PENCOCOKAN KATA SECARA ACAK DENGAN METODE ALGORITMA GENETIKA MENGGUNAKAN PROGRAM PASCAL

MULIA AFRIANI KARTIKA

*Program Studi Matematika,
Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Andalas,
Kampus UNAND Limau Manis Padang, Indonesia,
muliaafrianiartika@gmail.com*

Abstract. The Genetic Algorithm was introduced independently for the first time by Holland and De Jong in 1975. This algorithm is the stochastic optimization techniques based on the principle from evolution theory. Since then, numerous applications of genetic algorithm have been developed in many areas, such as scheduling and sequencing, reliability design, road network, and many others. This paper is addressed to discuss some steps performed in genetic algorithm processes. We also give an example of the algorithm in word matching randomly using Pascal programming, which specifically developed for academic purposes. Some factors that influence the accuracy of the resulting word matching at random is also explained in detail by running Pascal program repeatedly.

Kata Kunci: Genetic algorithm, word matching, optimization technique.

1. Pendahuluan

Algoritma genetika pertama kali diperkenalkan oleh John Holland dari Universitas Michigan. Algoritma genetika merupakan teknik pencarian dan optimasi yang terinspirasi oleh prinsip dari genetika dan seleksi alam (teori evolusi Darwin) [6]. Algoritma genetika digunakan untuk mendapatkan solusi yang tepat untuk masalah optimasi satu variabel atau multi variabel. Algoritma pencarian kata (*string*) atau sering disebut juga pencocokan kata adalah algoritma untuk melakukan pencarian kemungkinan kemunculan kata. Pencocokan kata dilakukan dengan memberikan sebuah kata sebagai target, kemudian membangkitkan kata secara acak yang dilakukan berkali-kali hingga pada akhirnya ditemukan kata yang menjadi target pada suatu populasi. Baik tidaknya hasil dari pencocokan kata secara acak tergantung pada nilai acuan yang diberikan, apakah yang dicari nilai tertentu, nilai maksimal maupun nilai minimal. Pencocokan dilakukan hingga diperoleh nilai yang sama atau mendekati nilai target yang diberikan, sehingga akurasi pencocokan ini ditentukan oleh kesamaan dengan nilai target [1].

2. Algoritma Genetika

Algoritma genetika merupakan suatu algoritma pencarian yang berbasis pada mekanisme seleksi alam dan genetika. Pada algoritma genetika, teknik pencarian

dilakukan sekaligus atas sejumlah solusi yang mungkin dikenal dengan istilah populasi. Individu yang terdapat dalam satu populasi disebut dengan istilah kromosom. Kromosom ini merupakan suatu solusi yang masih berbentuk simbol. Populasi awal dibangun secara acak, sedangkan populasi berikutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan generasi.

Algoritma genetika terdiri dari dua operasi yaitu operasi genetika dan operasi evolusi. Operasi genetika terdiri dari operator *crossover* dan operator mutasi. Pada operasi evolusi terdapat operator seleksi. Populasi generasi yang baru dibentuk dengan cara menyeleksi nilai *fitness* dari kromosom induk (*parent*) dan nilai *fitness* dari kromosom anak (*offspring*), serta menolak kromosom-kromosom yang lainnya sehingga ukuran populasi (jumlah kromosom dalam suatu populasi) konstan [1].

Fungsi evaluasi adalah penghubung antara algoritma genetika dan masalah yang akan diselesaikan. Fungsi evaluasi merupakan dasar untuk melakukan proses seleksi. Pada proses seleksi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang dinamakan *fitness*.

3. Pencocokan Kata Secara Acak dengan Algoritma Genetika Menggunakan Program Pascal

Misalkan diberikan sebuah kata sebagai target. Individu yang menjadi target terdiri dari n gen yang merupakan banyak huruf dalam kata. Misalkan dibangkitkan populasi awal yaitu $P(0)$. Selanjutnya, dilakukan proses seleksi, *crossover*, dan mutasi terhadap individu-individu di populasi awal, sehingga dihasilkan generasi baru dari populasi yaitu anak dari generasi sebelumnya. Generasi baru yaitu $(t+1)$ merupakan individu-individu yang terdapat pada populasi baru $P(t+1)$.

Langkah-langkah algoritma genetika, sebagaimana dituliskan oleh Masatoshi pada [6] adalah sebagai berikut.

- (1) (Kondisi awal) Dibangkitkan n individu secara acak dari populasi $P(0)$. Pilih generasi $t = 0$.
- (2) (Evaluasi) Hitung nilai *fitness* dari masing-masing individu di dalam populasi $P(t)$.
- (3) (Seleksi) Gunakan operator seleksi untuk populasi $P(t)$.
- (4) (*Crossover*) Gunakan operator *crossover* untuk populasi setelah melakukan proses seleksi.
- (5) (Mutasi) Gunakan operator mutasi untuk populasi setelah melakukan proses *crossover* untuk menghasilkan populasi baru $P(t+1)$ dari generasi selanjutnya $t+1$.
- (6) (Evaluasi) Hitung kembali nilai *fitness* dari individu yang baru terbentuk.
- (7) (Kondisi akhir) Jika $t = T$, maka berhenti. Jika tidak, kembali ke langkah dua.

Prosedur algoritma genetika ditunjukkan oleh *flowchart* pada Gambar 3.1.

3.1. Representasi Individu

Diberikan sebuah kata sebagai target, misalkan kata yang menjadi target adalah "MULIA". Individu yang merupakan target terdiri dari n gen. Masing-masing gen

Gambar 3.1. Flowchart prosedur dasar algoritma genetika

merupakan huruf dalam alfabet dan dikonversi ke dalam besaran numerik. Huruf "A" dikonversi ke angka 1, huruf "B" dikonversi ke angka 2, dan seterusnya huruf "Z" dikonversi ke angka 26, nilai dari gen dinyatakan dengan bilangan bulat antara 1 sampai 26 sesuai dengan urutan huruf dalam alfabet.

Apabila individu yang menjadi target dengan bentuk numerik $[t_1, t_2, \dots, t_n]$, dan individu yang dipilih secara acak dalam bentuk numerik individu ke- k adalah $[g_1^k, g_2^k, \dots, g_n^k]$ maka diperoleh nilai perbedaan E , yaitu

$$E = |g_1^k - t_1| + |g_2^k - t_2| + \dots + |g_n^k - t_n| = \sum_{i=1}^n |g_i^k - t_i|.$$

3.2. Membangkitkan Populasi Awal

Dibangkitkan beberapa individu secara acak dalam satu populasi yang dibahas. Misalkan satu populasi terdiri dari k individu dan masing-masing individu terdapat n gen. Dalam kasus ini dibangkitkan dua belas individu dengan lima gen berupa bilangan bulat, yang mana nilai gen terletak antara 1 sampai 26. Dengan menggunakan bahasa program Pascal yang terdapat dilampiran, diperoleh data populasi awal pada Gambar 3.2.

3.3. Fungsi Fitness

Fungsi *fitness* memastikan bahwa evolusi menuju optimasi dengan menghitung nilai *fitness* untuk masing-masing individu dalam populasi. Nilai *fitness* dari masing-

No	Individu									
1.	K	H	L	W	W	<-->	11	8	12	23 23
2.	E	V	Q	L	Z	<-->	5	22	17	12 26
3.	C	N	L	J	J	<-->	3	14	12	10 10
4.	C	D	Y	R	E	<-->	3	4	25	18 5
5.	M	F	D	O	I	<-->	13	6	4	15 9
6.	K	B	L	G	C	<-->	11	2	12	7 3
7.	S	C	P	U	H	<-->	19	3	16	21 8
8.	N	W	O	I	D	<-->	14	23	15	9 4
9.	T	L	Y	N	H	<-->	20	12	25	14 8
10.	K	K	M	C	B	<-->	11	11	13	3 2
11.	Y	B	C	X	T	<-->	25	2	3	24 20
12.	S	Q	V	V	V	<-->	19	17	22	22 22

Gambar 3.2. Data populasi awal

masing individu yang diperoleh secara acak adalah

$$fitness = 26(n) - E = 26(n) - \sum_{i=1}^n |g_i^k - t_i|$$

di mana:

n : banyak gen dari target ,

g_i^k : gen ke i dari individu ke k ,

t_i : gen ke i dari target

Selanjutnya akan dihitung nilai *fitness* untuk masing-masing individu. Dengan menggunakan rumus yang terdapat di atas, diperoleh data populasi awal individu dengan nilai *fitness* seperti yang tercantum pada Gambar 3.3.

No	Individu										Nilai Fitness
1.	K	H	L	W	W	<-->	11	8	12	23 23	<--> 79
2.	E	V	Q	L	Z	<-->	5	22	17	12 26	<--> 88
3.	C	N	L	J	J	<-->	3	14	12	10 10	<--> 103
4.	C	D	Y	R	E	<-->	3	4	25	18 5	<--> 77
5.	M	F	D	O	I	<-->	13	6	4	15 9	<--> 93
6.	K	B	L	G	C	<-->	11	2	12	7 3	<--> 105
7.	S	C	P	U	H	<-->	19	3	16	21 8	<--> 83
8.	N	W	O	I	D	<-->	14	23	15	9 4	<--> 121
9.	T	L	Y	N	H	<-->	20	12	25	14 8	<--> 89
10.	K	K	M	C	B	<-->	11	11	13	3 2	<--> 110
11.	Y	B	C	X	T	<-->	25	2	3	24 20	<--> 56
12.	S	Q	V	V	V	<-->	19	17	22	22 22	<--> 76

Gambar 3.3. Data populasi awal dengan nilai fitness

3.4. Seleksi

Seleksi dilakukan dengan menggunakan *roulette wheel*, untuk memilih induk dilakukan dengan menghitung persentase *fitness* setiap individu. Selanjutnya akan dihitung persentase dari nilai *fitness* masing-masing individu. Persentase nilai *fitness* masing-masing individu tercantum pada Gambar 3.4.

No	Individu	Nilai Fitness	Persentase
1.	K H L W W <--> 11 8 12 23 23 <-->	79 <-->	7
2.	E V Q L Z <--> 5 22 17 12 26 <-->	88 <-->	8
3.	C N L J J <--> 3 14 12 10 10 <-->	103 <-->	9
4.	C D Y R E <--> 3 4 25 18 5 <-->	77 <-->	7
5.	M F D O I <--> 13 6 4 15 9 <-->	93 <-->	8
6.	K B L G C <--> 11 2 12 7 3 <-->	105 <-->	9
7.	S C P U H <--> 19 3 16 21 8 <-->	83 <-->	7
8.	N W O I D <--> 14 23 15 9 4 <-->	121 <-->	11
9.	T L Y N H <--> 20 12 25 14 8 <-->	89 <-->	8
10.	K K M C B <--> 11 11 13 3 2 <-->	110 <-->	10
11.	Y B C X T <--> 25 2 3 24 20 <-->	56 <-->	5
12.	S Q V V V <--> 19 17 22 22 22 <-->	76 <-->	7

Gambar 3.4. Persentase nilai fitness

Dari persentase nilai *fitness* dapat diketahui bahwa jika individu pertama mempunyai persentase nilai *fitness* 7, maka individu pertama mempunyai lebar 7. Akibatnya individu pertama menempati nilai 1, 2, ..., 7. Seterusnya individu kedua belas mempunyai persentase nilai *fitness* 7, maka individu kedua belas mempunyai lebar 7. Akibatnya individu kedua belas menempati nilai 90, 91, ..., 96, sehingga diperoleh data pada Gambar 3.5.

No	Individu	Persen	Range
1.	K H L W W <--> 11 8 12 23 23 <-->	7 <-->	1, 2, ..., 7
2.	E V Q L Z <--> 5 22 17 12 26 <-->	8 <-->	8, 9, ..., 15
3.	C N L J J <--> 3 14 12 10 10 <-->	9 <-->	16, 17, ..., 24
4.	C D Y R E <--> 3 4 25 18 5 <-->	7 <-->	25, 26, ..., 31
5.	M F D O I <--> 13 6 4 15 9 <-->	8 <-->	32, 33, ..., 39
6.	K B L G C <--> 11 2 12 7 3 <-->	9 <-->	40, 41, ..., 48
7.	S C P U H <--> 19 3 16 21 8 <-->	7 <-->	49, 50, ..., 55
8.	N W O I D <--> 14 23 15 9 4 <-->	11 <-->	51, 52, ..., 66
9.	T L Y N H <--> 20 12 25 14 8 <-->	8 <-->	67, 68, ..., 74
10.	K K M C B <--> 11 11 13 3 2 <-->	10 <-->	75, 76, ..., 84
11.	Y B C X T <--> 25 2 3 24 20 <-->	5 <-->	85, 86, ..., 89
12.	S Q V V V <--> 19 17 22 22 22 <-->	7 <-->	90, 91, ..., 96

Gambar 3.5. Persentase nilai fitness dan Range

Dibangkitkan bilangan acak dari 1 sampai 96 sebanyak individu awal yaitu 12. Dengan menggunakan bahasa program Pascal diperoleh data individu yang terpilih pada proses seleksi dalam Gambar 3.6.

No	Terpilih	Susunan Kromosom
1.	12	19 17 22 22 22 <--> S Q V V V
2.	12	19 17 22 22 22 <--> S Q V V V
3.	8	14 23 15 9 4 <--> N W O I D
4.	7	19 3 16 21 8 <--> S C P U H
5.	1	11 8 12 23 23 <--> K H L W W
6.	9	20 12 25 14 8 <--> T L Y N H
7.	4	3 4 25 18 5 <--> C D Y R E
8.	2	5 22 17 12 26 <--> E V Q L Z
9.	1	11 8 12 23 23 <--> K H L W W
10.	7	19 3 16 21 8 <--> S C P U H
11.	7	19 3 16 21 8 <--> S C P U H
12.	1	11 8 12 23 23 <--> K H L W W

Gambar 3.6. Susunan Kromosom

3.5. Crossover

Crossover dilakukan dengan cara menukarkan nilai gen dari dua individu yang dilakukan secara acak. Prosedur awal untuk memilih induk mana yang akan mengalami proses *crossover* adalah menentukan probabilitas *crossover*. Probabilitas *crossover* berada antara 0,6 sampai 0,9 [6]. Selanjutnya dibangkitkan bilangan acak 0 sampai dengan 1 sebanyak individu awal yaitu 12, kemudian dibandingkan bilangan acak yang terpilih dengan probabilitas *crossover*. Individu yang mengalami *crossover* terdapat pada data pada Gambar 3.7.

No	Susunan Kromosom Anak										
1.	19	17	22	22	22	<-->	S	Q	V	V	V
2.	19	17	22	22	22	<-->	S	Q	V	V	V
3.	19	3	16	21	4	<-->	S	C	P	U	D
4.	14	23	15	9	8	<-->	N	W	O	I	H
5.	11	8	12	23	23	<-->	K	H	L	W	W
6.	20	12	25	14	8	<-->	T	L	Y	N	H
7.	5	4	25	18	5	<-->	E	D	Y	R	E
8.	3	22	17	12	26	<-->	C	V	Q	L	Z
9.	11	8	12	23	23	<-->	K	H	L	W	W
10.	19	3	16	21	8	<-->	S	C	P	U	H
11.	19	3	16	21	8	<-->	S	C	P	U	H
12.	11	8	12	23	23	<-->	K	H	L	W	W

Gambar 3.7. Crossover

3.6. Mutasi

Mutasi dilakukan dengan cara menggeser nilai gen dari individu yang akan dimutasi. Penggeseran ini dilakukan dengan cara melakukan penambahan atau pengurangan nilai gen pada posisi tertentu pada gen yang akan dimutasi. Prosedur awal yang dilakukan untuk mengetahui gen mana yang dimutasi adalah menentukan probabilitas mutasi. Menurut Goldberg [6], probabilitas mutasi dengan parameter antara 0,001 sampai 0,01 memberikan hasil yang optimal. Selanjutnya dibangkitkan bilangan acak 0 sampai dengan 1 sebanyak individu awal yaitu 12, kemudian dibandingkan bilangan acak yang terpilih dengan probabilitas mutasi. Individu yang mengalami mutasi terdapat pada data pada Gambar 3.8.

3.7. Membentuk Individu Baru

Anak hasil (*crossover*) dan mutasi menjadi generasi baru untuk dilakukan proses regenerasi. Pada generasi berikutnya, individu terbaik dapat dipertahankan dengan proses elitisme. Dengan menggunakan bahasa program Pascal diperoleh generasi selanjutnya dari individu seperti pada Gambar 3.9.

Individu yang terdapat pada generasi ke-2 dijadikan sebagai individu awal untuk dilakukan seleksi, *crossover* dan mutasi. Setelah melakukan serangkaian proses algoritma genetika tersebut diperoleh data generasi ke-3 seperti pada Gambar 3.10.

No	Susunan Kromosom Anak										
1.	14	17	22	22	22	<-->	N	Q	V	V	V
2.	19	17	22	22	22	<-->	S	Q	V	V	V
3.	19	14	16	21	4	<-->	S	N	P	U	D
4.	1	18	4	9	8	<-->	A	R	D	I	H
5.	14	14	12	23	23	<-->	N	N	L	W	W
6.	20	12	25	14	8	<-->	T	L	Y	N	H
7.	5	4	25	18	5	<-->	E	D	Y	R	E
8.	3	25	1	12	26	<-->	C	Y	A	L	Z
9.	11	8	12	23	23	<-->	K	H	L	W	W
10.	19	3	16	21	8	<-->	S	C	P	U	H
11.	19	3	16	21	8	<-->	S	C	P	U	H
12.	11	8	12	23	23	<-->	K	H	L	W	W

Gambar 3.8. Mutasi pada Individu

GENERASI KE - 2													
No	Susunan Kromosom										Nilai Fitness		
1.	N	Q	V	V	V	<-->	14	17	22	22	22	<-->	81
2.	A	R	D	I	H	<-->	1	18	4	9	8	<-->	100
3.	S	C	P	U	D	<-->	19	14	16	21	4	<-->	98
4.	A	R	D	I	H	<-->	1	18	4	9	8	<-->	100
5.	N	N	L	W	W	<-->	14	14	12	23	23	<-->	86
6.	T	L	Y	N	H	<-->	20	12	25	14	8	<-->	89
7.	E	D	Y	R	E	<-->	5	4	25	18	5	<-->	79
8.	C	Y	A	L	Z	<-->	3	25	1	12	26	<-->	77
9.	K	H	L	W	W	<-->	11	8	12	23	23	<-->	79
10.	S	C	P	U	H	<-->	19	3	16	21	8	<-->	83
11.	S	C	P	U	H	<-->	19	3	16	21	8	<-->	83
12.	K	H	L	W	W	<-->	11	8	12	23	23	<-->	79

Gambar 3.9. Generasi ke-2

GENERASI KE - 3													
No	Susunan Kromosom										Nilai Fitness		
1.	M	Y	W	H	H	<-->	13	25	23	8	8	<-->	107
2.	J	J	R	S	B	<-->	10	10	18	19	2	<-->	99
3.	W	I	U	M	J	<-->	23	9	21	13	10	<-->	86
4.	T	M	M	T	V	<-->	20	13	13	20	22	<-->	82
5.	J	W	R	M	J	<-->	10	23	18	13	10	<-->	106
6.	S	V	N	R	N	<-->	19	22	14	18	14	<-->	99
7.	Y	Y	W	H	H	<-->	25	25	23	8	8	<-->	95
8.	M	Y	W	H	H	<-->	13	25	23	8	8	<-->	107
9.	M	H	O	A	S	<-->	13	8	15	1	19	<-->	88
10.	Y	Y	W	H	H	<-->	25	25	23	8	8	<-->	95
11.	J	Y	W	S	B	<-->	10	25	23	19	2	<-->	101
12.	L	H	V	J	J	<-->	12	8	22	10	10	<-->	96

Gambar 3.10. Generasi ke-3

Dengan melakukan serangkaian proses algoritma genetika seperti seleksi, *crossover*, dan mutasi, diperoleh data generasi ke-4 seperti pada Gambar 3.11. dan seterusnya, hingga kata yang menjadi target ditemukan pada generasi ke-490 individu ke-9, dapat dilihat pada Gambar 3.12.

GENERASI KE - 4													
No	Susunan Kromosom										Nilai Fitness		
1.	J	Q	W	H	H	<-->	10	17	23	8	8	<-->	104
2.	J	L	V	S	B	<-->	10	12	22	19	2	<-->	97
3.	W	I	U	M	J	<-->	23	9	21	13	10	<-->	86
4.	T	M	M	T	V	<-->	20	13	13	20	22	<-->	82
5.	U	N	B	M	J	<-->	21	14	2	13	10	<-->	92
6.	S	V	N	R	N	<-->	19	22	14	18	14	<-->	99
7.	Y	Y	W	H	H	<-->	25	25	23	8	8	<-->	95
8.	J	Q	W	H	H	<-->	10	17	23	8	8	<-->	104
9.	M	H	O	A	S	<-->	13	8	15	1	19	<-->	88
10.	Y	Y	W	H	H	<-->	25	25	23	8	8	<-->	95
11.	J	V	W	S	B	<-->	10	22	23	19	2	<-->	104
12.	X	W	R	J	J	<-->	24	23	18	10	10	<-->	101

Gambar 3.11. Generasi ke-4

GENERASI KE - 490													
No	Susunan Kromosom											Nilai Fitness	
1.	Z	D	R	S	H	<-->	26	4	18	19	8	<-->	77
2.	Q	S	E	P	B	<-->	17	19	5	16	2	<-->	109
3.	J	S	N	P	J	<-->	10	19	14	16	10	<-->	107
4.	E	K	N	Q	V	<-->	5	11	14	17	22	<-->	81
5.	J	Z	L	B	J	<-->	10	26	12	2	10	<-->	106
6.	W	U	L	A	B	<-->	23	21	12	1	2	<-->	111
7.	J	S	E	P	H	<-->	10	19	5	16	8	<-->	104
8.	W	U	L	A	T	<-->	23	21	12	1	20	<-->	93
9.	M	U	L	I	A	<-->	13	21	12	9	1	<-->	130
10.	N	F	Z	H	H	<-->	14	6	26	8	8	<-->	92
11.	W	U	L	A	B	<-->	23	21	12	1	2	<-->	111
12.	P	N	F	E	J	<-->	16	14	6	5	10	<-->	101

Gambar 3.12. Generasi ke-490

4. Ucapan Terima kasih

Penulis mengucapkan terima kasih kepada Ibu Lyra Yulianti, Bapak Budi Rudianto, dan Bapak Zulakmal yang telah memberikan masukan dan saran sehingga paper ini dapat diselesaikan dengan baik.

Daftar Pustaka

- [1] Basuki, Achmad. 2003. *Algoritma Genetika Suatu Alternatif Penyelesaian Permasalahan Searching, Optimasi dan Machine Learning*. Politeknik Elektronika Negeri Surabaya. Surabaya
- [2] Coley, A David. 1999. *An Introduction to Genetic Algorithms for Scientists and Engineers*. World Scientific Publishing Co. Singapore
- [3] Kosasih, Djunaedi dan Rinaldo. *Analisis Aplikasi Algoritma Genetika untuk Pencarian Nilai Fungsi Maksimum*. Institut Teknologi Bandung. Bandung
- [4] Kusumadewi, Sri. 2005. *Pengantar Kecerdasan Buatan*. Jakarta
- [5] Purnomo, Andi. 1980. *Teknik Pemrograman Pascal*. Erlangga. Jakarta
- [6] Sakawa, Masatoshi. 2002. *Genetic Algorithms and Fuzzy Multiobjective Optimization*. Kluwer Academic Publishers. USA. Hal 11 - 30

- [7] Sanjoyo. 2006. *Aplikasi Algoritma Genetika*. Bandung