

## Optimasi Komposisi Menu Makanan bagi Penderita Penyakit Diabetes Melitus Tipe 2 dan Komplikasinya menggunakan *Hybrid* Algoritme Genetika dan *Simulated Annealing*

Muhammad Jibril Alqarni<sup>1</sup>, Imam Cholissodin<sup>2</sup>, Bayu Rahayudi<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya

Email: <sup>1</sup>mjibrilqarni@gmail.com, <sup>2</sup>imamcs@ub.ac.id, <sup>3</sup>ubay1@ub.ac.id

### Abstrak

Diabetes Melitus merupakan penyakit yang sudah umum dikalangan masyarakat. Salah satu hal preventif aktif untuk menangani penyakit diabetes melitus tipe 2 yaitu dengan melakukan olahraga teratur, dan makan makanan yang bergizi serta memiliki nutrisi yang cukup untuk 1 harinya. Dalam mendapatkan kalori atau energi yang cukup sesuai kebutuhan dari penderita, perhitungan dapat dilakukan secara manual. Namun jika proses dilakukan secara manual, akan memakan waktu yang lama sehingga jika hal ini diterapkan di lembaga kesehatan, akan sangat tidak efisien mengingat banyaknya pasien yang berada di antrian. Permasalahan ini dapat diselesaikan dengan menggunakan sistem kecerdasan buatan menggunakan algoritme genetika yang dilakukan hibridasi dengan *simulated annealing*. *Simulated annealing* dapat membantu algoritme genetika keluar dari kondisi lokal optimum, dikarenakan sifatnya yang dapat menerima solusi yang tidak lebih baik ataupun lebih baik dari solusi sebelumnya. *Simulated annealing* berhasil ditambahkan dalam membantu algoritme genetika keluar dari kondisi lokal optimum hal ini ditandai dengan didapatkannya nilai *fitness* tertinggi sebesar 0,998, dengan persentase selisih terhadap kebutuhan aktual pasien dari kalori sebesar 0,18%, karbohidrat sebesar 0,20%, protein sebesar 0,69% dan terakhir yaitu lemak sebesar 0,27%.

**Kata kunci:** diabetes tipe 2, algoritme genetika, optimasi, simulated annealing, adaptif cr mr

### Abstract

*Diabetes mellitus is a common disease among the people. One of the active preventive things to deal with type 2 diabetes mellitus is to do regular exercise, and eat nutritious foods and have adequate nutrition for 1 day. In getting enough calories or energy according to the needs of patients, calculations can be done manually. But if the process is done manually, it will take a long time so that if this is implemented in a health institution, it will be very inefficient given the large number of patients in the queue. This problem can be solved by using an artificial intelligence system using a genetic algorithm that is performed hybridization with simulated annealing. Simulated Annealing can help the genetic algorithm come out of optimum local conditions, due to its nature that can accept solutions that are not better or better than the previous solution. Simulated Annealing was successfully added to help the genetic algorithm out of optimum local conditions, this was indicated by the highest fitness value of 0.998, with the percentage difference between the patient's actual needs of calories by 0.18%, carbohydrate by 0.20%, protein by 0.69% and the last is fat at 0.27%.*

**Keywords:** diabetes type 2, genetic algorithm, optimization, simulated annealing, adaptive cr mr

### 1. PENDAHULUAN

Penyakit yang sudah umum dikalangan masyarakat salah satunya adalah Diabetes Melitus. Penyakit ini terlalu sering dianggap kecil, dan akibatnya dapat menyebabkan komplikasi penyakit lain bahkan hingga dapat menyebabkan kematian (Kementrian Kesehatan

RI, 2018). Diabetes merupakan masalah penting bagi kesehatan yang ada pada masyarakat, dimana diabetes merupakan salah satu dari empat penyakit yang tidak menular lainnya yang menjadi prioritas utama para pemimpin dunia saat ini (Kementrian Kesehatan RI, 2018). Diabetes sendiri telah menjadi penyebab kematian atau cause of death dari 1.5 juta

kematian pada tahun 2012 (Kementerian Kesehatan RI, 2018).

Di Indonesia sendiri pada tahun 2000, terdapat sebanyak 8.4 juta jiwa yang terkena penyakit diabetes, yaitu berada pada peringkat 4 dunia. Diperkirakan pada tahun 2030, Indonesia tetap berada pada peringkat 4 dunia dengan perkiraan terdapat 21.3 juta jiwa yang akan terkena penyakit diabetes (Kementerian Kesehatan RI, 2018). Diabetes sendiri terdiri atas 2 tipe yaitu diabetes tipe 1 dan tipe 2. Untuk diabetes tipe 1, sangat disayangkan belum dapat dicegah dengan ilmu kedokteran saat ini. Meskipun begitu, khusus untuk diabetes tipe 2, pendekatan yang efektif dapat menjadi solusi utama dalam pencegahan diabetes tipe 2 serta dapat mencegah komplikasi dan kematian yang premature yang bisa disebabkan oleh berbagai tipe diabetes (Kementerian Kesehatan RI, 2018).

Banyak penelitian telah melakukan research tentang diet untuk penyakit diabetes, akan tetapi mereka tidak memasukkan parameter yang cukup lengkap untuk perhitungan jumlah kalori untuk satu hari. Mereka hanya melakukan klasifikasi makanan, mana yang boleh untuk dimakan bagi penderita penyakit diabetes mellitus. Pada 2010 (Phanich, et al., 2010) menggunakan *Self-Organizing Map* (SOM) dan *K-means Clustering* untuk melakukan klustering pada makanan yang diperbolehkan untuk penyakit diabetes. Penelitian selanjutnya pada tahun 2011, melakukan penjadwalan diet untuk pasien kanker yang dilakukan oleh (Husain, et al., 2011) menggunakan Algoritme Genetika.

Pada penelitian lain yang menyangkut permasalahan yang sama menghasilkan kombinasi menu pangan yang dibutuhkan oleh penderita diabetes dengan biaya yang minimum, dimana output yang dihasilkan berupa makan pagi, makan siang serta makan malam (Rianawati & Mahmudy, 2015).

Sumber penelitian terakhir yang dilakukan oleh (Anggarsari, et al., 2017) dengan menggunakan algoritme genetika dilakukan *hybrid* menggunakan *simulated annealing* untuk gizi balita bisa dikatakan berhasil mencapai hasil yang optimum. Dengan menggunakan jumlah generasi sebanyak 50, lalu ukuran populasi sebanyak 100 lalu *cr* sebesar 0,8 serta *mr* 0,3, lalu untuk nilai dari *simulated annealing* dari *alpha* 0,8, temperatur awal 2 dan temperatur akhir 0.2. Hasil akhir menunjukkan selisih yang kecil antara solusi yang diberikan sistem terhadap kebutuhan gizi sebenarnya.

Fokus dari penelitian yang dilakukan ini

adalah dalam penggunaan metode *simulated annealing* pada optimasi komposisi menu makanan bagi penderita penyakit diabetes melitus tipe 2 dan komplikasinya untuk membantu permasalahan yang ada pada algoritme genetika yaitu cenderung mendapatkan solusi yang konvergensi dini dan juga untuk operator *crossover* dan mutasi pada proses reproduksi akan dilakukan secara adaptif, yakni tiap iterasi nilai *cr mr* akan berubah menyesuaikan jumlah generasi dari nilai *fitness* yang didapatkan pada iterasi sebelumnya. Dari digunakannya *hybrid* algoritme genetika dan *simulated annealing* pada optimasi komposisi menu makanan bagi penderita penyakit diabetes melitus tipe 2 dan komplikasinya, diharapkan keluaran mampu menghasilkan solusi menu makanan yang lebih baik dengan waktu komputasi yang tidak terlalu lama dibandingkan dengan hanya menggunakan algoritme genetika.

## 2. METODE PENELITIAN

Dalam melakukan penelitian ini, ada beberapa metode perhitungan yang harus dipahami sebelumnya. Perhitungan gizi pasien, merupakan salah satu hal yang penting dalam menentukan solusi akhir yang akan didapatkan nantinya menggunakan metode pencarian solusi.

Hal terpenting dalam diet tentu adalah komposisi dari makanan yang benar dan tepat yaitu bagaimana cara yang tepat untuk mengoptimalkan gizi serta nutrisi pada makanan yang akan dikonsumsi oleh seorang penderita penyakit diabetes melitus. Untuk penderita penyakit diabetes melitus tipe 2 dan komplikasinya, muncul masalah seperti sedikitnya pilihan makanan yang bisa dimakan dibandingkan orang yang tidak terkena penyakit diabetes melitus tipe 2.

Diabetes melitus adalah kumpulan beberapa gejala yang muncul pada manusia yang disebabkan oleh munculnya peningkatan glukosa darah akibat kurangnya jumlah insulin baik absolut maupun relatif (Suyanto, 2007). Diabetes juga bisa dikatakan kondisi pada tubuh tidak dapat memproduksi insulin dengan baik sehingga glukosa yang ada dalam darah menjadi menumpuk (Sutanto, 2010).

Menurut (Perkumpulan Endokrinologi Indonesia, 2011) komposisi yang disarankan terdiri dari karbohidrat dengan presentase dari kebutuhan kalori 45-65%, lemak sebesar 12-15%, dan protein 15-20%.

### 2.1 Perhitungan Kebutuhan Gizi Pasien

Berdasarkan (Perkumpulan Endokrinologi Indonesia, 2011) terdapat berbagai cara untuk dapat menentukan jumlah kalori yang dibutuhkan penderita diabetes mellitus tipe 2, yaitu dengan melakukan perhitungan kebutuhan kalori basal yang besarnya antara 25-30kal/kg BB ideal. Kasus perhitungan pada penelitian ini akan digunakan pasien dengan data pada Tabel 1.

Tabel 1 Data Pasien

Nama	AD
Umur	52 tahun
Berat Badan	65 kg
Tinggi Badan	165 cm
Level Aktifitas	Sedang Istirahat
Jenis Kelamin	Laki-Laki
Komplikasi	Jantung

Pertama akan dilakukan perhitungan berat badan ideal (BBI). Dikarenakan jenis kelamin dari pasien adalah laki laki dan tinggi badan > 165 maka Persamaan yang digunakan adalah Persamaan 1.

$$BBI = ((Tinggi\ Badan - 100) \times 1kg) \times 0,9 \quad (1)$$

Sehingga dari Persamaan 1, didapatkan perhitungan sebagai berikut:

$$BBI = ((165 - 100) \times 1kg) \times 0,9 = 58,5\ kg/m^2$$

Setelah mendapatkan nilai BBI, akan dilakukan perhitungan kebutuhan basal kalori berdasarkan jenis kelamin dan BBI, menggunakan Persamaan 2.

$$Kalori\ Basal = BBI \times 30\ kcal \quad (2)$$

Dari Persamaan 2, berdasarkan informasi pada Tabel 1, didapatkan perhitungan sebagai berikut:

$$Kalori\ Basal = 58,5 \times 30\ kcal = 1755$$

Dari kebutuhan basal kalori, akan dilakukan perhitungan faktor yang akan mempengaruhi kebutuhan energi akhir. Pertama adalah faktor umur. Pasien berusia 52 tahun maka, Persamaan perhitungannya menggunakan Persamaan 3.

$$Umur = Kalori\ Basal \times -5\% \quad (3)$$

Dari persamaan 3 didapatkan perhitungan sebagai berikut:

$$Faktor\ Umur = 1755 \times (-5\%) = -87,75$$

Faktor selanjutnya adalah faktor berat badan, sebelum menghitung faktor berat badan, akan ditentukan kategori berat badan. Untuk menentukan kategori berat badan akan dilakukan perhitungan indeks masa tubuh (IMT). Status berat badan bisa dilihat pada Tabel 2.

Tabel 2 Status Berat Badan

BB Kurang	IMT < 18,5
BB Normal	18,5 = IMT <= 22,9
BB Lebih	IMT >= 23
BB Obesitas	IMT >= 30

Perhitungan menghitung IMT bisa dilihat pada Persamaan 4.

$$IMT = \frac{Berat\ Badan}{Tinggi\ Badan^2} \times 100 \quad (4)$$

Berdasarkan Persamaan 4 didapatkan perhitungan sebagai berikut:

$$IMT = \frac{65\ kg}{1,65\ m^2} = 23,89$$

Setelah mendapatkan nilai indeks masa tubuh, akan dilakukan perbandingan nilai IMT terhadap Tabel 2. Berdasarkan Tabel 2 didapatkan bahwa pasien termasuk kategori BB Lebih. Maka dari itu, akan dilakukan perhitungan dengan Persamaan 5.

$$Faktor\ BB = Kalori\ Basal \times -(0,2) \quad (5)$$

Berdasarkan Persamaan 5 didapatkan perhitungan sebagai berikut:

$$Faktor\ BB = 1755 \times (-0,2) = -351$$

Selanjutnya yaitu menghitung faktor level aktifitas, berikut kondisi dari tingkat aktifitas pada Tabel 3.

Tabel 3 Faktor Tingkat Aktifitas

Kalori Basal * 10%	Keadaan Istirahat
Kalori Basal * 20%	Ringan: Pegawai kantor, guru, ibu rumah tangga)
Kalori Basal * 30%	Sedang: Pegawai industri, mahasiswa, militer yang sedang tidak perang).

Kalori Basal * 40%	Berat: Petani, butuh, atlet, militer dalam keadaan latihan.
Kalori Basal * 50%	Sangat Berat: Tukang Becak, tukang gali.

Berdasarkan Tabel 3, tingkat aktifitas pasien berada pada tingkat sedang istirahat, hal demikian akan digunakan Persamaan 6.

$$Aktifitas = Kalori Basal \times 10\% \quad (6)$$

Berdasarkan Persamaan 6, didapatkan perhitungan sebagai berikut:

$$Faktor Aktifitas = 1755 \times (10\%) = 175,5$$

Kebutuhan energi akhir akan dihitung jika semua nilai faktor yang menentukan kebutuhan energi akhir telah didapatkan. Perhitungan kebutuhan energi akhir dapat dilihat pada Persamaan 7.

$$Kalori = (Kalori Basal + Aktifitas) + Faktor Umur + Faktor BB \quad (7)$$

Berdasarkan Persamaan 7 didapatkan perhitungan sebagai berikut:

$$Kebutuhan Energi = 1755 + 175,5 + (-87,75) + (-351) = 1491,75$$

Kebutuhan nutrisi akan didapatkan berdasarkan persentase dari kebutuhan energi, perhitungannya menggunakan Persamaan 8 hingga Persamaan 10.

$$Karbohidrat = Kalori * 0,65 \quad (8)$$

$$Protein = Kalori * 0,15 \quad (9)$$

$$Lemak = Kalori * 0,20 \quad (10)$$

Berdasarkan Persamaan 8 hingga Persamaan 10, didapatkan perhitungan sebagai berikut:

$$Karbohidrat = 1491 * 0,45 = 242,40 \text{ gr}$$

$$Protein = 1491 * 0,20 = 55,94 \text{ gr}$$

$$Lemak = 1491 * 0,15 = 33,15 \text{ gr}$$

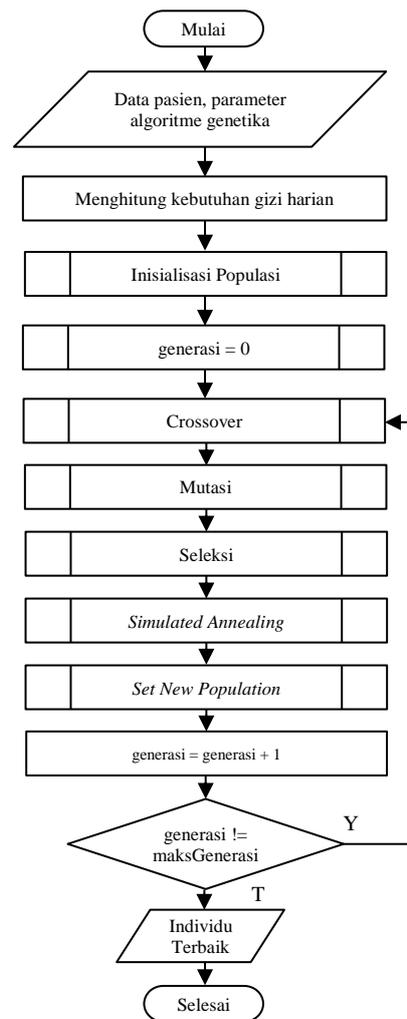
### 2.2 Hybrid Algoritme Genetika Simulated Annealing

Algoritme genetika memiliki kelemahan yang cenderung mendapatkan konvergensi yang prematur (Mahmudy & Rahman, 2011). Sedangkan *simulated annealing* memiliki kelemahan yang mana, hanya bisa menyimpan 1 solusi. Berdasarkan 2 kelemahan metode ini, akan digabung kedua metode dalam proses melakukan pencarian, yang mana kerja algoritme genetika adalah melakukan pencarian

solusi baru eksplorasi dan eksploitasi sedangkan untuk *simulated annealing* akan membantu algoritme genetika untuk keluar dari solusi lokal optimum (Sofianti, 2004). Diagram alir ditunjukkan pada Gambar 1.

Adapun proses dari *hybrid GASA* adalah sebagai berikut (Anggarsari, et al., 2017):

1. Melakukan masukan parameter yang dibutuhkan untuk proses GA dan *simulated annealing*.
2. Inialisasi populasi sebanyak *n* individu.
3. Melakukan proses reproduksi dengan operator *crossover* dan *mutation*.
4. Melakukan perhitungan *fitness*, dan seleksi menggunakan *elitism*.
5. Masuk ke proses *simulated annealing*, mengambil 1 individu terbaik pada populasi di generasi yang sama.
6. Proses terus berulang hingga jumlah iterasi mencapai maksimum.



Gambar 1 Diagram Alir *Simulated Annealing*

Langkah pertama adalah melakukan perhitungan kebutuhan gizi dari pasien. Setelah

melakukan perhitungan kebutuhan gizi, akan didapatkan kebutuhan gizi yang terdiri dari kalori, karbohidrat, protein dan lemak. Setelah itu akan dilakukan inialisasi populasi sebanyak  $n$  individu.

Kromosom terdiri dari 15 gen, masing-masing gen mewakili indeks dari makanan atau pada *database* merupakan kolom ID\_MAKANAN. Kromosom terdiri dari 3 bagian, 1 bagian terdiri dari 5 jenis makanan yang berbeda, masing-masing bagian terdiri dari makan pagi, makan siang dan makan malam. Bisa dilihat pada Tabel 4 contoh representasi kromosom:

Tabel 4 Representasi Kromosom

Makan Pagi					Makan Malam					
SP	SS	SN	P	SH	SN	P	SH	SN	P	SH
1	19	45	56	79	...	...	...	...	...	...

Keterangan:

- SP : Sumber Pokok
- SS : Sumber Sayuran
- SN : Sumber Nabati
- P : Pelengkap
- SH : Sumber Hewani

Setelah mengetahui representasi dari kromosom, akan dilakukan proses inialisasi sebanyak 3 individu sebagai contoh bisa dilihat pada Tabel 5.

Tabel 5 Inialisasi Populasi

P	Makan Pagi				
	SP	SS	SN	P	SH
P1	1	19	45	56	79
P2	14	19	42	52	79
P3	12	25	49	56	93

Keterangan:

$P_i$  : Individu ke- $i$

Dari individu yang ada, akan dilakukan proses reproduksi menggunakan operator *crossover* dan *mutation*. Metode yang digunakan dalam melakukan proses *crossover* yaitu menggunakan *extended immediate crossover*. Persamaan metode bisa dilihat pada Persamaan 11 dan Persamaan 12 (Mahmudy, 2013).

$$C1 = P1 + a(P2 - P1) \quad (11)$$

$$C2 = P2 + a(P1 - P2) \quad (12)$$

Keterangan :

- $C_i$  : Nilai gen baru kromosom anak
- $P_i$  : Nilai gen kromosom induk
- $a$  : Nilai random dengan *range* [0,1; 0,9]

Diasumsikan bahwa, nilai *alpha* adalah 0,2 sehingga didapatkan *offspring* dari hasil *crossover* sebagaimana pada Tabel 6.

Tabel 6 *Offspring* Hasil *Crossover*

P	Makan Pagi				
	SP	SS	SN	P	SH
C1	4	19	44	55	79
C2	11	19	43	53	79

Selanjutnya adalah proses *mutation*, metode yang digunakan dalam proses ini adalah *random mutation* (Sivanadam & Deepa, 2007). Proses dilakukan dengan memilih salah satu gen dari kromosom dan mengubahnya menggunakan Persamaan 13.

$$X_i' = X_i + r(\text{maks}X_i - \text{min}X_i) \quad (13)$$

Keterangan:

- $X_i$  : Nilai gen yang akan dirubah.
- $r$  : Bilangan random dengan *range* [-0,4; 0,4].
- $\text{maks}X_i$  : Nilai maksimal gen.
- $\text{min}X_i$  : Nilai minimal gen.

Diasumsikan bahwa nilai  $r$  adalah 0,01 sehingga *offspring* yang dihasilkan adalah sebagaimana pada Tabel 7.

Tabel 7 *Offspring* Hasil Mutasi

P	Makan Pagi				
	SP	SS	SN	P	SH
C3	12	25	49	56	93

Setelah didapatkan *offspring* dari proses reproduksi akan dihitung semua nilai *fitness* yang ada pada populasi menggunakan Persamaan 14.

$$f = \frac{\sum_{k=1}^4 (1 - \text{norGizi}_k)}{4} \quad (14)$$

Perhitungan *fitness* didapatkan dari nilai *pinalti*, yang mana *pinalti* merupakan nilai selisih antara kebutuhan aktual gizi pasien dengan hasil nutrisi menu makanan Persamaan yang digunakan dari Persamaan 15.

$$pinGizij = | kebGizik - giziMakanan_k | \quad (15)$$

Dikarenakan terdapat komplikasi pada penelitian ini mengenai diabetes melitus, pada penelitian ini diasumsikan bahwa pasien hanya memiliki 1 komplikasi, perhitungan bobot untuk pinalti setiap kebutuhan, dapat dilihat pada Tabel 8.

Tabel 8 Perhitungan Bobot Komplikasi

Jika Komplikasi, Jantung, stroke, mata, syaraf dan lambung.	(3*pinKal), (1*pinKarb), (1*pinPro), (3*pinLem)
Jika Komplikasi, Ginjal	(3*pinKal), (1*pinKarb), (3*pinPro), (1*pinLem)

Setelah dilakukan perhitungan nilai pinalti dari kebutuhan gizi dengan kandungan gizi makanan, akan dilakukan proses normalisasi. Perhitungan normalisasi mengacu pada Persamaan 16. Berikut adalah perhitungan dari proses normalisasi:

$$norGizik = \frac{pinGizik}{domainGizik} \quad (16)$$

Keterangan:

- $k$  : {Kalori, Karbohidrat, Protein, Lemak}
- pinGizi : Pinalti Gizi Makanan
- kebGizi : Kebutuhan gizi pasien
- giziMakanan : Kandungan gizi makanan
- norGizi : Normalisasi gizi
- domainGizi : Batas atas jenis gizi

Setelah didapatkan nilai pinalti yang telah dinormalisasi, selanjutnya akan dihitung menggunakan Persamaan 14. Proses selanjutnya setelah melakukan evaluasi adalah seleksi. Metode seleksi yang digunakan dalam penelitian ini adalah *elitism*. Metode dengan melakukan pengurutan berdasarkan nilai *fitness* tertinggi hingga terendah.

Proses selanjutnya adalah masuk ke bagian *simulated annealing*. Pada tahapan ini, akan diambil satu solusi yakni solusi terbaik pada generasi yang sama dan akan dijadikan sebagai inialisasi solusi awal untuk dilakukan modifikasi kromosom. Berikut adalah proses dari *simulated annealing*:

1. Perulangan dilakukan selama temperatur tinggi hingga mencapai titik beku atau rendah.
2. Lakukan juga perulangan sebanyak  $N$ .

3. Melakukan pencarian solusi tetangga dengan memodifikasi solusi terbaik.
4. Menghitung selisih *fitness* antara solusi terbaik sekarang dengan solusi yang telah dimodifikasi.
5. Jika solusi modifikasi tidak lebih baik dari solusi sekarang, solusi yang sekarang akan diganti menjadi solusi modifikasi, hal ini dilakukan pencarian *downhill move*.
6. Namun apabila solusi modifikasi lebih baik, akan masuk ke kondisi pengecekan menggunakan *probabilitas boltzmann* dengan membandingkan apakah nilai *probabilitas boltzmann* memiliki nilai lebih dari bilangan acak [0;1] yang dimunculkan. *Probabilitias Boltzmann* dihitung menggunakan Persamaan 17.

$$Prob Boltzmann = e^{-\frac{\Delta E}{T}} \quad (17)$$

Keterangan:

- $e$  : Bilangan *exp*
- $\Delta E$  : Selisih *fitness* solusi terbaik sekarang dengan solusi modifikasi
- $T$  : Temperatur awal

7. Jika proses pengecekan probabilitas *boltzmann* telah selesai dilakukan, ganti solusi sekarang dengan solusi modifikasi.
8. Jika telah selesai melakukan perulangan sebanyak  $N$ , akan dilakukan perhitungan ulang temperatur, menggunakan nilai alpha.
9. Jika proses iterasi *simulated annealing* telah selesai akan dilanjutkan proses *GA* di iterasi atau generasi selanjutnya.

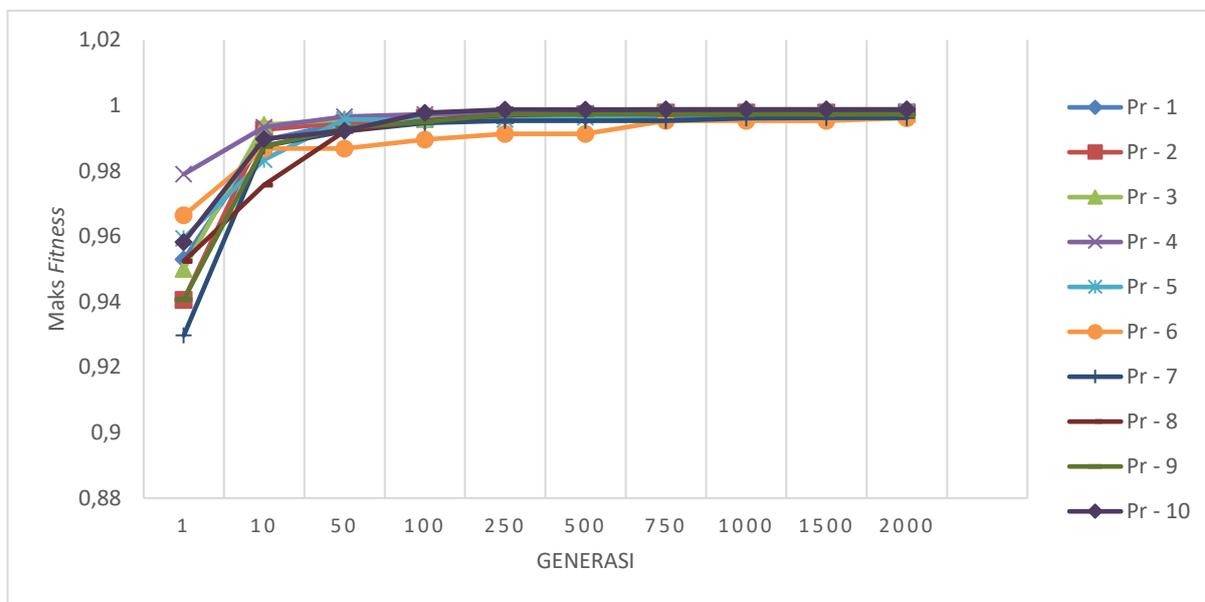
### 3. HASIL DAN PEMBAHASAN

Pada penelitian ini terdapat 4 jenis pengujian, yang pertama yaitu pengujian konvergensi, yang kedua yaitu pengujian jumlah populasi, selanjutnya ada pengujian *cr mr* adaptif, dan terkakhir adalah pengujian nilai *alpha*. Pada pengujian ini dilakukan sebanyak masing-masing pengujian sebanyak 10 kali. Parameter yang digunakan juga berbeda beda tiap jenis pengujian. Tujuan dari pengujian ini yakni, untuk melihat parameter kontrol terbaik yang didapatkan berdasarkan hasil nilai rata-rata *fitness* dari hasil pengujian yang telah dilakukan.

### 3.1 Pengujian Konvergensi

Pada pengujian konvergensi ini bertujuan untuk mengetahui pada generasi berapa *Hybrid GASA* mencapai titik konvergensi.. Adapun parameter yang digunakan untuk melakukan pengujian yaitu, jumlah populasi = 10  $cr = 0,1$   $mr = 0,1$   $T_0 = 30$   $Takhir = 0,2$ . Hasil pengujian dari konvergensi dapat dilihat pada Gambar 2.

Pengujian dimulai dari jumlah populasi sebesar 10 hingga mencapai jumlah populasi 100. Adapun parameter yang digunakan untuk melakukan pengujian yaitu, jumlah generasi=500  $cr = 0,1$   $mr = 0,1$   $\alpha = 0,8$   $T_0 = 30$   $Takhir = 0,2$ . Hasil pengujian dari jumlah populasi bisa dilihat pada Gambar 3.



Gambar 2 Grafik Hasil Pengujian Konvergensi

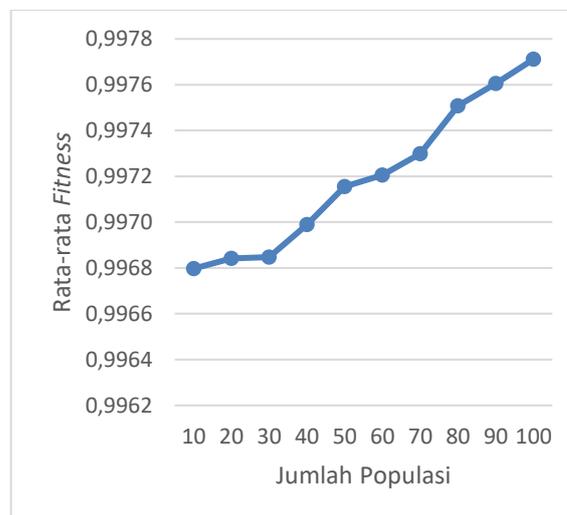
Keterangan:

Pr<sub>i</sub> : Percobaan ke-*i*

Berdasarkan Gambar 2 dapat dilihat bahwa konvergensi mulai terjadi pada generasi ke-250 hingga 2000. Dapat diambil kesimpulan bahwa, semakin besar nilai dari jumlah generasi atau iterasi, semakin besar pula nilai *fitness* yang akan didapatkan namun apabila nilai *fitness* di iterasi awal dan nilai *fitness* di iterasi yang sudah mencapai konvergensi memiliki selisih yang sangat kecil, kemungkinan hal ini merupakan lokal optimum. Hal ini dapat dikatakan solusi yang dicapai oleh kombinasi dari algoritme genetika dan *simulated annealing* berhasil keluar dari lokal optimum. Hal ini dapat terjadi dikarenakan sifat dari *simulated annealing* yang bisa menerima solusi terburuk dan terbaik, untuk mencari lokal minimum dan global minimum.

### 3.2 Pengujian Jumlah Populasi

Pada pengujian jumlah populasi ini bertujuan untuk mengetahui pada jumlah populasi berapa yang memungkinkan untuk mendapatkan solusi pencarian yang optimum. Pengujian ini dilakukan sebanyak 10 kali.



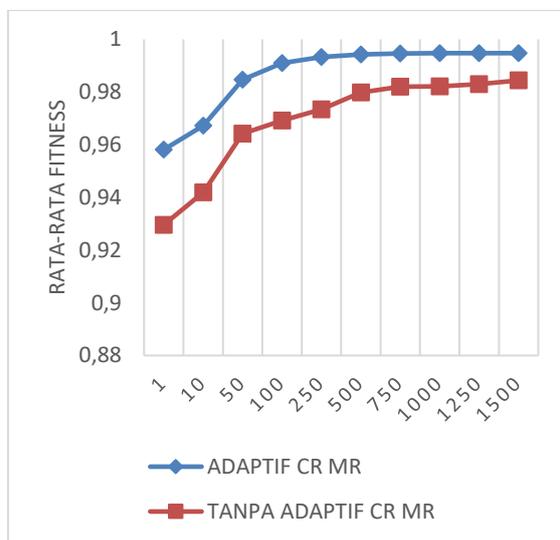
Gambar 3 Grafik Hasil Pengujian Populasi

Berdasarkan Gambar jumlah populasi dengan nilai *fitness* rata-rata tertinggi didapatkan pada jumlah populasi sebesar 100, dengan nilai *fitness* rata-rata sebesar 0,9976 untuk nilai rata-rata *fitness* terendah berada pada jumlah populasi sebesar 10 dengan nilai 0,9968. Dapat diambil kesimpulan semakin besar jumlah populasi awal, akan mendapatkan nilai *fitness*

yang semakin besar pula. Hal ini dapat terjadi dikarenakan semakin banyak solusi awal yang dibangkitkan, memungkinkan untuk melakukan pencarian solusi yang lebih luas dalam proses *crossover* dan *mutation*, yang mana diversitas dari solusi akan semakin banyak dihasilkan dan akan melakukan pencarian yang akan menghasilkan solusi yang optimum.

### 3.3 Pengujian Crossover Rate Mutation Rate Adaptif

Pengujian kombinasi dari *crossover rate* dan *mutation rate* juga berpengaruh penting pada eksplorasi dan eksploitasi dari solusi, yang mana nilai dari keduanya yang akan menentukan jumlah variasi solusi yang ada dari induk hingga anak atau *offspring*. Nilai *cr* awal diberi nilai sebesar 0,1, sedangkan untuk nilai *mr* awal diberi nilai sebesar 0,1. Jumlah populasi akan diberikan nilai sebesar 10 dan generasi akan diberikan nilai sebesar 1500. Nilai *simulated annealing* yang digunakan dari *alpha* hanya menggunakan nilai sebesar 0,8 dan untuk temperatur awal sebesar 30 dalam temperatur akhir 0,2. Hasil pengujian dari adaptif *cr mr* bisa dilihat pada Gambar 4.



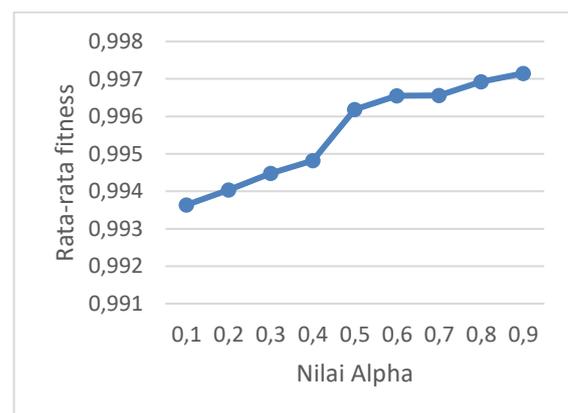
Gambar 4 Grafik Hasil Pengujian Adaptif CR MR

Berdasarkan Gambar 6.3, hasil rata-rata menggunakan adaptif *cr mr* lebih baik dibandingkan tidak menggunakan adaptif *cr mr*. dari perhitungan rata-rata selisih peningkatan nilai *fitness* antara penggunaan adaptif *cr mr* terhadap tanpa *cr mr* sebesar 0,0177. Hal ini disimpulkan bahwa dengan nilai *cr* dan *mr* yang adaptif, proses pencarian solusi bisa lebih optimum. Hal ini terjadi dikarenakan keseimbangan antara eksplorasi dan eksploitasi

daerah pencarian merupakan hal penting dalam algoritme heuristik (Mahmudy & Rahman, 2011). Pada iterasi awal nilai *cr* akan diberi nilai kecil dan nilai *mr* akan diberi nilai besar, hal ini dilakukan dengan tujuan proses pencarian terfokus kepada eksplorasi. Ketika iterasi mendekati maksimum, nilai *cr* akan cenderung membesar dan nilai *mr* akan cenderung mengecil guna proses terfokus pada proses eksploitasi, sehingga algoritme genetika dapat keluar dari lokal optimum dan kondisi konvergensi dini (Seissarina, et al., 2018).

### 3.4 Pengujian Nilai Alpha

Pada pengujian kali ini, nilai *alpha* akan digunakan pada proses *simulated annealing* jika dan hanya jika nilai selisih antara solusi baru terhadap solusi sebelumnya  $> 0$  dan nilai selisih itu akan dilakukan perhitungan menggunakan *probabilitas boltzmann* dan jika nilai *probabilitas boltzmann* ternyata kurang dari nilai random yang muncul baru akan dilanjutkan ke iterasi selanjutnya dengan mengubah nilai temperature awal menggunakan nilai *alpha*. Parameter yang digunakan untuk pengujian ini yaitu sebesar 500 untuk banyak generasi, jumlah populasi sebesar 10 dan kombinasi *crossover rate* dan *mutation rate* sebesar 0,1 dan 0,1 dan rate berubah tiap iterasi atau adaptif. Hasil nilai pengujian nilai *alpha* yang telah dilakukan bisa dilihat pada Gambar 5.



Gambar 5 Grafik Hasil Pengujian Nilai Alpha

Berdasarkan Gambar 5, grafik dapat dikatakan cenderung naik, hal ini dikarenakan nilai *alpha* yang semakin besar memungkinkan proses di tahap pencarian solusi di *simulated annealing* lebih lama. Nilai *alpha* bisa dikatakan sebagai *learning rate* yang mana proses di *simulated annealing* menerima segala solusi baik itu *uphill move* atau *downhill move*. Dengan demikian dapat disimpulkan semakin besar dari

nilai  $\alpha$ , akan memberi waktu yang lama untuk *simulated annealing* melakukan pencarian solusi.

### 3.5 Analisis Hasil

Pada bagian subbab ini akan dilakukan analisis perbandingan solusi *Hybrid Genetic Algorithm Simulated Snnealing*, solusi *Genetic Algorithm* tanpa *Simulated Annealing* dan solusi aktual dengan teknik manual yang dihitung oleh pakar. Parameter yang digunakan untuk melakukan perbandingan yaitu, jumlah populasi sebesar 10, jumlah generasi sebanyak 500, kombinasi nilai  $cr$   $mr$  yaitu masing-masing 0,1 akan tetapi berubah tiap iterasi secara adaptif. Nilai  $\alpha$  sebesar 0,8 dan temperatur awal sebesar 10. Parameter-parameter ini akan digunakan untuk membandingkan solusi yang didapatkan dari sistem, dengan solusi manual atau solusi ahli. Hasil solusi yang didapatkan ketiga solusi dapat dilihat pada Tabel 9.

Tabel 9 Hasil Solusi Nutrisi Terhadap Kebutuhan Pangan Pasien

Kebutuhan Gizi	Solusi		
	Pakar	GASA	GA
Kalori	1524	1489	1490
Karbohidrat	212,53	242,92	240,14
Protein	77,08	56,33	64,12
Lemak	45,70	33,25	32,91

Lalu akan dilakukan perbandingan ketiga solusi tersebut terhadap kebutuhan aktual pasien. Hasil dapat dilihat pada Tabel 10.

Tabel 10 Hasil Selisih Nutrisi Terhadap Kebutuhan Pangan Pasien

Kebutuhan Gizi	Solusi		
	Pakar	GASA	GA
Kalori	32,25	2,75	1,75
Karbohidrat	29,87	0,51	2,25
Protein	21,13	0,39	8,17
Lemak	12,55	0,09	0,24

Berdasarkan Tabel 10, solusi yang didapatkan *Hybrid GASA* memiliki nilai selisih yang terkecil diantara dua solusi lainnya. Adapun persentase dari *Hybrid GASA* terhadap

kebutuhan aktual dari pasien untuk kalori sebesar 0,18%, karbohidrat sebesar 0,20%, protein sebesar 0,69% dan terakhir yaitu lemak sebesar 0,27%.

### 4. KESIMPULAN DAN SARAN

Algoritme genetika dapat diimplementasikan dalam optimasi komposisi makanan bagi penderita penyakit diabetes melitus tipe 2, yang mana dalam proses algoritme genetika representasi kromosom direpresentasikan dalam bentuk integer positif, yang mana merupakan *ID\_MAKANAN* dari *database*. Untuk proses reproduksi yang menggunakan metode *extended immediate crossover* dan *random mutation* juga bisa dikatakan berhasil berjalan serta penerapan parameter kontrol yaitu *crossover rate* dan *mutation rate* secara adaptif. Dan untuk proses seleksi yang menggunakan metode *elitism* juga bisa dikatakan berhasil.

Metode *simulated annealing* berhasil ditambahkan dalam proses pencarian solusi, yang mana hasil dari pengujian membuktikan bahwa, solusi yang ditampilkan oleh *Hybrid algoritme genetika* dan *simulated annealing* berhasil mengungguli solusi yang hanya menggunakan algoritme genetika. Solusi yang dihasilkan menggunakan *simulated annealing* sebesar 0,998, sedangkan yang tidak menggunakan *simulated annealing* hanya sebesar 0,990. Persentase selisih dari solusi yang didapatkan menggunakan *Hybrid GASA* juga terbilang kecil, yaitu 0,18% untuk kalori lalu 0,20% pada karbohidrat, 0,69% untuk protein dan terakhir yaitu lemak sebesar 0,27%.

Perhitungan nilai *fitness* alangkah baiknya diteliti lebih lanjut untuk mendapatkan hasil yang maksimal. Lebih diutamakan untuk menentukan prioritas perhitungan selisih dari kebutuhan gizi seperti karbohidrat, protein atau lemak yang dapat diberi perlakuan berbeda jika nilai selisih bersifat negatif atau positif dalam perhitungan *fitness*.

Dalam penentuan porsi makanan sebaiknya dilakukan variasi porsi makanan. Hal ini dapat dilakukan dengan menggunakan representasi kromosom 2 segmen. Segmen pertama adalah indeks makanan, dan segmen kedua yaitu porsi atau berat dari makanan, yang mana porsi makanan yang didapat bisa berubah sesuai usia dan berat badan dari pasien.

### 5. DAFTAR PUSTAKA

- Anggarsari, F., Mahmudy, W. F. & Dewi, C., 2017. Optimasi Kebutuhan Gizi untuk Balita Menggunakan Hybrid Algoritma Genetika dan Simulated Annealing. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Volume 1, hal. 1668-1677.
- Husain, W., Wei, L. J., Cheng, S. L. & Zakaria, N., 2011. Application of data mining techniques in a personalized diet recommendation system for cancer patients. *Colloquium on Humanities*, hal. 239-244.
- Kementerian Kesehatan RI, 2018. Hari Diabetes Sedunia.
- Mahmudy, W. F., 2013. *Algoritma Evolusi*. Malang: Program Teknologi Informasi dan Ilmu Komputer (PTIIK).
- Mahmudy, W. F. & Rahman, M. A., 2011. Optimisasi fungsi multi-obyektif berkendala menggunakan algoritma genetika. *Jurnal Ilmiah Cursor*, Januari. Volume 6.
- Perkumpulan Endokrinologi Indonesia, 2011. *Konsensus Pengendalian dan Pencegahan Diabetes Mellitus Tipe 2 di Indonesia*, Jakarta: PB. PERKENI.
- Phanich, M., Pholkul, P. & Phimoltares, S., 2010. Food recommendation system using clustering. *Information Science and Applications (ICISA)*, hal. 1–8.
- Rianawati, A. & Mahmudy, W. F., 2015. Implementasi Algoritma Genetika untuk Optimasi Komposisi Makanan bagi Penderita Diabetes Mellitus. Volume 14.
- Seissarina, M. L., Cholissodin, I. & Nurwasito, H., 2018. Invigilator Examination Scheduling using Partial Random Injection and Adaptive Time Variant Genetic Algorithm. *Journal of Information Technology and Computer Science*, Volume 3, hal. 113-119.
- Sivanadam, S. N. & Deepa, S. N., 2007. *Introduction to Genetic Algorithms*. s.l.:Springer.
- Sofianti, T. D., 2004. Penjadwalan Multipurpose Batch Chemical Plant Dengan Metode Optimisasi Gabungan : Algoritma Genetika - Simulasi Annealing. *Proceedings, Komputer dan Sistem Intelijen (KOMMIT2004)*.
- Sutanto, 2010. *CEKAL (Cegah & Tangkal) Penyakit Modern*. Yogyakarta: Andi.
- Suyanto, 2007. *Artificial Intelligence : Searching, Reasoning, Planning and Learning*. Bandung: Informatika.
- Suyono, 2005. "Patofisiologi Diabetes Melitus" dalam *Penatalaksanaan Diabetes Melitus Terpadu*. Jakarta: Pusat Diabetes dan Lipid RSUP Nasional Dr. Cipto Mangunkusumo, Fakultas Kedokteran Universitas Indonesia.