

PENERAPAN ALGORITMA GENETIKA PADA SISTEM REKOMENDASI WISATA KULINER

*Agus Wahyu Widodo, **Wayan Firdaus Mahmudy

Program Studi Ilmu Komputer, Jurusan Matematika, FMIPA, Universitas Brawijaya
Jl. Veteran Malang, 65145

E-Mail: *a_wahyu_w@brawijaya.ac.id, **wayanfm@ub.ac.id

Abstrak

Perhatian masyarakat terhadap wisata kuliner mempengaruhi kecenderungan dan gaya dalam berwisata. Pelancong tidak hanya berfikir tentang tempat apa yang layak dikunjungi namun juga memikirkan makanan apa yang dapat dinikmati keunikan dan kenyamanannya. Kondisi ini membawa perubahan pertimbangan seorang pelancong melakukan pilihan terhadap alternatif kunjungan wisata. Rekomendasi tentang faktor pilihan wisata, kesesuaian selera tempat makan, dan waktu perjalanan menjadi hal yang penting. Dalam penelitian ini dilakukan pendekatan terhadap masalah tersebut dengan menganalogikan pada kasus pemilihan rute manakah yang memiliki biaya paling murah untuk dilalui seorang pelancong yang harus mengunjungi sejumlah m daerah tujuan wisata ($m > 1$). Tiap daerah tujuan harus dikunjungi tepat satu kali dan kemudian kembali lagi ke tempat semula (*Travelling Salesman Problem (TSP)*). *TSP* merupakan masalah kombinatorial yang solusi optimumnya hanya dapat didapatkan dengan mencoba semua kemungkinan sehingga memerlukan waktu komputasi yang cukup tinggi. Berbeda dengan fungsi *TSP* konvensional yang tujuannya hanya meminimalkan jarak, kasus rekomendasi ini harus mencocokkan pilihan wisata, kesesuaian selera tempat makan, dan waktu perjalanan. Algoritma Genetika (*GA*) terbukti sesuai untuk menyelesaikan masalah multi obyektif, sehingga dapat diterapkan untuk sistem rekomendasi wisata kuliner Metode *crossover* dengan satu titik potong dan mutasi dengan pergeseran gen pada *GA* mampu menyelesaikan masalah dengan jumlah iterasi kurang dari 40 dan dalam waktu kurang dari 10 detik.

Kata kunci: Algoritma Genetika, *Multi-Objective*, *Travelling Salesman Problem-TSP*, Wisata Kuliner.

Abstract

Recently, culinary tourism tends to be trend and style of the way people travelling. Culinary tourists do not only imagine about travel destinations but also the food that can be tasted the uniquely and comfortably, since this influences the travelers to decide destination options. Tourism options, an appropriate taste of food and travelling time become important things for travelers. This research studies the problem by analogizing cheapest cost to be destined by a traveler who must visit a number of tourist destination m ($m > 1$). Each destination should be visited exactly once and then he goes back again to the place originally, called Traveling Salesman Problem (TSP). TSP is a combinatorial problem in which the optimum solution could only be obtained by testing all possibilities so that it requires relatively high computation time. Unlike conventional TSP, the objective function only minimizes the distance, in the case, however, this recommendation should also be appropriate with the selection of destination, places to eat and time traveling. Genetic Algorithms (GA) proved to solve multi objective problems, so that recommendations system of culinary tourism can be applied. Using one-point crossover and genetic shift mutation, GA can solve the problem within 40 iterations at less than 10 seconds.

Key words: Genetic Algorithm, Multi-Objective, Travelling Salesman Problem-TSP, Culinary Tourism.

PENDAHULUAN

Terdapat kecenderungan bahwa ketika seseorang merencanakan kunjungan wisata, ia akan mencari tempat-tempat yang menarik untuk disinggahi dan sekaligus juga tempat-tempat makan yang unik. Hal ini memunculkan satu istilah baru yaitu wisata kuliner. Wisata kuliner tidak dapat diartikan hanya sebagai kegiatan bepergian atau menyinggahi warung-warung makan atau sejenisnya. Seperti kutipan dari Wikipedia [1], "...*Culinary Tourism is defined as the pursuit of unique and memorable eating and drinking experiences...*", yang berarti bahwa wisata kuliner merupakan usaha pencarian pengalaman makan dan minum yang unik dan sangat berkesan (*memorable*). Pencarian kesan dan keunikan menikmati makanan inilah yang menjadi hal yang menarik dari wisata kuliner.

Sebagai bagian dari kepariwisataan pada umumnya, wisata kuliner juga tidak terlepas dari persoalan penentuan alternatif tujuan dan tempat singgah yang sesuai dengan harapan. Dalam wisata kuliner tempat tujuan dan singgah dapat berupa rumah atau warung makan. Umumnya, harapan pelancong adalah rumah makan yang dikunjungi sesuai dengan selera dan jumlah anggota rombongan yang dibawanya. Selain itu, batasan waktu dan efektifitas rute juga merupakan hal yang perlu diperhatikan.

Pemilihan alternatif dari beberapa kondisi seperti yang telah disebutkan sebelumnya merupakan masalah optimasi. Masalah ini dapat diandaikan sebagai persoalan pemilihan rute manakah yang memiliki biaya paling murah untuk dilalui seorang pelancong yang harus mengunjungi sejumlah m daerah tujuan wisata ($m > 1$). Suatu kondisi dimana tiap daerah harus dikunjungi tepat satu kali dan kemudian kembali lagi ke tempat semula disebut sebagai *Travelling Salesman Problem (TSP)*. Jika pada daerah yang dikunjungi terdapat kendala waktu buka dan waktu tutup (*time-window/TW*) tempat tersebut, maka permasalahan tersebut menjadi *TSP-TW*. *TSP* merupakan masalah kombinatorial yang solusi optimumnya hanya dapat diperoleh dengan mencoba semua kemungkinan sehingga memerlukan waktu komputasi yang cukup tinggi. Untuk mengatasi kasus demikian dapat digunakan metode heuristik, yaitu suatu metode pencarian yang didasarkan atas intuisi atau

aturan-aturan empiris untuk memperoleh solusi yang lebih baik daripada solusi yang telah dicapai sebelumnya [2]. Metode heuristik yang dapat diterapkan pada masalah optimasi misalnya adalah Metode *Hillclimbing*, Algoritma *Simulated Annealing*, dan Algoritma Genetika (*GA*) [3][4][5].

Kasus rekomendasi wisata kuliner memerlukan perubahan model matematis *TSP* konvensional yang tujuannya hanya meminimalkan jarak. Dalam kasus rekomendasi ini banyak tujuan yang ingin dioptimalkan, (multi obyektif) seperti pencocokan pilihan wisata, kesesuaian selera tempat makan, dan waktu perjalanan. Pada setiap tujuan yang dikunjungi juga terdapat kendala waktu buka dan waktu tutup tempat tersebut (*time-window*).

GA merupakan salah satu metode heuristik yang merupakan cabang dari *evolutionary algorithm*, yaitu suatu teknik untuk memecahkan masalah-masalah optimasi yang rumit dengan menirukan proses evolusi makhluk hidup. *GA* terbukti sesuai digunakan untuk menyelesaikan masalah multi obyektif. *GA* berkembang seiring dengan perkembangan teknologi informasi yang sangat pesat. Algoritma ini banyak digunakan dalam bidang fisika, biologi, ekonomi, sosiologi dan lain-lain yang sering menghadapi masalah optimasi dengan model matematika yang kompleks atau bahkan sulit dibangun [6].

Oleh karena itu, pada penelitian ini dikembangkan *GA* yang dipakai untuk memenuhi fungsi multi obyektif. Fungsi multi obyektif tersebut tidak hanya bertujuan meminimalkan jarak, akan tetapi juga memenuhi selera. Beberapa batasan digunakan untuk memenuhi alokasi waktu tertentu. Sebagai contoh, waktu makan siang antara pukul 12.00 sampai dengan 14.00, atau rumah makan X hanya dapat dikunjungi pada pukul 15.00 hingga 21.00.

PEMILIHAN RUTE

Penentuan rute perjalanan merupakan salah satu permasalahan yang sering dihadapi dalam kehidupan sehari-hari. Salah satu contoh yaitu rute manakah yang memiliki biaya paling murah untuk dilalui seorang *salesman* ketika harus mengunjungi sejumlah daerah. Tiap daerah tersebut harus dikunjungi tepat satu kali

kemudian kembali lagi ke tempat semula. Permasalahan tersebut dikenal sebagai *Travelling Salesman Problem (TSP)*. Karena kendala *time-window* yang ada merupakan *soft-constraint* maka penanganannya diletakkan dalam perhitungan *fitness* yang diwujudkan dalam bentuk penalti. Permasalahan *TSP* dimodelkan dalam bentuk graf lengkap. Pada kasus *TSP*, graf lengkap tersebut memiliki n buah simpul yang menyatakan daerah-daerah yang harus dikunjungi oleh *salesman*. Bobot pada tiap garis pada graf tersebut menyatakan jarak tiap daerah [7]. Secara matematis, *TSP* dapat diformulasikan dalam Persamaan (1). Sedangkan faktor kendala dapat ditunjukkan dalam Persamaan (2) dan (3).

$$Z = \min \left\{ \sum_{i=1}^n \sum_{j=1}^n C_{ij} X_{ij} \right\} \quad (1)$$

$$\sum_{i=1}^n x_{ij} = 1, \text{ untuk } j = 1, 2, 3, \dots, n-1 \quad (2)$$

$$\sum_{j=1}^n x_{ij} = 1 \text{ untuk } i = 1, 2, 3, \dots, n-1 \quad (3)$$

Jika i dan j secara berurutan adalah simpul awal dan tujuan suatu perjalanan *salesman*, maka dapat dinyatakan beberapa kondisi berikut:

1. $x_{ij} = 1$ apabila ada perjalanan *salesman* dari simpul i menuju simpul j .
2. $x_{ij} = 0$ apabila tidak ada perjalanan *salesman* dari simpul i menuju simpul j .
3. C_{ij} menyatakan jarak dari simpul i menuju simpul j .

Persamaan (2) dan (3) menjamin bahwa setiap simpul hanya dikunjungi sekali oleh *salesman*.

Pada penelitian ini dilakukan perubahan pada formulasi *TSP*, yaitu penambahan suatu variabel yang mempertimbangkan kesesuaian selera dan waktu pada formulasi standar. Hal ini ditunjukkan dalam Persamaan (4).

$$Z = \min \left\{ \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \right\} + \text{kesesuaian} \quad (4)$$

Kesesuaian yang dimaksud dalam Persamaan (4) adalah nilai dari kesesuaian waktu dan selera. Formulasi selera ini didapatkan dengan cara melakukan evaluasi hasil *GA* secara berulang.

ALGORITMA GENETIKA (GA)

Pada pencarian solusi suatu masalah terkadang dibutuhkan formulasi matematika yang kompleks untuk memberikan solusi yang pasti. Solusi optimum mungkin dapat diperoleh akan tetapi memerlukan proses perhitungan yang panjang dan tidak praktis. Untuk mengatasi kasus tersebut dapat digunakan metode heuristik, yaitu suatu metode pencarian yang didasarkan atas intuisi atau aturan-aturan empiris untuk memperoleh solusi yang lebih baik daripada solusi yang telah dicapai sebelumnya [2]. Metode heuristik tidak selalu menghasilkan solusi terbaik, namun jika dirancang dengan baik akan menghasilkan solusi yang mendekati optimum dalam waktu yang cepat. Algoritma Genetika (*GA*) adalah salah satu cabang *evolutionary algorithms*, yaitu suatu teknik optimasi yang didasarkan pada genetika alami. Untuk menghasilkan suatu solusi optimal, *GA* melakukan proses pencarian di antara sejumlah alternatif titik optimal berdasarkan fungsi probabilitas [8].

Masalah utama pada *GA* adalah bagaimana memetakan satu masalah menjadi satu *string* kromosom. Siklus perkembangan *GA* diawali dengan pembuatan himpunan solusi baru (*initialization*) yang terdiri atas sejumlah *string* kromosom dan ditempatkan pada penampungan populasi. Kemudian dilakukan proses reproduksi dengan memilih individu-individu yang akan dikembangbiakkan. Penggunaan operator-operator genetik seperti pindah silang (*crossover*) dan mutasi (*mutation*) terhadap individu-individu yang terpilih dalam penampungan individu akan menghasilkan keturunan atau generasi baru. Setelah proses evaluasi untuk perbaikan populasi, maka generasi-generasi baru ini akan menggantikan himpunan populasi asal. Siklus ini akan berlangsung berulang kali sampai tidak dihasilkan perbaikan keturunan, atau sampai kriteria optimum ditemukan.

Setelah individu-individu dalam populasi telah terbentuk, maka langkah selanjutnya adalah menghitung nilai *fitness* setiap individu. Fungsi *fitness* sendiri bertujuan untuk mengetahui baik tidaknya solusi yang ada pada suatu individu dan setiap individu pada populasi harus memiliki nilai pembandingnya. Selanjutnya, nilai *fitness* digunakan untuk solusi terbaik dengan cara pengurutan nilai *fitness* dari individu-individu.

Posisi	1	2	3	4	5	6	7	8	9
Gen	5	7	6	1	8	4	9	3	2

Gambar 1. Representasi Kromosom untuk Sembilan Titik Tujuan.

Tabel 1. Hasil Uji Metode Crossover dan Mutasi.

crossover mutasi	Nilai fitness terbaik pada percobaan ke-					Rata-Rata
	1	2	3	4	5	
c1 m1	820	790	820	820	820	814
c1 m2	920	980	880	920	920	924
c2 m1	740	620	520	740	740	672
c2 m2	760	730	650	630	710	696

Setelah kromosom terbentuk, maka selanjutnya dilakukan proses perhitungan fitness. Aturan perhitungan fungsi fitness yang digunakan dapat dilihat pada Persamaan (5). Cost adalah total jarak yang ditempuh dari setiap tempat yang dilewati dari titik awal sampai titik akhir. Dari rumusan tersebut maka nilai fitness akan berbanding terbalik dengan nilai cost. Fitness terbaik, yaitu fitness terbesar, dapat dicapai dengan nilai cost terkecil. Nilai fitness tidak pernah mencapai nilai 1, akan tetapi hanya mendekati 1 untuk fitness terbaik.

$$fitness = \frac{1}{(cost + 1)} \quad (5)$$

Pada penerapannya nanti, nilai fitness dapat dinyatakan dengan bilangan lebih dari 1 atau bahkan nilai minus. Hal ini dapat dilakukan dengan mengubah Persamaan (5) menjadi Persamaan (6), dimana nilai fitness dihitung dengan mengurangkan suatu konstanta dengan nilai cost. Nilai konstanta ini dapat dihitung dari jumlah maksimal cost yang mungkin dari suatu persoalan. Dalam kasus ini, nilai konstanta adalah 1000. Jika digunakan Persamaan (6), maka fitness adalah terbaik jika memiliki nilai terbesar.

$$fitness = c - cost \quad (6)$$

Kromosom dengan nilai fitness terbesar akan terpilih sebagai kromosom terbaik. Metode seleksi yang digunakan adalah seleksi roda roulette, dimana nilai fitness terbesar mempunyai kemungkinan lebih besar untuk

menjadi induk pada generasi selanjutnya. Sedangkan untuk nilai fitness terburuk akan digantikan oleh individu baru dengan nilai fitness yang lebih besar. Dalam penelitian ini digunakan Persamaan (6).

Persamaan (5) perlu disesuaikan jika diterapkan pada kasus multi obyektif seperti pada sistem rekomendasi wisata kuliner ini. Jika terdapat dua obyektif/tujuan, yaitu optimalisasi jarak dan kesesuaian waktu, maka dapat dibuat dua nilai fitness yaitu fitness1 dan fitness2. Fitness akhir adalah jumlah dari kedua fitness yang ada. Pada kedua fitness dapat dikalikan suatu nilai bobot w1 dan w2 jika diharapkan salah satu tujuan dianggap lebih penting atau diprioritaskan daripada tujuan lainnya. Pernyataan ini dapat dituangkan dalam Persamaan (7).

$$fitness = c_1 \cdot fitness_1 + c_2 \cdot fitness_2 \quad (7)$$

Penelitian ini dilakukan dengan cara membentuk model genetika dari perjalanan wisata kuliner. Fungsi fitness dibangun dengan memodifikasi fungsi fitness TSP agar sesuai dengan persoalan multi obyektif. Model genetika yang dihasilkan diterapkan dalam bentuk struktur data dan perangkat lunak. Tahapan selanjutnya adalah pengujian sistem dengan menggunakan data simulasi yang sudah diketahui nilai optimumnya. Kombinasi dari variasi metode crossover (c) dan mutasi (m) merupakan parameter dari pengujian GA genetika yang dibangun. Pengujian ini dilakukan untuk menguji validitas program dan performansi program, serta menemukan nilai parameter yang sesuai guna memperoleh hasil yang optimal.

Representasi Kromosom

Representasi kromosom diperlukan untuk menjelaskan setiap individu dalam populasi. Setiap individu atau kromosom tersusun atas urutan gen dari suatu alfabet. Suatu alfabet dapat terdiri dari digit biner (0 dan 1), floating point, integer, simbol-simbol (seperti A, B, C), matrik, dan lain sebagainya [6].

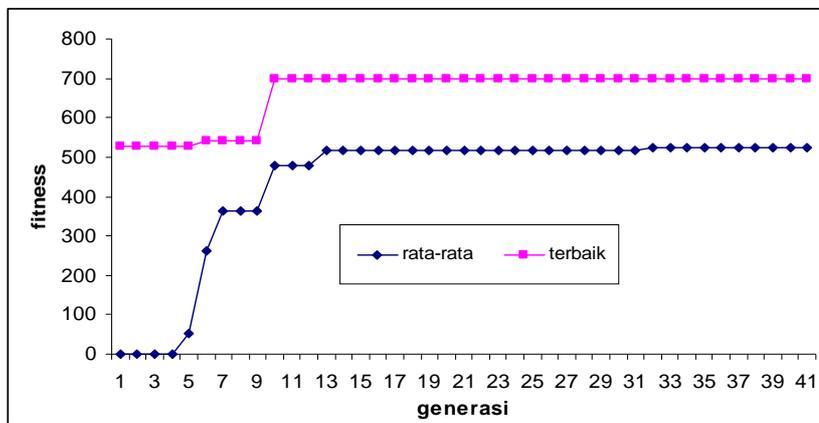
Penelitian ini menggunakan representasi kromosom bilangan bulat bertipe integer. Misalkan terdapat sembilan titik tujuan yang akan dikunjungi, maka kromosom dapat digambarkan pada Gambar 1. Gen-gen yang ada menunjukkan urutan daerah yang akan dikunjungi.

Tabel 2. Daftar Tempat Wisata.

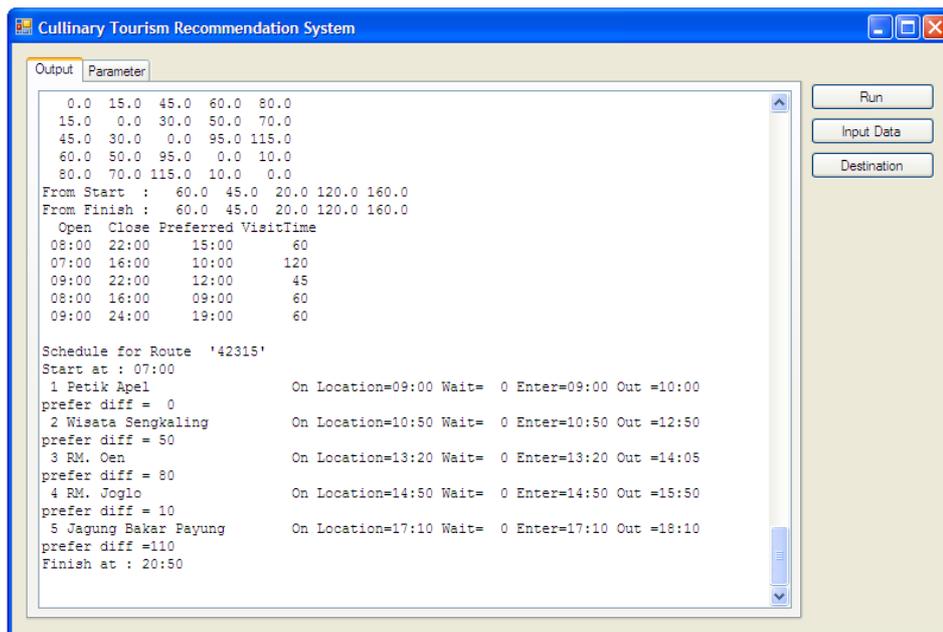
No	Tujuan Wisata	Waktu Buka	Waktu Tutup	Waktu Terbaik	Lama Kunjungan
1	RM. Joglo	08:00	22.00	15.00	60 menit
2	Wisata Sengkaling	07.00	16.00	10.00	120 menit
3	RM. Oen	09.00	22.00	12.00	45 menit
4	Petik Apel	08:00	16.00	09.00	60 menit
5	Jagung Bakar Payung	09:00	24.00	19.00	60 menit

Tabel 3. Jadwal Hasil Uji Coba Contoh Kasus 2.

Urutan	Tujuan Wisata	Tiba di lokasi	Waktu Tunggu	Masuk	Keluar	Beda
1	Petik Apel (4)	09:00	0	09:00	10:00	0
2	Wisata Sengkaling (2)	10:50	0	10:50	12:50	50
3	RM. Oen (3)	13:20	0	13:20	14:05	80
4	RM. Joglo (1)	14:50	0	14:50	15:50	10
5	Jagung Bakar Payung (5)	17:10	0	17:10	18:10	110



Gambar 2. Perubahan Nilai *Fitness*.



Gambar 3. Tampilan Hasil Perangkat Lunak Sistem Rekomendasi Wisata Kuliner.

HASIL DAN PEMBAHASAN

Untuk menguji metode *crossover* dan mutasi yang paling sesuai dilakukan uji coba pada data dengan ukuran populasi sebesar 10 dan generasi maksimum 1000. Nilai $w1$ (bobot untuk optimasi jarak) dan $w2$ (bobot untuk optimasi kesesuaian waktu kunjungan) masing-masing sebesar 0,5. Metode seleksi yang digunakan adalah *elitist selection* yang menampung induk dan *offspring* dalam satu penampungan. Kemudian, sepuluh individu terbaik dipertahankan hidup untuk memasuki generasi selanjutnya. Nilai *crossover rate* ditetapkan 0,2 dan *mutation rate* sebesar 0,1. Simbol yang digunakan untuk metode *crossover* dan mutasi adalah sebagai berikut:

1. $c1$: *crossover* dengan satu titik potong (*one point cut crossover*).
2. $c2$: *crossover* dengan dua titik potong (*two point cut crossover*).
3. $m1$: mutasi dengan pertukaran gen.
4. $m2$: mutasi dengan pergeseran gen.

Tabel 1 menunjukkan hasil uji coba dengan menggunakan kombinasi dari metode *crossover* dan mutasi pada lima kali uji coba. Dari Tabel 1 terlihat bahwa metode $c1$ $m2$ secara konsisten menghasilkan nilai *fitness* terbaik. Hal ini sejalan dengan penelitian Gen (1997) pada kasus penyelesaian masalah *TSP* [3].

Gambar 2 menunjukkan perilaku *GA* dalam menyelesaikan masalah. Pada generasi-generasi awal (di bawah 10), *GA* dengan cepat memperbaiki nilai *fitness* yang didapat. Pada generasi selanjutnya, proses perbaikan *fitness* melambat dan akhirnya didapatkan *fitness* terbaik kurang dari 40 generasi. Pada kasus dengan daerah kunjungan kurang dari 10, *GA* dijalankan 10 kali dan solusi terbaik (*fitness* terbaik) ditemukan pada maksimal generasi ke-40. Untuk menyelesaikan 40 iterasi pada ukuran populasi sebesar 10, *GA* hanya memerlukan waktu kurang dari 10 detik (uji coba dilakukan pada *PC Pentium IV 2,4 GHz*). Hal ini menunjukkan bahwa *GA* sangat efisien untuk menyelesaikan masalah rute perjalanan wisata.

Dalam penelitian ini tidak dilakukan uji coba kombinasi ukuran populasi, *crossover rate* dan *mutation rate* yang terbaik. Hal ini dikarenakan pada kasus kunjungan wisata sangat jarang ditemukan banyaknya daerah kunjungan lebih dari sepuluh. Konfigurasi nilai

parameter yang sudah ditentukan sudah cukup untuk menyelesaikan kasus kunjungan wisata dalam waktu kurang dari 10 detik.

Dengan menggunakan parameter yang sudah ditentukan, metode *crossover* dan mutasi yang sesuai, maka dihasilkan sistem rekomendasi kunjungan wisata kuliner yang siap pakai. Daftar daerah wisata beserta waktu buka, waktu tutup, waktu yang dianjurkan untuk mengunjungi lokasi tersebut, dan jarak antar lokasi wisata sudah tersimpan di basisdata. Pengguna tinggal memilih lokasi wisata yang akan dikunjungi beserta rencana lama kunjungan. Hasil dari sistem ini adalah rute perjalanan yang direkomendasikan. Rute perjalanan ini meliputi lokasi tujuan, waktu tiba, keluar, dan waktu tunggu serta beda waktu antara kedatangan dengan waktu yang dianjurkan untuk mengunjungi suatu lokasi wisata kuliner. Tabel 2 menunjukkan contoh data tempat wisata, jam buka dan tutup, waktu terbaik untuk melakukan kunjungan, serta rencana lama kunjungan. Rute perjalanan yang dihasilkan secara rinci dapat dilihat pada Tabel 3. Kolom beda pada Tabel 3 menunjukkan selisih waktu masuk (yang sebenarnya) dengan waktu terbaik untuk melakukan kunjungan. Tampilan hasil program dapat dilihat dalam Gambar 3.

Selain meminimalkan total jarak tempuh antar tujuan wisata, *GA* juga berusaha meminimalkan selisih waktu masuk (yang sebenarnya) dengan waktu terbaik untuk melakukan kunjungan. Dalam program, pengguna dapat mengatur kecenderungan meminimalkan jarak atau meminimalkan selisih waktu dengan mengatur variabel bobot $w1$ dan $w2$. Misalkan pengguna cenderung untuk memilih kesesuaian waktu kunjungan tanpa menghiraukan total jarak tempuh, maka dapat ditentukan nilai $w1 = 0$ dan $w2 = 1$.

SIMPULAN DAN SARAN

Dari penelitian dapat diambil beberapa simpulan sebagai berikut:

1. Model rekomendasi wisata kuliner dapat dibentuk dengan memodifikasi model fungsi tujuan *TSP* normal menjadi fungsi *TSP* dengan *time-window* (*TSP-TW*) yang mempertimbangkan kesesuaian waktu kunjungan. Modifikasi dilakukan dengan menjumlahkan nilai kesesuaian pada fungsi

TSP dengan *time-window*. Fungsi ini dievaluasi pada tiap tahap regenerasi *GA* yang berlaku.

2. Dalam sistem rekomendasi wisata kuliner dengan menggunakan *GA* ini, nilai *fitness* terbaik didapatkan dari metode *crossover* dengan satu titik potong dan mutasi dengan pergeseran gen. Kombinasi metode *crossover* dan mutasi ini menghasilkan nilai *fitness* rata-rata sebesar 924.
3. Aplikasi yang telah dibuat dapat menghasilkan rute dan jadwal kunjungan wisata dalam waktu yang cepat. Dari hasil pengujian didapatkan bahwa *GA* mampu menyelesaikan masalah dalam 40 iterasi dalam waktu kurang dari 10 detik.

Dari simpulan yang dibuat maka saran yang diberikan adalah:

1. Sistem rekomendasi wisata kuliner dapat dikembangkan lebih lanjut antarmukanya dan dimanfaatkan pada aplikasi kepariwisataan yang sudah ada.
2. Data sekunder tentang jarak antar lokasi wisata yang terdapat pada situs navigasi.net dapat digunakan sebagai data masukan informasi tujuan wisata sistem rekomendasi.
3. Pengujian tingkat akurasi dengan cara membandingkannya dengan sistem yang mungkin serupa (sebagai contoh adalah program *Garmin*) dapat dijadikan alternatif pengujian.

DAFTAR PUSTAKA

- [1] Anonymous. *Culinary tourism*. 2007. URL: http://en.wikipedia.org/wiki/Culinary_tourism, diakses tanggal 28 Maret 2008.
- [2] Taha HA. *Operations Research-An Introduction 6th ed*. Upper Saddle River NJ 07458: Prentice Hall. 2002.
- [3] Gen M and Cheng R. *Genetic Algorithms and Engineering Design*. New York: John Wiley & Sons. 1997.
- [4] Mahmudy WF. Penerapan Algoritma Genetika pada Optimasi Model Penugasan. *Jurnal Natural*. 10: 197-207. 2006.
- [5] Nallamottu UB, Chambers TL, and Simon WE. Comparison of the Genetic Algorithm to Simulated Annealing Algorithm in Solving Transportation Location-allocation Problems With Euclidean Distances. *Proceedings of the 2002 ASEE Gulf-Southwest Annual Conference, The University of Louisiana at Lafayette*. 20-22. 2002.
- [6] Rennard JP. *Genetic Algorithms Viewer 1.0*. 2000. URL: <http://www.rennard.org/alife/english/gavgb.html>, diakses tanggal 26 Mei 2006.
- [7] Setiyono B. Pembuatan Perangkat Lunak Penyelesaian Multi Travelling Salesman Problem (m-TSP). *KAPPA*. 3: 55-65. 2002.
- [8] Michalewicz Z. *Genetic Algorithms + Data Structures = Evolution Programs*. Heidelberg: Springer-Verlag. 1996.