on Computational Intelligence and Design*, pp. 31-33.

Juan, S., Dan, S. Y. & Min, X., 2009. *Research on Application of IGA (Immune Genetic Algorithm) to the Solution of Course-Timetabling Problem*. s.l., IEEE, pp. 1105-1109.

Lei, W. & Licheng, J., 2000. *Immune Evolutionary Algorithms*. Beijing, IEEE, pp. 1655 - 1662.

Muchtadi, D., 2014. *Pengantar Ilmu Gizi*. Bandung: Alfabeta.

Wang, L., Jiao, L. & Member, S., 1998. *The Immune Genetic Algorithm and Its Convergence*. Beijing, IEEE, pp. 1347-1350.

William, A., 2010. *Gambaran Status Gizi Anak di Panti Asuhan Yayasan Terima Kasih Abadi Kecamatan Medan Barat Tahun 2010*, Medan: s.n.

Yan-cong, Z. et al., 2011. *Research on Test Paper Auto-Generating Based on Immune Genetic Algorithm*. Mianyang, IEEE, pp. 2498 - 2502.