

## ANALISIS TINGKAT OPTIMASI ALGORITMA GENETIKA DALAM HUKUM KETETAPAN HARDY-WEINBERG PADA BIN PACKING PROBLEM

Terry Noviar Panggabean  
STMIK ITMI

Jl. Timah Putih / Komp.Asia Mega Mas Blok G No.15-18  
Tnp.grup@gmail.com

**Abstrak**—Karna representasi abstrak dari beberapa sistem pengambilan keputusan yang nyata dalam kehidupan sehari hari membuat masalah optimasi kombinatorial umumnya sangat sulit untuk dipecahkan. Bin packing problem ialah solusi terbaik dalam mengatasi masalah optimasi kombinatorial, yang digunakan untuk mencari sebuah objek secara optimal dari sekelompok himpunan objek yang berhingga. Serangkaian pendekatan hybrid telah dikembangkan dalam hal ini untuk memecahkan masalah Bin Packing. Metaheuristik adalah salah satu pendekatan tingkat tinggi dalam memandu dalam memodifikasi beberapa metode heuristik lainnya untuk mencari tingkat optimasi yang lebih baik. Genetic Algorithm atau Algoritma Genetika juga merupakan metode metaheuristik yang digunakan untuk menyelesaikan berbagai masalah dalam hal peningkatan optimasi. Dalam algoritma genetika terdapat bermacam-macam varian. Dalam penelitian dipaparkan mengenai taksonomi dari algoritma genetika paralel (Parallel Genetic Algorithm) yang memiliki kemampuan yang lebih baik dari algoritma genetika konvensional dalam hal kinerja dan skalabilitasnya. Tetapi algoritma genetika paralel ini hanya cocok untuk permasalahan jaringan komputer heterogen dan sistem terdistribusi. Berdasarkan penelitian yang sudah pernah dilakukan sebelumnya dan dari uraian diatas maka penulis tertarik untuk melakukan penelitian bagaimana menerapkan hukum ketetapan Hardy-Weinberg dari bidang biologi kedalam algoritma genetika melakukan analisis tingkat optimasi terhadap Bin Packing Problem..

**Keywords**— Genetic Algoritihm, Hardy-Weinberg, Bin Packing Problem.

### I. PENDAHULUAN

Dokumen Bin packing problem ialah solusi terbaik dalam mengatasi masalah optimasi kombinatorial, yang digunakan untuk mencari sebuah objek secara optimal dari sekelompok himpunan objek yang berhingga. Serangkaian pendekatan hybrid telah dikembangkan dalam hal ini untuk memecahkan masalah Bin Packing. Metaheuristik adalah salah satu pendekatan tingkat tinggi dalam memandu dalam memodifikasi beberapa metode heuristik lainnya untuk mencari tingkat optimasi yang lebih baik. (Hopper, 2000)

Genetic Algorithm atau Algoritma Genetika juga merupakan metode metaheuristik yang digunakan untuk menyelesaikan berbagai masalah dalam hal peningkatan optimasi. Genetic Algorithm diciptakan pada tahun 1975 oleh John Holland yang mengemukakan komputasi berbasis evolusi dalam bukunya yang berjudul “Adaption in Natural and Artificial Intelligence”. Tujuannya adalah untuk merancang komputer untuk dapat meng-aplikasi-kan apa saja yang terdapat dari beberapa makhluk hidup. Holland mengemukakan sebuah algoritma yang memfokuskan pada manipulasi string dalam bentuk binary bit yang diambil dari konsep abstrak dari evolusi alam. Tahapan Algoritma Genetika yang dikemukakan dapat direpresentasikan sebagai tahapan yang berurutan sebagai bentuk populasi dari

kromosom buatan menjadi sebuah populasi baru. (Negnevitsky, 2005)

Menurut Gen & Cheng (1997), algoritma genetika memiliki beberapa kelebihan yaitu Algoritma ini hanya melakukan sedikit perhitungan matematis yang berhubungan dengan masalah apa yang ingin diselesaikan. Kemudian operator-operator evolusi yang digunakan membuat algoritma ini sangat efektif dalam hal pencarian global. Dan terakhir memiliki fleksibilitas yang tinggi untuk dihibridkan dengan metode pencarian lainnya supaya lebih efektif. Untuk itu penulis memilih algoritma ini karna algoritma ini mempunyai fleksibilitas yang tinggi dalam memadukan metode heuristik lainnya. (Sivanandam & Deepa, 2008)

Hukum Hardy-Weinberg atau yang sering disebut dengan Hukum Ketetapan Hardy-Weinberg menyatakan bahwa frekuensi alel dan frekuensi genotip dalam suatu populasi akan tetap konstan, yaitu berada dalam kesetimbangan dari satu generasi ke generasi berikutnya kecuali apabila terdapat pengaruh-pengaruh tertentu yang mengganggu kesetimbangan tersebut. Pengaruh-pengaruh yang dapat mengganggu kesetimbangan antara lain perkawinan tak acak, mutasi, seleksi, ukuran populasi terbatas, dan aliran gen. (Vogel & Motulsky, 1997)

Bin Packing Problem adalah permasalahan pengambilan putusan yang bersifat NP-Complete,

dimana telah terbukti dan diuji dengan Algoritma First Fit. Bagaimana nilai optimasi yang dicapai pada Bin Packing Problem jika diuji dengan algoritma genetika yang menerapkan hukum ketetapan Hardy-Weinberg? Seberapa besar tingkat keberhasilan yang dicapai dari algoritma genetika yang menerapkan hukum ketetapan Hardy-Weinberg dalam Bin Packing Problem?

Melalui penelitian ini diharapkan diperoleh kesimpulan mengenai parameter-parameter yang dihilangkan yaitu seleksi dan mutasi pada algoritma genetika dengan menerapkan hukum ketetapan Hardy-Weinberg pada Bin Packing Problem.

Berdasarkan penelitian yang sudah pernah dilakukan sebelumnya dan dari uraian diatas maka penulis tertarik untuk melakukan penelitian bagaimana menerapkan hukum ketetapan Hardy-Weinberg dari bidang biologi kedalam algoritma genetika teknik optimasi dan melakukan analisis terhadap Bin Packing Problem.

## II. TINJAUAN PUSTAKA

### A. Bin Packing Problem

Bin Packing Problem merupakan Terminologi atau tingkat lanjut daripada Knapsack Problem. Dimana n item dan n knapsack (atau bin), dengan :

$$W_j = \text{besar daripada item } j$$

$$c = \text{kapasitas daripada bin (wadah) tersebut}$$

Menetapkan masing-masing item kedalam suatu bin (wadah) sehingga besar total semua item tidak melebihi setiap bin (wadah) dan memberikan jumlah bin (wadah) yang minimum. Jika dituliskan dalam persamaan sebagai berikut :

$$\text{minimize } z = \sum_{i=1}^n y_i$$

$$\text{Subjek dari } \sum_{i=1}^n w_j x_{ij} \leq c y_i, \quad i \in N = \{1, \dots, n\}$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j \in N,$$

$$y_i = 0 \text{ or } 1, \quad i \in N,$$

$$x_{ij} = 0 \text{ or } 1, \quad i \in N,$$

Dimana,

$$y_i = \begin{cases} 1 & \text{jika bin } i \text{ sudah terpakai;} \\ 0 & \text{kebalikannya,} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{jika item } j \text{ disesuaikan dengan Bin (wadah) } i; \\ 0 & \text{kebalikannya.} \end{cases}$$

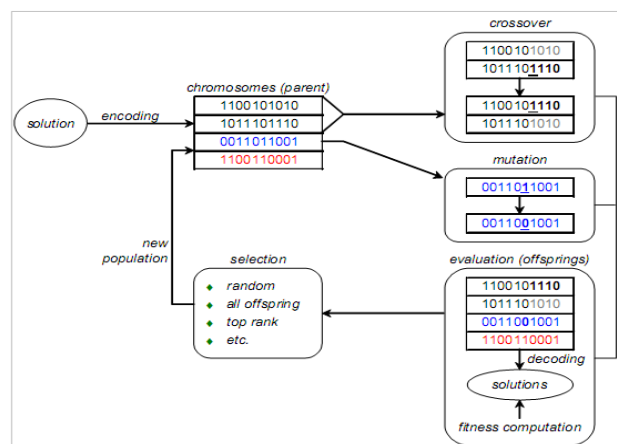
Kesimpulannya adalah pada Bin Packing Problem memiliki 2 jenis masalah yaitu bagaimana mencari Bin Packing yang mempunyai objek yang berukuran tetap yang akan dimasukkan dan ukuran yang tetap daripada Bin Packing tersebut. (Martello, S., & Toth, P. 1990).

### B. Algoritma Genetika

Tahun 1975, John Holland, salah satu pendiri Computing Evolution, memperkenalkan konsep algoritma genetika. Tujuannya adalah untuk merancang komputer untuk dapat meng-aplikasi-kan apa saja yang terdapat dari beberapa makhluk hidup. Sebagai seorang ilmuwan komputer, Holland prihatin dengan algoritma yang memanipulasi string pada digit biner. Dia melihat algoritma ini sebagai bentuk abstrak evolusi alami. GA dapat diwakili oleh urutan langkah-langkah prosedur untuk bergerak dari satu kromosom buatan untuk membentuk populasi baru. Menggunakan seleksi alami dan teknik yang diambil dari genetika yang dikenal sebagai crossover dan mutasi. Setiap kromosom terdiri dari sejumlah gen, dan setiap gen diwakili oleh bilangan 0 atau 1 (Negnevitsky, 2005).

Algoritma genetika memakai mekanisme seleksi alam dan ilmu genetika sehingga istilah-istilah pada Algoritma Genetika akan sejalan dengan istilah-istilah pada seleksi alam dan ilmu genetika pada umumnya. Sebuah solusi yang dikembangkan dalam algoritma genetika disebut sebagai kromosom, sedangkan kumpulan kromosom-kromosom tersebut disebut sebagai populasi.

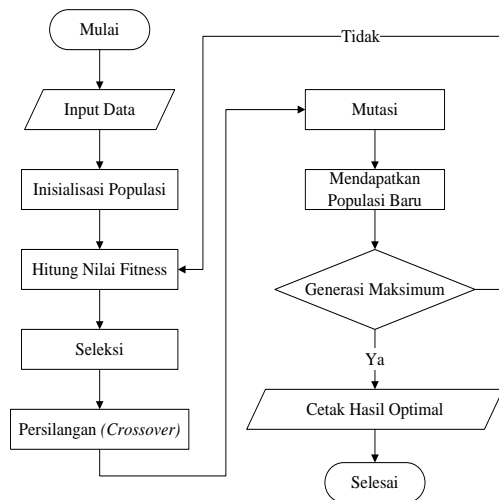
Secara umum tahapan proses dari algoritma genetika diperlihatkan pada Gambar 1.1. Seperti terlihat pada gambar kromosom merupakan representasi dari solusi. Operator genetika yang terdiri dari crossover dan mutasi dapat dilakukan keduanya atau hanya salah satu saja yang selanjutnya operator evolusi dilakukan melalui proses seleksi kromosom dari parent (generasi induk) dan dari offspring (generasi turunan) untuk membentuk generasi baru (new population) yang diharapkan akan lebih baik dalam memperkirakan solusi optimum, proses iterasi kemudian berlanjut sesuai dengan jumlah generasi yang ditetapkan.



Gbr. 1 Ilustrasi tahapan proses dari algoritma genetika (Gen & Cheng, 1997)

Algoritma genetika memberikan pilihan untuk menentukan nilai parameter dengan menduplikasi cara reproduksi genetika, pembentukan kromosom baru, proses migrasi gen, serta seleksi alami seperti yang terjadi pada organisme hidup. Secara umum Algoritma

Genetika dapat diilustrasikan melalui diagram pada Gambar di bawah ini.



Gbr. 2 Diagram Alir Struktur Umum Algoritma Genetika

Pada tahun 1908, seorang ahli matematika berkebangsaan Inggris Godfrey Harold Hardy (1877-1947) dan seorang dokter berkebangsaan Jerman Wilhelm Weinberg (1862-1937) secara terpisah menguraikan kondisi penting tentang keseimbangan genetik. Mereka menemukan dasar-dasar frekuensi alel dan genetik dalam suatu populasi terpisah menemukan suatu hubungan matematik dari frekuensi gen dalam populasi, yang kemudian dikenal dengan hukum Hardy-Weinberg (prinsip kesetimbangan). Pernyataan itu menegaskan bahwa frekuensi alel dan genotip suatu populasi (gene pool) selalu konstan dari generasi ke generasi dengan kondisi tertentu. Hukum ini digunakan sebagai parameter untuk mengetahui apakah dalam suatu populasi sedang berlangsung evolusi ataukah tidak. Hukum Hardy-Weinberg ini menjelaskan bahwa keseimbangan genotip AA, Aa, dan aa, serta perbandingan gen A dan gen a dari generasi ke generasi akan selalu sama dan tetap dipertahankan dalam suatu populasi bila memenuhi beberapa syarat (Vogel & Motulsky, 1997).

Hukum Hardy-Weinberg menyatakan bahwa di bawah suatu kondisi yang stabil, baik frekuensi gen maupun perbandingan genotip akan tetap (konstan) dari generasi ke generasi pada populasi yang berbiak secara seksual.

Syarat berlakunya asas Hardy-Weinberg:

1. Setiap gen mempunyai viabilitas dan fertilitas yang sama
2. Perkawinan terjadi secara acak
3. Tidak terjadi mutasi gen atau frekuensi terjadinya mutasi, sama besar.
4. Tidak terjadi migrasi
5. Jumlah individu dari suatu populasi selalu besar

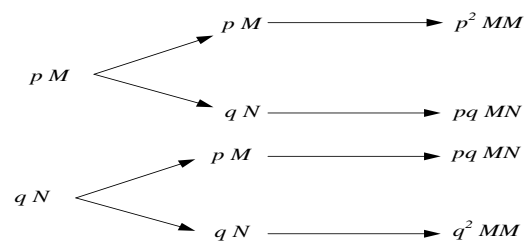
Jika lima syarat yang diajukan dalam kesetimbangan Hardy Weinberg tadi banyak dilanggar, jelas akan terjadi evolusi pada populasi tersebut, yang akan menyebabkan perubahan perbandingan alel dalam populasi tersebut. Definisi evolusi sekarang

dapat dikatakan sebagai perubahan dari generasi ke generasi dalam hal frekuensi alel atau genotipe populasi. Dalam perubahan dalam kumpulan gen ini (yang merupakan skala terkecil), spesifik dikenal sebagai mikroevolusi.

Hukum Hardy-Weinberg ini berfungsi sebagai parameter evolusi dalam suatu populasi. Bila frekuensi gen dalam suatu populasi selalu konstan dari generasi ke generasi, maka populasi tersebut tidak mengalami evolusi. Bila salah satu saja syarat tidak dipenuhi maka frekuensi gen berubah, artinya populasi tersebut telah dan sedang mengalami evolusi.

Hukum Hardy-Weinberg menyatakan populasi mendelian yang berukuran besar sangat memungkinkan terjadinya kawin acak (panmixia) di antara individu-individu anggotanya. Artinya, tiap individu memiliki peluang yang sama untuk bertemu dengan individu lain, baik dengan genotipe yang sama maupun berbeda dengannya. Dengan adanya sistem kawin acak ini, frekuensi alel akan senantiasa konstan dari generasi ke generasi.

Dengan kawin acak, hubungan antara frekuensi alel dan frekuensi genotipe sangat sederhana karena perkawinan acak individu setara dengan serikat acak gamet. Secara konseptual, kita mungkin membayangkan semua gamet suatu populasi sebagai hadiah dalam wadah besar. Untuk membentuk genotipe zigot, pasang gamet ditarik dari wadah secara acak. Untuk lebih spesifik, mempertimbangkan alel M dan N pada golongan darah MN, yang frekuensi alel adalah p dan q, masing-masing (ingat bahwa  $p + q = 1$ ). Frekuensi genotipe diharapkan dengan kawin acak dapat disimpulkan dari gambar 3 berikut (Daniel, et al. 1998):



Gbr. 3 Diagram Frekuensi genotype untuk kawin acak.

Genotipe yang dapat dibentuk dengan dua alel akan ditampilkan di sebelah kanan, dan dengan perkawinan acak, frekuensi masing-masing genotipe dihitung dengan mengalikan frekuensi alel dari gamet yang sesuai. Namun, MN genotipe dapat dibentuk dalam dua cara-alel M bisa datang dari ayah (bagian atas diagram) atau dari ibu (bagian bawah diagram). Dalam setiap kasus, frekuensi genotipe MN adalah pq; mengingat kedua kemungkinan, kita menemukan bahwa frekuensi MN adalah  $pq + pq = 2pq$ .

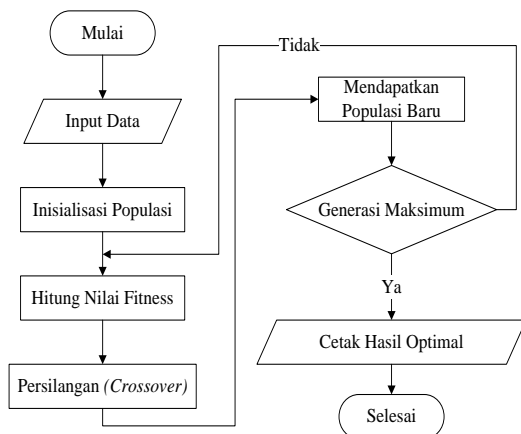
Di samping kawin acak, ada persyaratan lain yang harus dipenuhi bagi berlakunya hukum Hardy-Weinberg, yaitu tidak terjadi migrasi, mutasi, dan seleksi. Dengan perkataan lain, terjadinya peristiwa-peristiwa ini serta sistem kawin yang tidak acak akan

mengakibatkan perubahan frekuensi alel (Vogel & Motulsky, 1997).

### III. HASIL DAN PEMBAHASAN

#### A. Proses Bin Packing Problem Menggunakan Algoritma Genetika Menerapkan Hukum Ketetapan Hardy-Weinberg

Dalam pengambilan keputusan penerapan hukum ketetapan Hardy-Weinberg kedalam algoritma genetika prosesnya tidak jauh berbeda dengan algoritma genetika konvensional yang telah ada. Tetapi tahapan proses seleksi dan mutasi tidak dipakai sehingga dalam penerapan hukum Hardy-Weinberg hanya menggunakan tahapan-tahapan yang diilustrasikan pada gambar 4 berikut ini.



Gbr. 4 Proses Tahapan Penerapan Hukum Ketetapan Hardy-Weinberg dalam Algoritma Genetika (Perdana, 2015)

#### B. Pendefinisian Individu

Pada pendefinisian individu yang digunakan pada penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika sama seperti pendefinisian individu pada algoritma genetika umumnya yaitu pendefinisian individu dapat direpresentasikan dalam bentuk  $K1 = (x_1, x_2, x_3, \dots, x_n)$ . Dimana K merupakan kromosom dan x merupakan objek yang direpresentasikan dimasukkan ke dalam wadah atau bin.

#### C. Pembentukan populasi awal.

Untuk pembentukan populasi awal menggunakan proses yang sama seperti pada algoritma genetika umum yaitu dibentuk dari sejumlah kromosom. Kemudian dibentuk populasi awal dengan menggunakan Contoh A dimana kromosom yang akan dibentuk hanya 6 buah kromosom dari 10 buah gen yang diacak dikarenakan pada algoritma genetika yang menerapkan hukum ketetapan Hardy-Weinberg mempunyai syarat bahwa kromosom yang dibentuk haruslah berjumlah genap. Penulis mengambil sampel hanya 6 kromosom dikarenakan 6 kromosom yang dibentuk tidak terlalu lama dalam melakukan iterasi jika diasumsikan jumlah gen hanya 10 buah, dapat dilihat pada perhitungan sebagai berikut:

Nilai Objek pada Contoh A : {  $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}$  } ... { 45, 49, 49, 37, 46, 54, 55, 51, 46, 47 }

Data objeknya diambil sampel dari data jurnal Burke E. K., Hyde M. R. & Kendall G pada tahun 2010 yang berjudul *Providing a Memory Mechanism to Enhance the Evolutionary Design of Heuristics* pada *Proceedings of the IEEE World Congress on Computational Intelligence*, Barcelona, Spanyol halaman 3883-3890.

Kromosom[1] =  $x_4, x_2, x_1, x_3, x_5, x_6, x_7, x_9, x_8, x_{10}$  ... { 37 49 45 49 46 54 55 46 51 47 }

Kromosom[2] =  $x_2, x_3, x_1, x_5, x_4, x_7, x_6, x_8, x_9, x_{10}$  ... { 49 49 45 46 37 55 54 51 46 47 }

Kromosom[3] =  $x_{10}, x_2, x_4, x_5, x_3, x_6, x_7, x_1, x_9, x_8$  ... { 47 49 37 46 49 54 55 45 46 51 }

Kromosom[4] =  $x_4, x_3, x_{10}, x_2, x_5, x_9, x_7, x_1, x_6, x_8$  ... { 37 49 47 49 46 46 55 45 54 51 }

Kromosom[5] =  $x_{10}, x_8, x_4, x_9, x_7, x_2, x_1, x_3, x_5, x_6$  ... { 47 51 37 46 55 49 45 49 46 54 }

Kromosom[6] =  $x_{10}, x_4, x_8, x_9, x_6, x_1, x_2, x_5, x_7, x_3$  ... { 47 37 51 46 54 45 49 46 55 49 }

#### D. Proses Persilangan

Proses persilangan atau crossover dilakukan untuk melahirkan kromosom baru yang mewarisi sifat-sifat induknya. Kromosom baru yang terbentuk berasal dari duakromosom induk yang disilangkan. Pada penelitian ini digunakan teknik Partially Mapped Crossover sama seperti pada algoritma genetika umum. Dimana teknik ini akan melakukan persilangan pada dua titik yang ditentukan secara acak. Namun untuk penerapan hukum ketetapan Hardy-Weinberg, proses persilangan dilakukan pada seluruh kromosom yang terdapat pada individu secara acak. Misalkan Contoh A yang telah dibentuk populasinya di sub bab C Kemudian dilakukan pemilihan pasangan kromosom yang akan di crossover untuk seluruh kromosom seperti dibawah ini :

Kromosom[1] = { 37 ; 49 ; 45 ; 49 ; 46 ; 54 ; 55 ; 46 ; 51 | 47 }  
Kromosom[6] = { 47 ; 37 ; 51 ; 46 ; 54 ; 45 ; 49 ; 46 ; 55 | 49 }

Kromosom[2] = { 49 ; 49 ; 45 ; 46 ; 37 ; 55 | 54 ; 51 ; 46 ; 47 }  
Kromosom[4] = { 37 ; 49 ; 47 ; 49 ; 46 ; 46 | 55 ; 45 ; 54 ; 51 }

Kromosom[3] = { 47 ; 49 ; 37 ; 46 | 49 ; 54 ; 55 ; 45 ; 46 ; 51 }  
Kromosom[5] = { 47 ; 51 ; 37 ; 46 | 55 ; 49 ; 45 ; 49 ; 46 ; 54 }

Gbr. 5 Proses sebelum persilangan dilakukan

Kromosom[1] = { 37 ; 49 ; 45 ; 49 ; 46 ; 54 ; 55 ; 46 ; 51 | 49 }  
Kromosom[6] = { 47 ; 37 ; 51 ; 46 ; 54 ; 45 ; 49 ; 46 ; 55 | 47 }

Kromosom[2] = { 49 ; 49 ; 45 ; 46 ; 37 ; 55 | 55 ; 45 ; 54 ; 51 }  
Kromosom[4] = { 37 ; 49 ; 47 ; 49 ; 46 ; 46 | 54 ; 51 ; 46 ; 47 }

Kromosom[3] = { 47 ; 49 ; 37 ; 46 | 55 ; 49 ; 45 ; 49 ; 46 ; 54 }  
Kromosom[5] = { 47 ; 51 ; 37 ; 46 | 49 ; 54 ; 55 ; 45 ; 46 ; 51 }

Gbr. 6 Proses persilangan sudah dilakukan

Dikarenakan adanya duplikasi gen pada setiap kromosom hasil crossover maka diperlukan penggantian nilai gen pada posisi gen yang sama namun bukan hasil persilangan. Sebagai contoh pada gambar 7 dan 8 berikut.

|   |   |
|---|---|
| Kromosom[1] = {37 ; 49 ; 45 ; 49 ; 46 ; 54 ; 55 ; 46 ; 51   49} | Kromosom[6] = {47 ; 37 ; 51 ; 46 ; 54 ; 45 ; 49 ; 46 ; 55   47} |
| Kromosom[2] = {49 ; 49 ; 45 ; 46 ; 37 ; 55   55 ; 45 ; 54 ; 51} | Kromosom[4] = {37 ; 49 ; 47 ; 49 ; 46 ; 46   54 ; 51 ; 46 ; 47} |
| Kromosom[3] = {47 ; 49 ; 37 ; 46   55 ; 49 ; 45 ; 49 ; 46 ; 54} | Kromosom[5] = {47 ; 51 ; 37 ; 46   49 ; 54 ; 55 ; 45 ; 46 ; 51} |

Gbr. 7 Proses persilangan sudah dilakukan belum terjadi penggantian nilai gen

|   |   |
|---|---|
| Kromosom[1] = {37 ; 47 ; 45 ; 49 ; 46 ; 54 ; 55 ; 46 ; 51   49} | Kromosom[6] = {49 ; 37 ; 51 ; 46 ; 54 ; 45 ; 49 ; 46 ; 55   47} |
| Kromosom[2] = {49 ; 49 ; 47 ; 46 ; 37 ; 46   55 ; 45 ; 54 ; 51} | Kromosom[4] = {37 ; 49 ; 45 ; 49 ; 55 ; 46   54 ; 51 ; 46 ; 47} |
| Kromosom[3] = {47 ; 51 ; 37 ; 46   55 ; 49 ; 45 ; 49 ; 46 ; 54} | Kromosom[5] = {47 ; 49 ; 37 ; 46   49 ; 54 ; 55 ; 45 ; 46 ; 51} |

Gbr. 8 Proses persilangan sudah dilakukan dan sudah terjadi penggantian nilai gen

Sehingga didapat hasil persilangan sebagai berikut:

- Kromosom[1] = {37 ; 47 ; 45 ; 49 ; 46 ; 54 ; 55 ; 46 ; 51 ; 49}
- Kromosom[6] = {49 ; 37 ; 51 ; 46 ; 54 ; 45 ; 49 ; 46 ; 55 ; 47}
- Kromosom[2] = {49 ; 49 ; 47 ; 46 ; 37 ; 46 ; 55 ; 45 ; 54 ; 51}
- Kromosom[4] = {37 ; 49 ; 45 ; 49 ; 55 ; 46 ; 54 ; 51 ; 46 ; 47}
- Kromosom[3] = {47 ; 51 ; 37 ; 46 ; 55 ; 49 ; 45 ; 49 ; 46 ; 54}
- Kromosom[5] = {47 ; 49 ; 37 ; 46 ; 49 ; 54 ; 55 ; 45 ; 46 ; 51}

Maka didapat hasil fitness jika setiap objek ditambahkan dan harus lebih kecil sama dengan kapasitas wadah (bin) secara berurutan. Apabila penjumlahan nilai objek sudah lebih besar dari kapasitas bin maka akan dibuat bin yang baru. Dapat dilihat pada penyelesaian dibawah sebagai berikut :

- Kromosom [1] = 6 Bin, susunan bin = [37 47], [45 49], [46 54], [55], [46 51], [49]
- Kromosom [2] = 6 Bin, susunan bin = [49 49], [47 46], [37 46], [55 45], [54], [51]
- Kromosom [3] = 5 Bin, susunan bin = [47 51], [37 46], [55 45], [49 49], [46 54]
- Kromosom [4] = 6 Bin, susunan bin = [37 49], [45 49], [55], [46 54], [51 46], [47]

- Kromosom [5] = 6 Bin, susunan bin = [47 49], [37 46], [49 45], [54 46], [55], [51]
- Kromosom [6] = 6 Bin, susunan bin = [49 37], [51 46], [54 45], [49 46], [55], [47]

Dan didapat Fitness yang terbaik dari 6 kromosom ini hanya sebesar 5 Bin pada Kromosom[3] dalam iterasi ini. Pada iterasi ini hasil perhitungan fitness tiap kromosom sudah menunjukkan adanya optimasi terhadap banyak bin karna fitness terbaiknya adalah 5 wadah hanya tidak terlalu nampak hasil optimasi lebih tinggi yang diberikan dibandingkan dengan penggunaan algoritma decreasing first fit pada gambar 3.34 dan akan dicoba lagi dengan data yang lebih besar ataupun pengulangan sampai generasi ke-n (selanjutnya) pada algoritma genetika yang menerapkan hukum ketetapan Hardy-Weinberg.

Pada tahap berikutnya pengujian dilakukan dengan data yang lebih besar yang diambil dari data jurnal Burke E. K., Hyde M. R. & Kendall G pada tahun 2010 yang berjudul *Providing a Memory Mechanism to Enhance the Evolutionary Design of Heuristics* pada *Proceedings of the IEEE World Congress on Computational Intelligence*, Barcelona, Spanyol halaman 3883-3890.

Data ini akan diberikan nama Model\_TZ5 dengan algoritma genetika yang menerapkan Hukum Ketetapan Hardy-Weinberg dan diuji terhadap wadah sebesar 100, 120, 170, 240 dan 310 sebagai fitness serta dilakukan terhadap 1000, 2000 dan 3000 pengulangan generasi. Hasil pengujian tersebut dapat dilihat pada tabel 3.1, 3.2, 3.3, 3.4, dan 3.5 dibawah ini :

TABEL I  
HASIL PENGUJIAN MENGGUNAKAN ALGORITMA GENETIKA HARDY-WEINBERG DENGAN NILAI KAPASITAS BIN 100

| Populasi                        | Banyak Wadah yang terpakai (Fitness) |          |          | Nilai Fitness Rata-rata |
|---------------------------------|--------------------------------------|----------|----------|-------------------------|
|                                 | Gen 1000                             | Gen 2000 | Gen 3000 |                         |
| 20                              | 173                                  | 173      | 174      | 173                     |
| 30                              | 173                                  | 173      | 173      | 173                     |
| 40                              | 173                                  | 172      | 173      | 173                     |
| Total Fitness terbaik rata-rata |                                      |          |          | 173                     |

TABEL II  
HASIL PENGUJIAN MENGGUNAKAN ALGORITMA GENETIKA HARDY-WEINBERG DENGAN NILAI KAPASITAS BIN 120

| Populasi                        | Banyak Wadah yang terpakai (Fitness) |          |          | Nilai Fitness Rata-rata |
|---------------------------------|--------------------------------------|----------|----------|-------------------------|
|                                 | Gen 1000                             | Gen 2000 | Gen 3000 |                         |
| 20                              | 144                                  | 145      | 144      | 144                     |
| 30                              | 143                                  | 144      | 145      | 144                     |
| 40                              | 144                                  | 143      | 143      | 143                     |
| Total Fitness terbaik rata-rata |                                      |          |          | 173                     |

TABEL III  
HASIL PENGUJIAN MENGGUNAKAN ALGORITMA GENETIKA HARDY-WEINBERG DENGAN NILAI KAPASITAS BIN 170

| Populasi                               | Banyak Wadah yang terpakai ( <i>Fitness</i> ) |          |          | Nilai <i>Fitness</i> Rata-rata |
|--|---|----------|----------|--------------------------------|
|  | Gen 1000                                      | Gen 2000 | Gen 3000 |                                |
| 20                                     | 102   | 102      | 101      | 102                            |
| 30                                     | 102   | 101      | 100      | 101                            |
| 40                                     | 103   | 102      | 101      | 102                            |
| Total <i>Fitness</i> terbaik rata-rata |   |          |          | 102                            |

TABEL IV  
HASIL PENGUJIAN MENGGUNAKAN ALGORITMA GENETIKA HARDY-WEINBERG DENGAN NILAI KAPASITAS BIN 240

| Populasi                               | Banyak Wadah yang terpakai ( <i>Fitness</i> ) |          |          | Nilai <i>Fitness</i> Rata-rata |
|--|---|----------|----------|--------------------------------|
|  | Gen 1000                                      | Gen 2000 | Gen 3000 |                                |
| 20                                     | 71  | 71       | 72       | 71                             |
| 30                                     | 72  | 71       | 71       | 71                             |
| 40                                     | 71  | 71       | 71       | 71                             |
| Total <i>Fitness</i> terbaik rata-rata |   |          |          | 71                             |

TABEL V  
HASIL PENGUJIAN MENGGUNAKAN ALGORITMA GENETIKA HARDY-WEINBERG DENGAN NILAI KAPASITAS BIN 2310

| Populasi                               | Banyak Wadah yang terpakai ( <i>Fitness</i> ) |          |          | Nilai <i>Fitness</i> Rata-rata |
|--|---|----------|----------|--------------------------------|
|  | Gen 1000                                      | Gen 2000 | Gen 3000 |                                |
| 20                                     | 55  | 55       | 55       | 55                             |
| 30                                     | 56  | 55       | 55       | 55                             |
| 40                                     | 55  | 55       | 55       | 55                             |
| Total <i>Fitness</i> terbaik rata-rata |   |          |          | 71                             |

#### IV. PENUTUP

Dari pengamatan yang dilakukan oleh penulis dari hasil pengujian terhadap Model\_TZ5 pada tahap percobaan yang dilakukan pada penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika dapat disimpulkan bahwa penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika lebih baik dalam menemukan solusi atau *fitness* terbaik namun dalam segi kecepatan hukum ketetapan Hardy-Weinberg dalam algoritma genetika lebih lambat dalam menemukan *fitness* terbaiknya dibandingkan dengan algoritma lainnya pada model data di atas.

Berdasarkan hasil percobaan dan analisis yang dilakukan terhadap Model\_TZ5 dimana masing-masing data diuji sebanyak tiga kali dengan berbagai banyaknya lapisan generasi, setiap tahapnya diperoleh hasil bahwa penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika memperoleh

*fitness* yang semakin baik. Namun untuk kecepatan yang dihasilkan untuk memperoleh banyak wadah, algoritma genetika yang menerapkan hukum ketetapan Hardy-Weinberg sangat lama. Dari segi konsistensi hasil dari pengamatan yang penulis lakukan yang dimiliki oleh penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika menghasilkan solusi yang cukup. Namun jika dilihat dari tiap percobaan yang dilakukan solusi yang dihasilkan pasti tidak konsisten dari setiap ujicoba dikarenakan pada algoritma genetika yang menerapkan hukum ketetapan Hardy-Weinberg masih menggunakan bilangan acak pada setiap tahapannya.

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut:

1. Penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika lebih mencapai nilai optimal dalam menemukan best *fitness* (wadah paling sedikit) ataupun pada bin packing problem dalam menemukan banyaknya wadah. Namun penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika sangat lambat dalam menemukan solusi. Dari segi konsistensi hasil yang diperoleh penerapan hukum ketetapan Hardy-Weinberg dalam algoritma genetika tetap konsisten dalam memberikan solusi yang lebih optimal pada setiap percobaan.
2. Semakin besar jumlah kromosom dan jumlah generasi yang dimasukkan maka hasil yang diperoleh lebih optimal. Ini sesuai dengan salah satu syarat berlakunya dari hukum ketetapan Hardy-Weinberg yaitu jumlah individu atau kromosom dari suatu populasi selalu besar. (Perdana, 2015)
3. Dengan menghilangkan proses seleksi dan proses mutasi pada algoritma genetika seperti pada syarat berlakunya hukum ketetapan Hardy-Weinberg yang menyatakan bahwa suatu kondisi yang stabil, baik frekuensi gen maupun perbandingan genotip akan tetap (konstan) dari generasi ke generasi pada populasi yang berbiak secara seksual, juga dapat diterapkan pada permasalahan komputasi seperti bin packing problem dan dapat memperoleh hasil yang optimal.

Adapun saran yang diberikan pada penelitian ini adalah sebagai berikut:

1. Penelitian ini dapat dikembangkan lagi dengan menambahkan jumlah generasi yang lebih besar dan jumlah kromosom yang lebih besar lagi sehingga dapat diperoleh perbandingan hasil yang lebih baik.
2. Pada penelitian selanjutnya dapat ditambahkan pengujian terhadap teknik-teknik dari biologi evolusi lainnya untuk memberi perbandingan terhadap algoritma genetika umum.
3. Penambahan data uji yang lebih bervariasi sehingga didapat hasil yang lebih bervariasi.

UCAPAN TERIMA KASIH

Ucapan terima kasih penulis sampaikan kepada saudara Aditya Perdana, S.T., M.Kom., yang membantu saya dalam menyelesaikan penelitian ini. Karna penelitiannya yang berjudul *Analisis Performansi Pada Penerapan Hukum Ketetapan Hard-Weinberg Dalam Algoritma Genetika* membuat penulis tertarik melakukan analisa dan berbagai percobaan pada permasalahan *Bin Packing Problem*.

REFERENSI

- [1] [1] Burke E. K., Hyde M. R. & Kendall G. Providing a Memory Mechanism to Enhance the Evolutionary Design of Heuristics. Proceedings of the IEEE World Congress on Computational Intelligence. Barcelona, Spain. pp 3883-3890. 2010.
- [2] Daniel, K., Hirshleifer, D., Subrahmanyam, A., A theory of overconfidence, self-attribution, and security market under- and over-reactions. Journal of Finance 53, in press. Davis, L. 1991. Handbook of Genetic Algorithms. New York: Van Nostrand Reinhold. 1998.
- [3] Hopper, Eva., Two-dimensional Packing utilising Evolutionary Algorithms and other Meta-Heuristic Methods. Thesis. University Of Wales, 1-73., 2000.
- [4] Martello, S., & Toth, P. Knapsack Problem : Algorithms and Computer Implementations. John Wiley & Sons Ltd, Chapter 8, 221-245. 1990.
- [5] Negnevitsky, M., Artificial Intelligence : A Guide to Intelligent Systems, Second Edition. Addison-Wesley: Harlow. 2005.
- [6] Perdana, A. Analisis Performansi Pada Penerapan Hukum Ketetapan Hard-Weinberg Dalam Algoritma Genetika, Thesis. Magister Teknik Informatika Fasilkom-TI USU, 5-28. 2015.
- [7] Sivanandam, S. N., & Deepa, S. N. Introduction to Genetic Algorithms. Springer: Berlin. 2008.
- [8] Vogel, F., & Motulsky, A. G. Human Genetics : Problems and Approaches. Springer: Berlin. 1997.