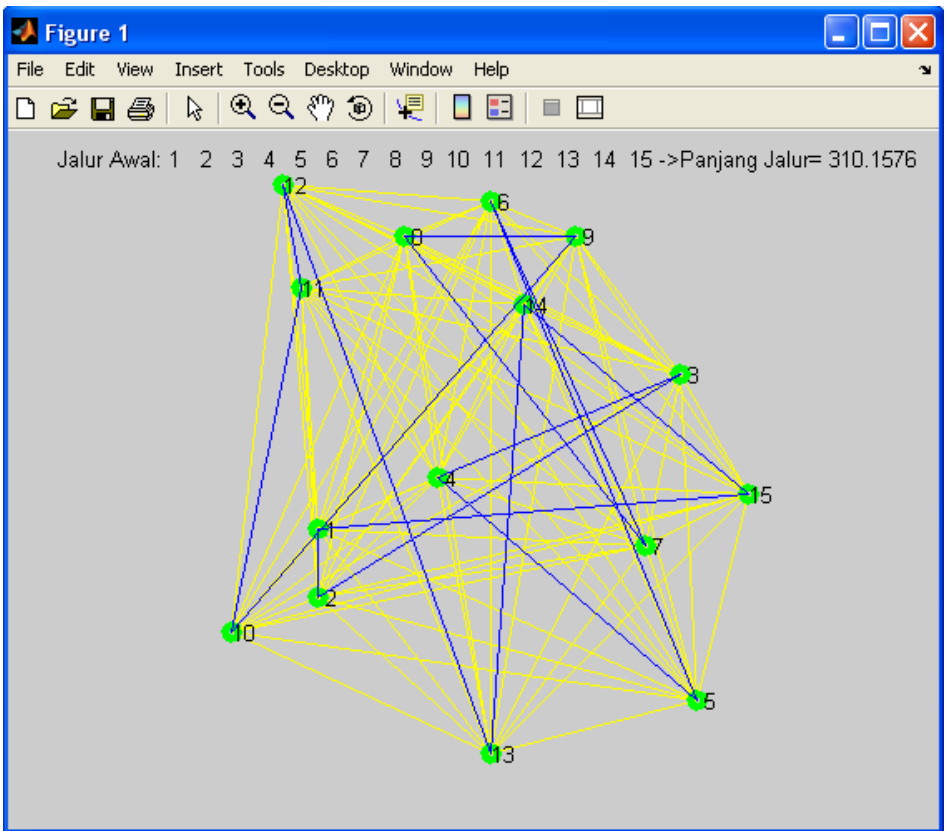
BAB IV
ANALISIS MASALAH

4.1 Tampilan Program

Persoalan TSP yang dibahas pada tugas akhir ini memiliki kompleksitas atau ruang solusi yang jauh lebih besar dari TSP biasa yakni TSP asimetris dan simetris. Kompleksitas pada TSP ini adalah sebesar *n!*. Hal ini dapat dipahami, karena memiliki node asal dan tujuan yang sama maka tidak akan ada panjang jalur yang sama dengan rute yang berbeda.

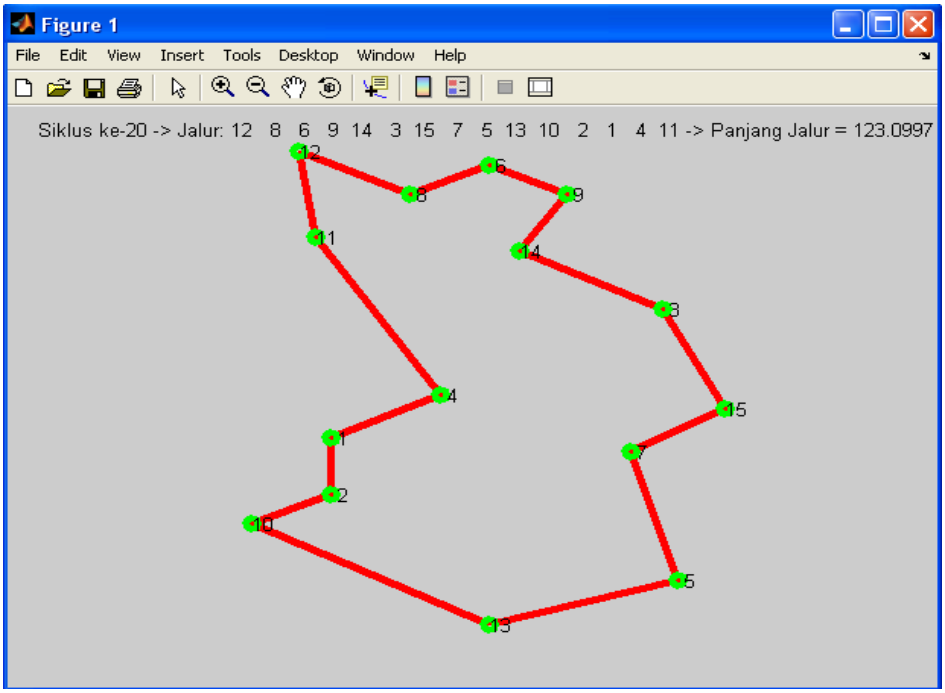
Berikut ini adalah contoh tampilan dari seluruh program setelah dijalankan (*run*) pada program utama dari program yang dirancang pada bab 3.



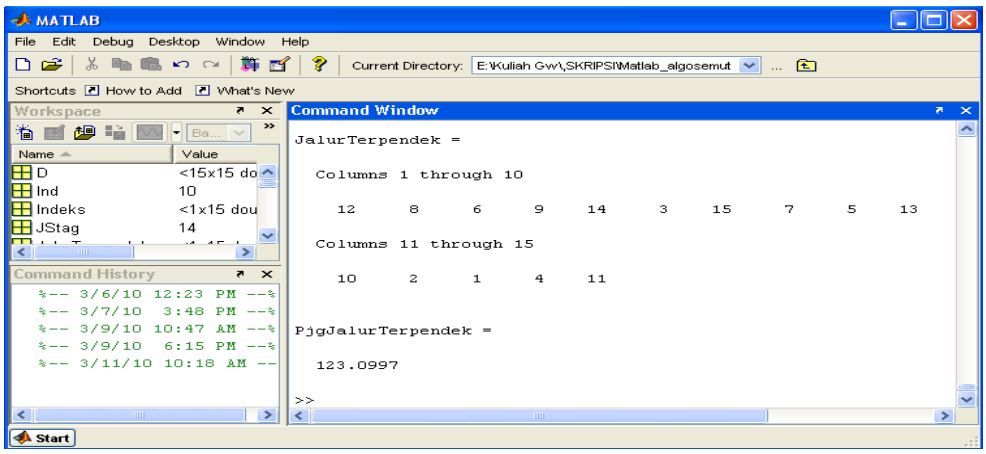
Gambar 4.1 Tampilan setelah program dijalankan

Gambar 4.1 menampilkan program setelah di jalankan dan belum dieksekusi, pengeksekusian dilakukan dengan penekanan sembarang tombol pada keyboard. Keadaan jalur awal berurutan dari 1 sampai 15.

Gambar 4.2 berikut ini adalah contoh gambar bentuk rute atau gambar jalur yang telah dieksekusi dengan siklus sebanyak 20. Grafik dibawah menampilkan jalur dari rute yang dilalui.



Gambar 4.2 Ilustrasi jalur



Gambar 4.3 Urutan jalur terpendek

Gambar 4.3 di atas adalah tampilan pada program Matlab yang menyatakan urutan node dari jalur awal hingga akhir. Hasil terbaik ini tentunya mereprentasikan jalur terbaik yang berisi urutan node yang harus dilalui yang memiliki panjang jalur sependek mungkin. Jalur terbaik ini merupakan solusi akhir yang diharapkan.

Ketiga gambar di atas merupakan contoh tampilan program apabila variabel-variabel input diset seperti berikut:

1. Alfa (α) = 1.00
2. Beta (β) = 1.00
3. Rho (ρ) = 0.50
4. t_{ij} awal = 0.01
5. Maksimun Siklus = 20
6. Koordinat node-node yang dilewati seperti pada tabel 4.1

4.2 Pengujian Program Algoritma Ant Colony System

Apabila pada penentuan nilai awal ditentukan koordinat kota seperti pada tabel 4.1 berikut ini:

Tabel 4.1 Koordinat seluruh node

Kota	Sumbu-X	Sumbu-Y
1	5	13
2	5	9
3	22	26
4	12	16
5	27	3
6	15	32
7	24	12
8	10	30
9	20	30
10	0	7
11	4	27
12	3	33

13	15	0
14	17	26
15	30	15

Pada tabel 4.1 di atas koordinat dapat kita ubah sesuai yang kita ingin kan, bahkan kita bisa menginputkan lebih banyak lagi koordinatnya. Dengan bertambahnya koordinat maka jalur yang di tempuh akan lebih panjang, karena koordinat satu ke koordinat yang lainnya terdapat bobot yang berbeda.

Tabel 4.2 Beberapa hasil pengujian dengan variasi nilai *α* dan *β*

No	Pengendali Intensitas Jejak semut (α)	Pengendali Visibilitas (β)	Panjang Jalur				Rute Terbaik
			Run-1	Run-2	Run-3	Yang Terbaik	
1	1	0,01	146.362	131.979	**123.81**	123.81	1-10-2-13-5-7-15-3-14-9-6-8-12-11-4
2	1	0,1	134.35	**129.35**	149.03	129.35	1-11-12-8-6-9-14-3-15-7-5-13-4-2-10
3	1	0,5	**124.19**	130.332	127.826	124.19	14-9-3-15-7-5-13-10-2-1-4-11-12-8-6
4	1	0,7	**123.099**	**123.099**	124.19	123.099	7-15-3-14-9-6-8-12-11-4-1-2-10-13-5
5	1	1	123.81	**123.099**	**123.099**	123.099	5-7-15-3-14-9-6-8-12-11-4-1-2-10-13
6	0,8	0,01	**144.938**	145.691	152.38	144.938	9-6-14-3-15-5-7-13-2-10-1-4-8-11-12
7	0,8	0,1	146.956	156.305	**144.844**	144.844	1-2-10-13-5-7-3-15-4-11-12-8-6-9-14
8	0,8	0,5	131.98	128.537	**127.826**	127.826	1-2-10-13-5-7-15-3-14-9-6-8-11-12-4
9	0,8	0,7	127.252	**123.099**	124.19	123.099	11-12-8-6-9-14-3-15-7-5-13-10-2-1-4
10	0,8	1	124.190	123.810	**123.099**	123.099	6-8-12-11-4-1-2-10-13-5-7-15-3-14-9
11	0,6	0,01	**157.39**	157.734	174.345	157.39	11-12-8-6-9-7-15-3-14-4-13-5-2-10-1
12	0,6	0,1	**130.1**	159.51	172.69	130.1	9-6-12-11-8-4-1-10-2-13-5-7-15-3-14
13	0,6	0,5	**132.481**	143.87	144.761	132.481	6-8-12-11-1-2-10-13-5-7-15-3-4-14-9
14	0,6	0,7	131.391	**129.208**	136.117	129.208	7-3-9-14-6-8-12-11-4-1-2-10-13-5-15
15	0,6	1	**127.252**	133.391	130.332	127.252	9-3-15-7-5-13-10-2-1-4-11-12-8-14-6
16	0,4	0,01	198.303	**172.641**	184.49	172.641	8-2-10-13-5-4-1-14-7-15-3-9-6-11-12
17	0,4	0,1	189.275	178.896	**162.384**	162.384	5-13-10-2-1-15-7-4-11-12-8-14-9-6-3
18	0,4	0,5	170.54	157.413	**154.736**	154.736	3-15-7-5-13-10-12-11-8-6-9-14-1-2-4
19	0,4	0,7	150.73	138.69	**135.812**	135.812	14-8-12-11-4-1-2-10-13-5-15-7-3-6-9
20	0,4	1	148.245	138.898	**131.978**	131.978	13-5-7-15-3-9-6-14-8-11-12-4-1-2-10
21	0,3	0,01	194.553	196.39	**191.802**	191.802	8-14-11-2-4-9-6-12-3-7-15-5-13-10-1
22	0,3	0,1	**187.05**	191.76	194	187.05	13-1-2-4-15-3-9-8-11-14-6-12-10-7-5
23	0,3	0,5	**155.77**	160.251	177.804	155.77	3-15-7-5-13-2-10-4-1-14-6-9-8-11-12
24	0,3	0,7	162.41	**161.66**	163.155	161.66	3-5-13-4-1-2-10-14-8-11-12-6-9-7-15
25	0,3	1	**145.93**	158.84	151.623	145.93	12-1-2-10-13-7-5-15-3-4-14-9-6-8-11
26	0,2	0,01	204.16	197.72	**195.65**	195.65	12-14-9-3-7-13-1-2-10-15-5-6-8-4-11
27	0,2	0,1	**188.02**	199.313	209.392	188.02	5-3-15-7-9-8-14-6-2-1-10-12-11-4-13
28	0,2	0,5	183.633	**171.88**	177.482	171.88	3-2-4-8-12-11-6-9-14-1-10-13-7-5-15
29	0,2	0,7	**169.564**	181.086	170.502	169.564	15-7-2-10-1-4-8-6-14-9-12-11-13-5-3
30	0,2	1	152.34	168.86	**152.14**	152.14	4-1-2-10-13-5-7-15-8-14-3-9-6-12-11
31	0,1	0,01	202.768	**200.3**	224.192	200.3	10-11-7-5-4-1-12-8-9-6-14-15-3-13-2
32	0,1	0,1	210.77	**194.975**	207.324	94.975	3-14-4-1-6-15-5-13-10-2-12-11-8-9-7

33	0,1	0,5	182.397	182.22	**180.896**	180.896	12-8-9-11-1-10-2-4-14-6-3-15-5-13-7
34	0,1	0,7	182.07	172.78	**150.78**	150.78	10-2-1-4-7-3-14-6-8-11-12-9-15-5-13
35	0,1	1	162.554	157.203	**149.442**	149.442	5-7-15-3-4-9-14-6-8-11-12-10-2-1-13

Dari data pada tabel 4.2 terlihat bahwa jalur terbaik diperoleh pada intensitas jejak semut 0,8 dan 1 serta pengendali visibilitas pada 0,7 dan 1. Hal ini mengandung arti bahwa dengan meng-set nilai intensitas jejak semut dan visibilitas seperti nilai-nilai di atas, maka program algoritma semut dapat mencari solusi mendekati solusi yang diharapkan.

Sebaiknya parameter *pheromon*e lokal diberi nilai $0<\alpha<1$, dimaksudkan untuk menghindari akumulasi feromon yang tidak terbatas pada sisi tersebut. Untuk pemilihan jumlah semut sebaiknya kurang dari atau sama dengan jumlah simpul. Dan sebaiknya penempatan awal semut adalah satu semut untuk satu simpul. Hal ini untuk menjamin adanya lintasan awal dari simpul tersebut pada iterasi pertama.

Tabel di bawah ini menunjukkan beberapa hasil pengujian sistem algoritma semut dengan mengubah-ubah jumlah iterasi atau siklus serta mempertahankan nilai intensitas jejak semut (α) dan pengendali visibilitas (β).

Nilai α yang diambil sebagai sampel adalah 1 sedangkan nilai β yang diambil sebagai sampel adalah 1. Hal ini dikarenakan tabel 4.2 menunjukkan performa algoritma semut yang sangat baik untuk mencari solusi yang ‘bagus’.

Tabel 4.3 Beberapa hasil pengujian dengan variasi nilai α dan β (α = 1 dan β =1)

No	Jumlah siklus	Panjang Jalur				Rute Terbaik
		Run-1	Run-2	Run-3	Yang Terbaik	
1	5	**151.96**	167.34	169.68	151.96	10-4-1-2-13-7-5-15-3-9-14-8-6-12-11
2	10	132.84	133.39	**128.12**	128.12	14-9-6-8-12-11-4-1-2-10-13-5-15-7-3
3	15	**123.099**	**123.099**	**123.099**	123.099	10-2-1-4-11-12-8-6-9-14-3-15-7-5-13
4	20	**123.099**	**123.099**	**123.099**	123.099	7-15-3-14-9-6-8-12-11-4-1-2-10-13-5
5	25	**123.099**	**123.099**	**123.099**	123.099	8-12-11-4-1-2-10-13-5-7-15-3-14-9-6
6	30	**123.099**	**123.099**	**123.099**	123.099	4-11-12-8-6-9-14-3-15-7-5-13-10-2-1

Pada tabel 4.3 terlihat jelas bahwa jumlah iterasi yang dipergunakan memiliki pengaruh yang cukup signifikan, semakin banyak jumlah iterasi yang

dipergunakan maka solusi yang dihasilkan lebih mendekati optimum. Seperti yang terlihat pada jumlah siklus 15, 20, 25, dan 30, yang terbaik dari hasil *running* yang telah dilakukan semuanya sama.

4.3 Pengujian Program Algoritma Genetika

Apabila pada penentuan nilai awal ditentukan koordinat kota seperti pada tabel 4.4 berikut ini:[7]

Tabel 4.4 koordinat node yang akan di uji

Kota	Sumbu-X	Sumbu-Y
XYst	1	1
XYend	20	19
1	4	12
2	3	9
3	3	18
4	6	6
5	16	16
6	8	12
7	8	19
8	9	3
9	7	14
10	13	14
11	15	6
12	17	10
13	15	7
14	13	3
15	7	19

Pada tabel 4.5 di bawah ini menunjukkan beberapa hasil pengujian sistem algoritma genetika dengan mengubah-ubah probabilitas *crossover Pc* dan probabilitas mutasi *Pm*. Apabila jumlah populasi diset 100 dan jumlah generasi diset 100.

Tabel 4.5 Beberapa hasil pengujian dengan variasi nilai *Pc* dan *Pm*

No	Prob. crossover	Prob. mutasi	Panjang Jalur				Rute Terbaik
			Run-1	Run-2	Run-3	Rata2	
1	0,9	0,01	**80,552**	94,224	98,009	90,928	4-8-14-11-13-12-10-5-6-2-1-9-15-3-7
2	0,9	0,02	**86,993**	95,957	91,592	91,514	10-12-13-11-14-8-4-2-3-15-7-6-1-9-5
3	0,9	0,05	92,289	**84,428**	94,686	90,468	8-14-4-2-1-7-15-3-9-6-12-13-11-5-10
4	0,9	0,1	103,627	92,641	**77,899**	91,389	2-6-9-10-15-7-3-1-4-8-14-13-11-12-5
5	0,9	0,3	100,366	**98,865**	103,760	100,997	2-8-13-11-12-14-4-1-9-6-10-5-15-7-3
6	0,9	0,6	102,936	100,695	**94,834**	99.488	4-1-2-9-10-7-3-15-6-14-11-8-13-12-5
7	0,9	0,9	101,565	**100,682**	100,965	101,071	6-9-2-1-10-12-11-13-4-14-8-3-15-7-5
8	0,7	0,01	**81,983**	88,928	85,975	85,629	4-8-14-13-11-12-2-1-9-6-3-15-7-5-10
9	0,7	0,02	92,797	**81,632**	89,756	88,062	2-6-9-7-15-3-1-4-8-12-13-11-14-10-5
10	0,7	0,05	**76,121**	87,201	95,410	86,244	4-8-14-11-13-12-9-3-15-7-1-2-6-10-5
11	0,7	0,1	**81,969**	96,432	91,462	89,954	1-3-15-7-6-10-12-13-11-14-8-4-2-9-5
12	0,7	0,3	100,480	**100,315**	101,291	100,695	4-2-8-12-13-11-14-6-10-5-1-3-9-15-7
13	0,7	0,6	106,268	**98,359**	103,965	102,864	4-2-9-6-10-7-1-3-15-5-13-14-8-11-12
14	0,7	0,9	108,474	98,176	**96,766**	101,139	4-8-11-14-13-6-9-2-15-7-1-3-10-12-5
15	0,5	0,01	**83,362**	86,964	88,164	86,163	4-2-1-3-15-7-9-10-6-12-13-11-14-8-5
16	0,5	0,02	**71,914**	88,057	86,970	82,314	2-4-8-14-11-13-12-10-6-1-9-3-15-7-5
17	0,5	0,05	95,196	**80,745**	90,303	88,748	2-1-7-15-3-6-12-13-11-14-8-4-9-10-5
18	0,5	0,1	**82,780**	88,344	86,625	85,916	4-8-14-13-12-11-2-3-7-15-1-9-6-10-5
19	0,5	0,3	97,155	**94,942**	100,555	97,551	1-15-3-2-4-8-13-11-14-12-10-9-6-7-5
20	0,5	0,6	**95,972**	109,834	106,073	103,959	8-15-7-3-9-4-2-1-6-11-14-12-13-10-5
21	0,5	0,9	**98,626**	100,910	109,754	103,096	2-1-4-14-11-12-13-6-9-7-3-15-8-10-5
22	0,3	0,01	82,828	**76,740**	88,670	82,746	8-14-4-2-1-9-3-15-7-6-11-13-12-10-5
23	0,3	0,02	**75,244**	84,472	78,613	79,443	2-1-9-3-15-7-10-12-13-11-14-8-4-6-5
24	0,3	0,05	86,439	**71,204**	82,928	80,190	8-4-2-1-3-15-7-9-6-14-11-13-12-10-5
25	0,3	0,1	94,453	90,218	**86,924**	90,532	14-8-11-13-4-2-6-1-3-7-15-9-10-12-5
26	0,3	0,3	96,706	104,623	**96,555**	99,295	4-9-5-15-7-3-1-2-6-10-11-14-8-13-12
27	0,3	0,6	109,411	**104,357**	108,091	107,286	4-3-6-9-8-14-11-13-12-7-15-1-2-5-10
28	0,3	0,9	101,613	102,831	**99,613**	101,352	2-1-8-14-11-13-3-15-7-5-10-9-6-4-12
29	0,1	0,01	**70,886**	84,166	82,630	79,227	2-4-8-14-11-13-12-6-9-1-3-15-7-10-5
30	0,1	0,02	74,994	76,048	**73,082**	74,708	8-14-11-12-13-4-2-1-6-9-3-15-7-10-5
31	0,1	0,05	83,606	**80,947**	81,487	82,013	4-8-14-11-13-12-10-3-6-2-1-9-7-15-5
32	0,1	0,1	82,268	**74,402**	89,923	82,197	2-4-8-14-11-13-12-5-10-6-1-9-3-15-7
33	0,1	0,3	97,521	**94,203**	95,614	95,779	4-8-2-1-5-7-15-3-9-6-12-13-11-14-10
34	0,1	0,6	101,417	104,413	**101,335**	102,388	2-4-13-8-14-6-9-15-1-3-7-10-12-11-5
35	0,1	0,9	**92,218**	92,953	96,573	93,915	2-6-1-7-3-15-9-11-14-4-8-13-12-10-5

Dari data pada tabel 4.5 terlihat bahwa rata-rata jalur terbaik diperoleh pada probabilitas *crossover* 0,3 dan 0,1 serta probabilitas mutasi pada 0,01 dan 0,02. Hal ini mengandung arti bahwa dengan menset nilai probabiltas *crossover* dan mutasi seperti nilai-nilai di atas, maka program AG dapat mencari solusi mendekati solusi yang diharapkan.

Semakin tinggi nilai probabilitas pindah silang, semakin cepat struktur baru diperkenalkan dalam populasi. Tetapi, jika probabilitas pindah silang terlalu tinggi, struktur dengan unjuk kerja yang baik dapat hilang dengan lebih cepat dari seleksi sehingga populasi tidak bisa meningkatkan unjuk kerjanya lagi. Sebaliknya, probabilitas yang rendah akan menghalangi proses pencarian.

4.4 Perbandingan ACS Dengan Algoritma Genetik (AG)

Untuk membuktikan keoptimumannya, ACS akan dibandingkan dengan algoritma lain yaitu AG. Dalam melakukan perbandingan ini, ACS menggunakan nilai – nilai parameter, antara lain:

1. Alfa (α) = 1.00
2. Beta (β) = 1.00
3. Rho (ρ) = 0.50
4. t_{ij} awal = 0.01
5. Maksimun Siklus = 20

Sedangkan parameter untuk AG diambil dari Tugas Akhir Nendang Zulfikar "*Aplikasi Algoritma Genetik Untuk Mencari Rute Terpendek N-Buah Node"* Jurusan Teknik Komputer, Universitas Komputer Indonesia 2008.

Berikut adalah nilai – nilai parameternya:[7]

1. Jumlah node *JumGen* = 15
2. Jumlah populasi *PopSize* = 100
3. Probabilitas crossover *Pc* = 0,1
4. Probabilitas mutasi *Pm* = 0,01
5. Jumlah generasi *MaxG* = 100

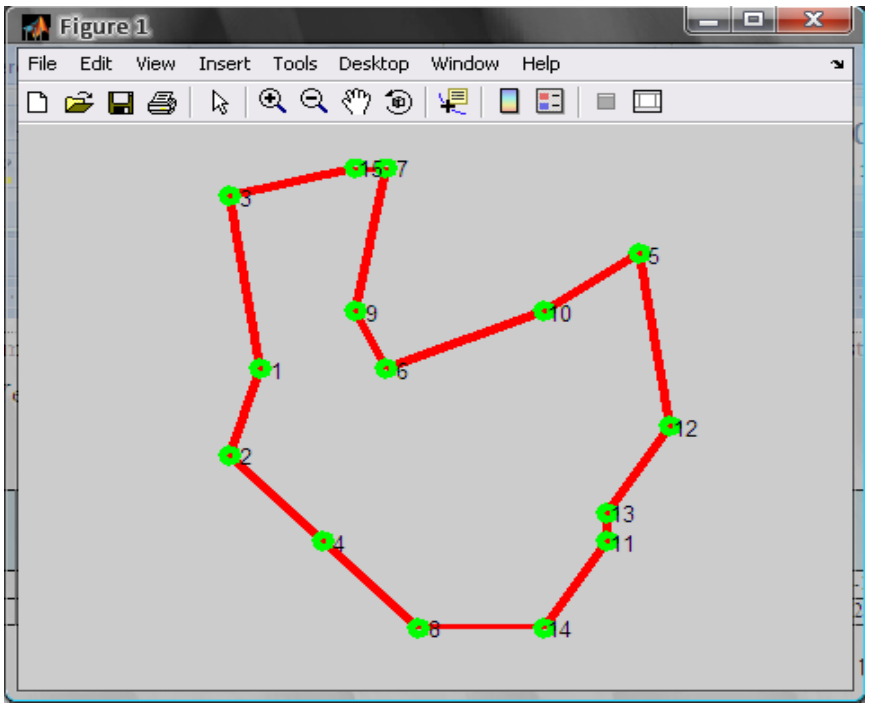
Koordinat yang akan diuji disini menggunakan tabel 4.4. Untuk AG menggunakan koordinat asal XYst (1,1) dan tujuan XYend (20,19) karena dalam AG memiliki koordinat asal dan dan tujuan berbeda. Sedangkan dalam ACS

memiliki koordinat asal dan tujuan yang sama. Jadi dalam tabel 4.6 XYst dan XYend hanya untuk AG saja.

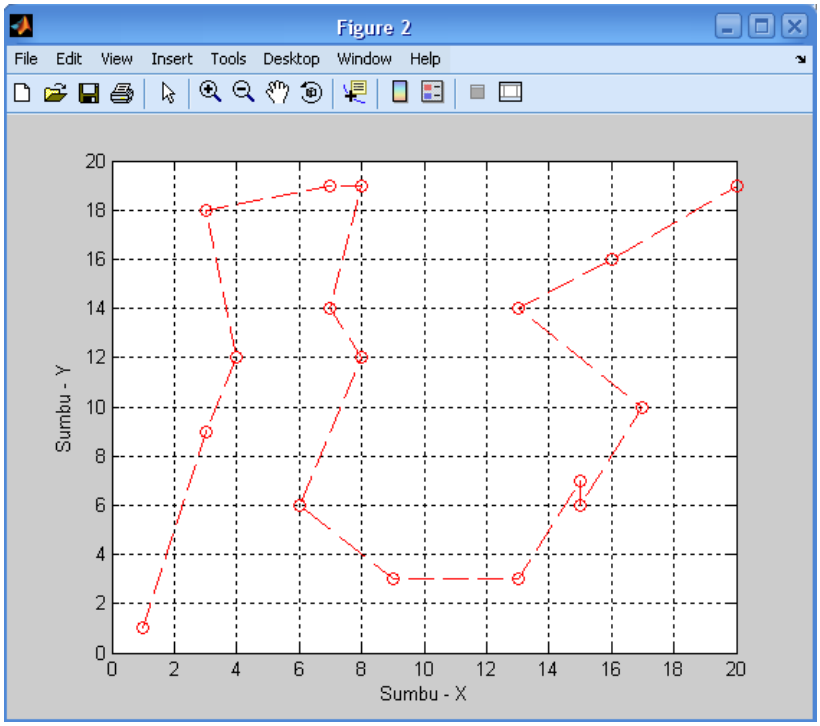
Tabel 4.6 Hasil Perbandingan ACS dengan AG

No	Jenis Algoritma	Panjang Jalur				Rute Terbaik
		Run-1	Run-2	Run-3	Yang Terbaik	
1	**ACS**	**57.4731**	57.6341	**57.4731**	57.4731	5-10-6-9-7-15-3-1-2-4-8-14-11-13-12
2	**AG**	**70.732**	81.101	75.265	70.732	1-3-15-7-9-6-2-4-8-14-11-13-12-10-5

Pada tabel 4.6 terlihat jelas perbandingn ACS dengan AG sangat besar sekali. Pada percobaan ini dilakukan dengan menggunakan nilai inisialisasi terbaik dari masing-masing algoritma, percobaan ini menggunakan 15 kota dan dilakukan *running* sebanyak tiga kali lalu diambil yang terbaiknya.



Gambar 4.4 Tour terbaik untuk ACS dengan 15 kota



Gambar 4.5 Tour terbaik untuk AG dengan 15 kota

Gambar diatas adalah gambar dengan jalur terbaik yang diperoleh dari percobaan menggunakan masing-masing algoritma. Pada tabel 4.4 memperoleh jalur yang terbaiknya yaitu 57.4731, sedangkan untuk tabel 4.5 memperoleh jalur terbaiknya yaitu 70,732.

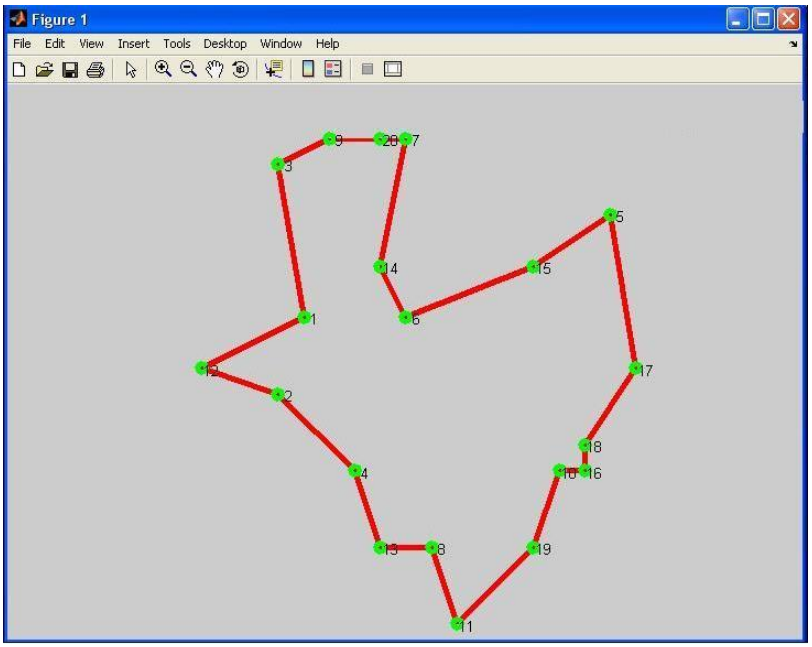
Tabel 4.7 Perbandingan ACS Dengan AG
Menggunakan Jumlah Kota Yang Banyak

No	Jenis Masalah	ACS	AG
1	20 kota	**66.9395**	92.069
2	30 kota	**237.0296**	313.481
3	40 kota	**260.6499**	398.011
4	50 kota	**227.485**	473.983
5	75 kota	**609.924**	1609.86

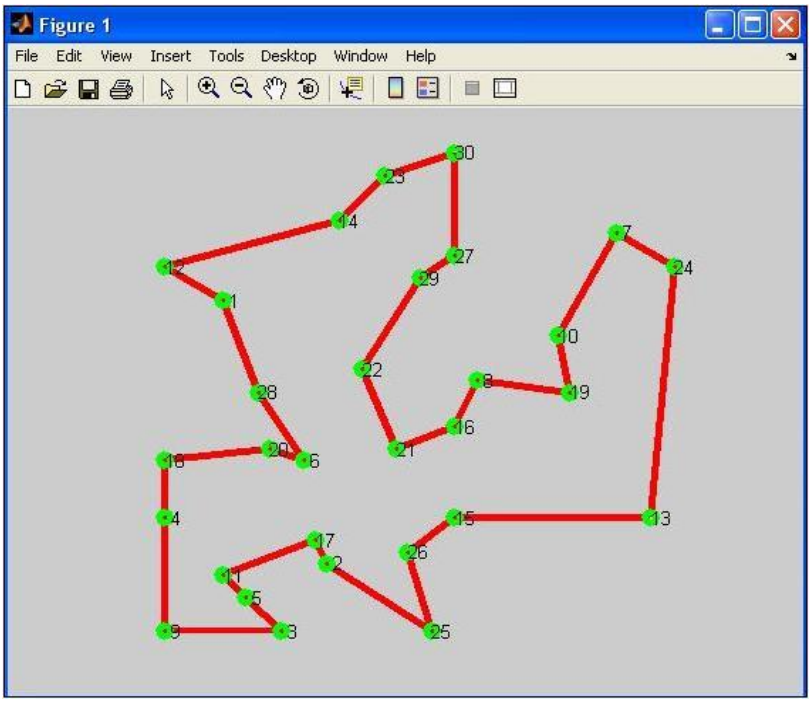
Tabel 4.7 diatas memperlihatkan hasil terbaik dari sepuluh kali *running* yang diperoleh melalui percobaan yang dilakukan dimana terlihat jelas ACS lebih

unggul dari AG. ACS sangat cocok untuk permasalahan optimasi TSP yang lebih besar dan rumit dimana hal ini merupakan kelemahan dari AG. Sebagai bukti pada permasalahan 75 kota ACS mampu menghasilkan tur terbaik yang panjangnya hanya 609.924 sedangkan untuk AG memperoleh hasil hampir tiga kali lipat.

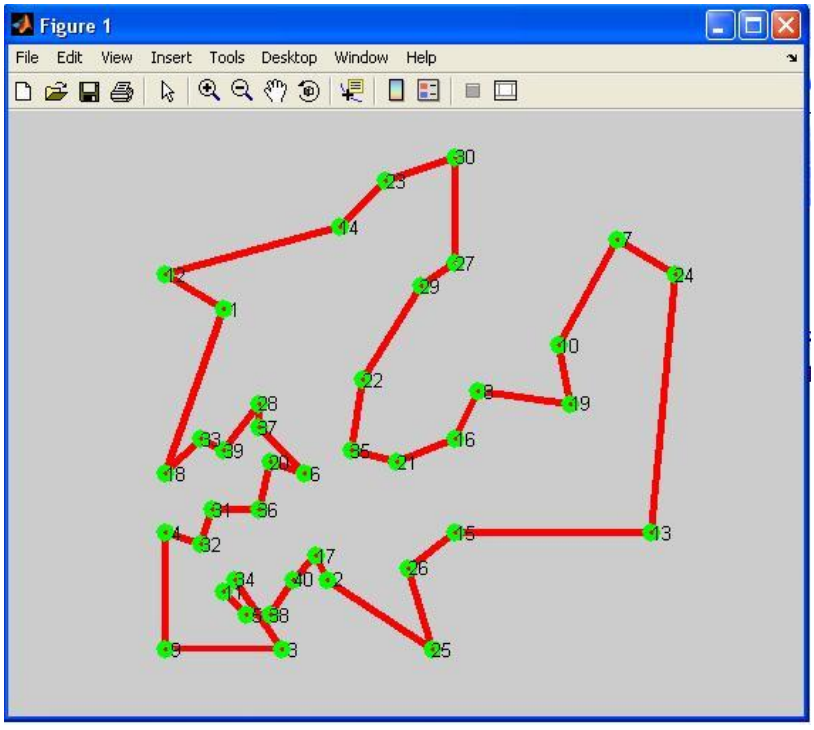
Tour terbaik dari setiap jenis permasalahan adalah yang dicetak tebal dan jalur yang ditempuh untuk mendapatkan tur terbaik dari setiap jenis permasalahan dapat dilihat pada gambar – gambar berikut.



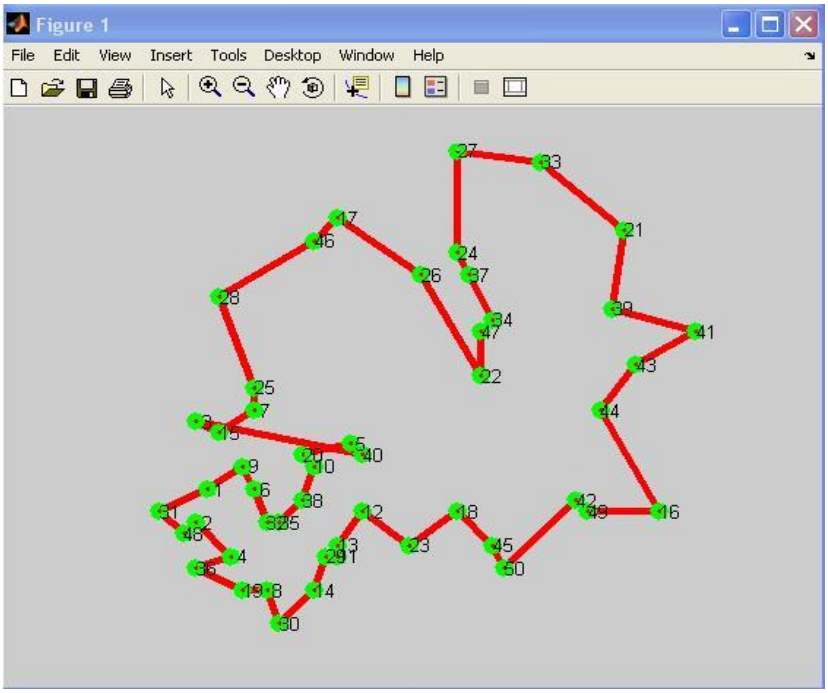
Gambar 4.6 Tour terbaik dari masalah 20 kota



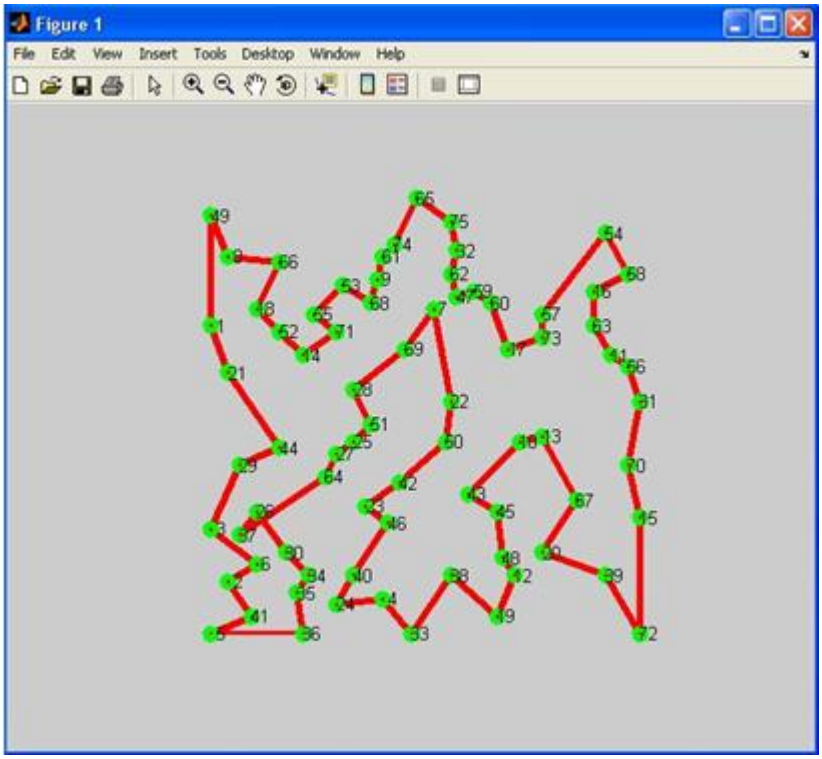
Gambar 4.7 Tour terbaik dari masalah 30 kota



Gambar 4.8 Tour terbaik dari masalah 40 kota



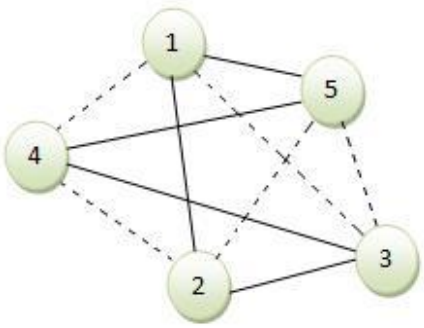
Gambar 4.9 Tour terbaik dari masalah 50 kota



Gambar 4.10 Tour terbaik dari masalah 75 kota

4.5 Contoh Penyelesaian TSP Kurva Tertutup

Diketahui suatu graph:



Gambar 4.11 Contoh Kasus Graph

Dengan jarak antar kota (d_{ij}) sebagai berikut :

Tabel 4.8 Tabel jarak antar kota

Kota	1	2	3	4	5
1	0	5	7	4	2
2	5	0	2	3	6
3	7	2	0	6	3
4	4	3	6	0	5
5	2	6	3	5	0

Parameter yang digunakan adalah:

Alfa (α) = 1.00

Beta (β) = 1.00

Rho (ρ) = 0.50

τ_{ij} awal = 0.01

Siklus maksimum (NC_{max}) = 10

Tetapan siklus (Q) = 1

Banyaknya semut = 5

Dari jarak kota yang telah diketahui dapat dihitung visibilitas antar kota (η_{ij}) = $1/d_{ij}$

Tabel 4.9 Tabel visibilitas jarak antar kota

Kota	1	2	3	4	5
1	0	0.2	0.143	0.25	0.5
2	0.2	0	0.5	0.33	0.166
3	0.143	0.5	0	0.166	0.33
4	0.25	0.33	0.166	0	0.2
5	0.5	0.166	0.33	0.2	0

Dengan mengambil jumlah semut yang sama untuk setiap kota = 1, maka akan diperoleh :

Siklus ke-1:

Isi Tabu Awal:

- Tabu [1,1] = 1
- Tabu [2,1] = 2
- Tabu [3,1] = 3
- Tabu [4,1] = 4
- Tabu [5,1] = 5

Untuk t = 1

Jumlah semut tiap kota = 1 1 1 1 1

Semut ke- 1

- Tabu list = 1
- Probabilitas dari kota 1 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0) + (0.01*0.2) + (0.01*0.143) + (0.01*0.25) + (0.01*0.5) = 0.01093

  Dengan demikian dapat dihitung probabilitas dari kota 1 menuju setiap kota =

  Kota 1 = 0.00

  Kota 2 = $(0.01)^{1.00} .(0.2)^{1.00}$ / 0.01093 = 0.183

  Kota 3 = $(0.01)^{1.00} .(0.143)^{1.00}$ / 0.01093 = 0.130

Kota 4 = $(0.01)^{1.00}$.$(0.25)^{1.00}$ / 0.01093 = 0.229

Kota 5 = $(0.01)^{1.00}$.$(0.5)^{1.00}$ / 0.01093 = 0.458

- Probabilitas komulatif = 0.000 0.183 0.313 0.542 1.000
- Bilangan random yang dibangkitkan = 0.699, maka kota yang terpilih adalah kota 5
- Tabu list = 1 5

Semut ke- 2

- Tabu list = 2
- Probabilitas dari kota 2 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0.2) + (0.01*0) + (0.01*0.5) + (0.01*0.33) + (0.01*0.166) = 0.01196

  Dengan demikian dapat dihitung probabilitas dari kota 2 menuju setiap kota =

  Kota 1 = $(0.01)^{1.00}$.$(0.2)^{1.00}$ / 0.01196 = 0.167

  Kota 2 = 0.00

  Kota 3 = $(0.01)^{1.00}$.$(0.5)^{1.00}$ / 0.01196 = 0.418

  Kota 4 = $(0.01)^{1.00}$.$(0.33)^{1.00}$ / 0.01196 = 0.276

  Kota 5 = $(0.01)^{1.00}$.$(0.166)^{1.00}$ / 0.01196 = 0.139
- Probabilitas komulatif = 0.167 0.167 0.585 0.861 1.000
- Bilangan random yang dibangkitkan = 0.435, maka kota yang terpilih adalah kota 3
- Tabu list = 2 3

Semut ke- 3

- Tabu list = 3
- Probabilitas dari kota 3 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0.143) + (0.01*0.5) + (0.01*0) + (0.01*0.166) + (0.01*0.33) = 0.01139

  Dengan demikian dapat dihitung probabilitas dari kota 3 menuju setiap kota =

Kota 1 = $(0.01)^{1.00}$.$(0.143)^{1.00}$ / 0.01139 = 0.126

Kota 2 = $(0.01)^{1.00}$.$(0.5)^{1.00}$ / 0.01139 = 0.439

Kota 3 = 0.00

Kota 4 = $(0.01)^{1.00}$.$(0.166)^{1.00}$ / 0.01139 = 0.146

Kota 5 = $(0.01)^{1.00}$.$(0.33)^{1.00}$ / 0.01139 = 0.289

- Probabilitas komulatif = 0.126 0.565 0.565 0.711 1.000
- Bilangan random yang dibangkitkan = 0.369, maka kota yang terpilih adalah kota 2
- Tabu list = 3 2

Semut ke- 4

- Tabu list = 4
- Probabilitas dari kota 4 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha}$.$[\eta_{ij}]^{\beta}$ = (0.01*0.25) + (0.01*0.33) + (0.01*0.166) + (0.01*0) + (0.01*0.2) = 0.00946

  Dengan demikian dapat dihitung probabilitas dari kota 4 menuju setiap kota =

  Kota 1 = $(0.01)^{1.00}$.$(0.25)^{1.00}$ / 0.00946 = 0.265

  Kota 2 = $(0.01)^{1.00}$.$(0.33)^{1.00}$ / 0.00946 = 0.349

  Kota 3 = $(0.01)^{1.00}$.$(0.166)^{1.00}$ / 0.00946 = 0.175

  Kota 4 = 0.00

  Kota 5 = $(0.01)^{1.00}$.$(0.2)^{1.00}$ / 0.00946 = 0.211

- Probabilitas komulatif = 0.265 0.614 0.789 0.789 1.000
- Bilangan random yang dibangkitkan = 0.299, maka kota yang terpilih adalah kota 2
- Tabu list = 4 2

Semut ke- 5

- Tabu list = 5
- Probabilitas dari kota 5 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0.5) + (0.01*0.166) + (0.01*0.33) + (0.01*0.2) + (0.01*0) = 0.01196

Dengan demikian dapat dihitung probabilitas dari kota 5 menuju setiap kota =

Kota 1 = $(0.01)^{1.00} .(0.5)^{1.00}$ / 0.01196 = 0.418

Kota 2 = $(0.01)^{1.00} .(0.166)^{1.00}$ / 0.01196 = 0.139

Kota 3 = $(0.01)^{1.00} .(0.33)^{1.00}$ / 0.01196 = 0.276

Kota 4 = $(0.01)^{1.00} .(0.2)^{1.00}$ / 0.01196 = 0.167

Kota 5 = 0.00

- Probabilitas komulatif = 0.418 0.557 0.833 1.000 1.000
- Bilangan random yang dibangkitkan = 0.389, maka kota yang terpilih adalah kota 1
- Tabu list = 5 1

Untuk t = 2:

Jumlah semut tiap kota = 1 2 1 0 1

Semut ke -1 :

- Tabu list = 1 5
- Probabilitas dari kota 5 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0) + (0.01*0.166) + (0.01*0.33) + (0.01*0.2) + (0.01*0) = 0.00696

  Dengan demikian dapat dihitung probabilitas dari kota 1 menuju setiap kota =

  Kota 1 = 0.00

  Kota 2 = $(0.01)^{1.00} .(0.166)^{1.00}$ / 0.00696 = 0.239

  Kota 3 = $(0.01)^{1.00} .(0.33)^{1.00}$ / 0.00696 = 0.474

  Kota 4 = $(0.01)^{1.00} .(0.2)^{1.00}$ / 0.00696 = 0.287

  Kota 5 = 0.00

- Probabilitas komulatif = 0.000 0.239 0.713 1.000 1.000
- Bilangan random yang dibangkitkan = 0.699, maka kota yang terpilih adalah kota 3

- Tabu list = 1 5 3

Semut ke- 2

- Tabu list = 2 3
- Probabilitas dari kota 3 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0.143) + (0.01*0) + (0.01*0) + (0.01*0.166) + (0.01*0.33) = 0.00639

  Dengan demikian dapat dihitung probabilitas dari kota 2 menuju setiap kota =

  Kota 1 = $(0.01)^{1.00} .(0.143)^{1.00}$ / 0.00639 = 0.224

  Kota 2 = 0.00

  Kota 3 = 0.00

  Kota 4 = $(0.01)^{1.00} .(0.166)^{1.00}$ / 0.00639 = 0.260

  Kota 5 = $(0.01)^{1.00} .(0.33)^{1.00}$ / 0.00639 = 0.516
- Probabilitas komulatif = 0.224 0.224 0.224 0.484 1.000
- Bilangan random yang dibangkitkan = 0.435, maka kota yang terpilih adalah kota 5
- Tabu list = 2 3 5

Semut ke- 3

- Tabu list = 3 2
- Probabilitas dari kota 2 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha} .[\eta_{ij}]^{\beta}$ = (0.01*0.2) + (0.01*0) + (0.01*0) + (0.01*0.33) + (0.01*0.166) = 0.01196

  Dengan demikian dapat dihitung probabilitas dari kota 3 menuju setiap kota =

  Kota 1 = $(0.01)^{1.00} .(0.2)^{1.00}$ / 0.01139 = 0.287

  Kota 2 = 0.00

  Kota 3 = 0.00

  Kota 4 = $(0.01)^{1.00} .(0.33)^{1.00}$ / 0.01139 = 0.474

Kota 5 = $(0.01)^{1.00}$.$(0.166)^{1.00}$ / 0.01139 = 0.239

- Probabilitas komulatif = 0.287 0.287 0.287 0.761 1.000
- Bilangan random yang dibangkitkan = 0.694, maka kota yang terpilih adalah kota 4
- Tabu list = 3 2 4

Semut ke- 4

- Tabu list = 4 2
- Probabilitas dari kota 2 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha}$.$[\eta_{ij}]^{\beta}$ = (0.01*0.2) + (0.01*0) + (0.01*0.5) + (0.01*0) + (0.01*0.166) = 0.00866

  Dengan demikian dapat dihitung probabilitas dari kota 4 menuju setiap kota =

  Kota 1 = $(0.01)^{1.00}$.$(0.2)^{1.00}$ / 0.00866 = 0.231

  Kota 2 = 0.00

  Kota 3 = $(0.01)^{1.00}$.$(0.5)^{1.00}$ / 0.00866 = 0.577

  Kota 4 = 0.00

  Kota 5 = $(0.01)^{1.00}$.$(0.166)^{1.00}$ / 0.00866 = 0.192
- Probabilitas komulatif = 0.231 0.231 0.808 0.808 1.000
- Bilangan random yang dibangkitkan = 0.742, maka kota yang terpilih adalah kota 3
- Tabu list = 4 2 3

Semut ke- 5

- Tabu list = 5 1
- Probabilitas dari kota 1 ke setiap kota berikutnya dapat dihitung dengan persamaan (2.1)

  Untuk $\Sigma[\tau_{ij}]^{\alpha}$.$[\eta_{ij}]^{\beta}$ = (0.01*0) + (0.01*0.2) + (0.01*0.143) + (0.01*0.25) + (0.01*0) = 0.00593

  Dengan demikian dapat dihitung probabilitas dari kota 5 menuju setiap kota =

  Kota 1 = 0.00

Kota 2 = $(0.01)^{1.00}$.($0.2)^{1.00}$ / 0.00593 = 0.337

Kota 3 = $(0.01)^{1.00}$.$(0.143)^{1.00}$ / 0.00593 = 0.241

Kota 4 = $(0.01)^{1.00}$.$(0.25)^{1.00}$ / 0.00593 = 0.422

Kota 5 = 0.00

- Probabilitas komulatif = 0.00 0.337 0.578 1.000 1.000
- Bilangan random yang dibangkitkan = 0.853, maka kota yang terpilih adalah kota 4
- Tabu list = 5 1 4

Untuk t = 3:

Jumlah semut tiap kota = 0 0 2 2 1

Semut ke -1 :

- Tabu list = 1 5 3
- Probabilitas dari kota 3 = 0.000 0.750 0.000 0.250 0.000
- Probabilitas komulatif = 0.000 0.750 0.750 1.000 1.000
- Bilangan random = 0.596, maka kota yang terpilih adalah kota 2
- Tabu list = 1 5 3 2

Semut ke -2 :

- Tabu list = 2 3 5
- Probabilitas dari kota 5 = 0.714 0.000 0.000 0.286 0.000
- Probabilitas komulatif = 0.714 0.714 0.714 1.000 1.000
- Bilangan random = 0.634, maka kota yang terpilih adalah kota 1
- Tabu list = 2 3 5 1

Semut ke -3 :

- Tabu list = 3 2 4
- Probabilitas dari kota 4 = 0.556 0.000 0.000 0.000 0.444
- Probabilitas komulatif = 0.556 0.556 0.556 0.556 1.000
- Bilangan random = 0.777, maka kota yang terpilih adalah kota 1
- Tabu list = 3 2 4 1

Semut ke -4 :

- Tabu list = 4 2 3
- Probabilitas dari kota 3 = 0.302 0.000 0.000 0.000 0.698
- Probabilitas komulatif = 0.302 0.302 0.302 0.302 1.000
- Bilangan random = 0.835, maka kota yang terpilih adalah kota 5
- Tabu list = 4 2 3 5

Semut ke -5 :

- Tabu list = 5 1 4
- Probabilitas dari kota 4 = 0.000 0.665 0.335 0.000 0.000
- Probabilitas komulatif = 0.000 0.665 1.000 1.000 1.000
- Bilangan random = 0.585, maka kota yang terpilih adalah kota 2
- Tabu list = 5 1 4 2

Untuk t = 4:

Jumlah semut tiap kota = 2 2 0 0 1

Semut ke -1 :

- Tabu list = 1 5 3 2
- Probabilitas dari kota 2 = 0.000 0.000 0.000 1.000 0.000
- Probabilitas komulatif = 0.000 0.000 0.000 1.000 1.000
- Bilangan random = 0.277, maka kota yang terpilih adalah kota 4
- Tabu list = 1 5 3 2 4

Semut ke -2 :

- Tabu list = 2 3 5 1
- Probabilitas dari kota 1 = 0.000 0.000 0.000 1.000 0.000
- Probabilitas komulatif = 0.000 0.000 0.000 1.000 1.000
- Bilangan random = 0.856, maka kota yang terpilih adalah kota 4
- Tabu list = 2 3 5 1 4

Semut ke -3 :

- Tabu list = 3 2 4 1

- Probabilitas dari kota 1 = 0.000 0.000 0.000 0.000 1.000
- Probabilitas komulatif = 0.000 0.000 0.000 0.000 1.000
- Bilangan random = 0.382, maka kota yang terpilih adalah kota 5
- Tabu list = 3 2 4 1 5

Semut ke -4 :

- Tabu list = 4 2 3 5
- Probabilitas dari kota 5 = 1.000 0.000 0.000 0.000 0.000
- Probabilitas komulatif = 1.000 1.000 1.000 1.000 1.000
- Bilangan random = 0.575, maka kota yang terpilih adalah kota 1
- Tabu list = 4 2 3 5 1

Semut ke -5 :

- Tabu list = 5 1 4 2
- Probabilitas dari kota 2 = 0.000 0.000 1.000 0.000 0.000
- Probabilitas komulatif = 0.000 0.000 1.000 1.000 1.000
- Bilangan random = 0.585, maka kota yang terpilih adalah kota 3
- Tabu list = 5 1 4 2 3

Akhir Siklus ke-1

Jumlah Semut Tiap Kota = 1 0 1 2 1

Isi Tabu Akhir :

- Tabu [1] = 1 5 3 2 4
- Tabu [2] = 2 3 5 1 4
- Tabu [3] = 3 2 4 1 5
- Tabu [4] = 4 2 3 5 1
- Tabu [5] = 5 1 4 2 3

Pada siklus berikutnya isi pada tabu list akan kembali seperti awal atau dikosongkan, begitu juga dengan semut akan berisi masing – masing satu pada tiap kota. Hal ini akan berulang hingga sesuai dengan NC_{max} yang telah ditentukan oleh *user* pada tahap inisialisasi.