



Optimasi Pada Traveling Salesman Problem (TSP) dengan Pendekatan Simulasi Annealing

Jose Rizal

Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Bengkulu, Indonesia

Diterima 20 Juni; Disetujui 29 Juni 2007

Abstrak – Tulisan ini membahas salah satu penerapan dari simulasi bersyarat (*conditional simulation*) yaitu Simulasi Annealing dalam mencari rute terpendek (optimasi) dari permasalahan *Traveling Salesman Problem* (TSP). Proses Simulasi Annealing analogi dengan proses pada pendinginan logam cair. Dalam aplikasi Simulasi Annealing pada TSP, terdapat proses pertukaran rute-rute perjalanan guna mendapatkan rute perjalanan yang menghasilkan total jarak perjalanan keseluruhan yang minimum. Algoritma Metropolis-Hasting digunakan sebagai kriteria pengujian diterima atau tidaknya pertukaran rute perjalanan dari dua titik. Sebagai studi kasus, diberikan suatu contoh permasalahan TSP dimana untuk menjalankan algoritma Simulasi Annealing menggunakan bantuan Software Matlab.

Kata Kunci: Simulasi Annealing (Conditional Simulation); Traveling Salesman Problem (TSP); Algoritma Metropolis-Hasting

1. Pendahuluan

Salah satu permasalahan dalam bidang transportasi darat adalah mencari suatu rute perjalanan terpendek yang dapat di tempuh dari titik awal keberangkat menuju titik akhir (tujuan), dengan harapan biaya perjalanan yang dikeluarkan dan waktu tempuh perjalanan seminimum mungkin. Masalah seperti ini dikategorikan dalam suatu permasalahan kombinatorial yang lebih dikenal dengan Traveling Salesman Problem (TSP). Inti dari TSP yakni dalam melakukan satu kali perjalanan, seorang salesman diharuskan mengunjungi beberapa tempat tujuan dan diakhiri dengan kembali ke tempat awal keberangkatan. Terdapat beberapa pendekatan yang dapat digunakan dalam mencari solusi optimum dalam menjawab masalah TSP, diantaranya Genetik Algoritma [10], Pendekatan secara Stokastik [9], Neural Network [4] [6] dan Program Linier [5].

Dalam tulisan ini akan dibahas aplikasi Simulasi Annealing dalam mencari solusi optimum masalah TSP. Menurut [2] dan [8], metode ini dinilai ampuh dalam mencari solusi optimum yang bersifat numerik salah satunya pada kasus TSP, karena metode ini mampu menghindari kondisi jebakan optimum lokal [12]. Dua definisi yang digunakan dalam Simulasi Annealing pada

TSP yaitu: Model Distribusi Seragam dan Definisi jarak pada Ruang Euclid.

Definisi 1 [7]

Misalkan X peubah acak kontinu, peubah acak ini hanya memiliki nilai pada suatu selang terbatas (a,b) dan fungsi kepadatan peluang dari X konstan sepanjang selang, $f(x) = c$ dengan $c \in \mathbb{R}$. X berdistribusi Seragam pada selang (a,b) , jika fungsi kepadatan peluang dari X adalah

$$f(x; a, b) = \begin{cases} \frac{1}{b-a} & a < x < b \\ 0 & \text{untuk } x \text{ lainnya} \end{cases} \quad (1)$$

Notasi untuk X peubah acak kontinu yang berdistribusi seragam pada selang (a,b) adalah $X \sim U(a,b)$. Fungsi Distribusi dari X adalah

$$P(X \leq x) = F(x; a, b) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a < x < b \\ 1 & b \leq x \end{cases} \quad (2)$$

Definisi 2 [1]

Misalkan vektor $u = (u_1, u_2, \dots, u_n)$ dan $v = (v_1, v_2, \dots, v_n)$, norma (jarak) Euclides pada \mathbb{R}^n didefinisikan sebagai berikut:

$$d(u,v) = \|u-v\| = \sqrt{(u_1-v_1)^2 + (u_2-v_2)^2 + \dots + (u_n-v_n)^2} \quad (3)$$

Untuk $n = 2$, jarak Euclidis pada R^2 didefinisikan sebagai berikut:

$$d(u,v) = \|u-v\| = \sqrt{(u_1-v_1)^2 + (u_2-v_2)^2} \quad (4)$$

Prinsip Dasar Simulasi Annealing [11]

Metode Simulasi Annealing dikembangkan dari analogi pada proses pendinginan cairan (batuan cair atau logam cair). Pada temperatur tinggi molekul-molekul cairan mempunyai tingkat energi yang tinggi sehingga relatif mudah bergerak terhadap molekul lainnya. Bila temperatur diturunkan, molekul-molekul akan mengatur dirinya untuk mencari konfigurasi dengan tingkat energi yang lebih rendah. Dengan menurunkan temperatur secara perlahan, molekul-molekul tersebut diberi kesempatan untuk mengatur diri sehingga diperoleh suatu keadaan stasioner dengan tingkat energi yang minimum. Bila penurunan temperatur terjadi secara mendadak akan diperoleh *polycrystalline* atau *amorphous* dengan tingkat energi yang tidak minimum.

2. Metode Penelitian

Sesuai dengan proses annealing pada proses pendinginan logam, metode Simulasi Annealing pada TSP terdiri dari 5 (lima) komponen utama, yaitu:

1. Konfigurasi Sistem atau image awal proses, analogi dengan posisi awal iterasi.
2. Fungsi Objektif, fungsi ini didefinisikan sebagai fungsi sasaran yang diminimumkan, analogi dengan energi (total jarak yang ditempuh salesman).
3. Parameter Kontrol, analogi dengan temperatur sistem dan merupakan parameter bebas.
4. Mekanisme untuk merubah konfigurasi interaksi antar titik pengamatan, analogi dengan pertukaran rute-rute perjalanan antar titik.
5. *Annealing/Cooling Schedule*, dinotasikan dengan $T_n = f(T_0, i, N)$, analogi dengan menurunkan temperatur untuk setiap iterasinya.

Mekanisme pertukaran adalah proses iterasi pada Simulasi Annealing yang menyebabkan perubahan harga fungsi objektif. Mekanisme pertukaran merupakan analogi dari interaksi antar molekul pada proses pendinginan logam. Mekanisme pertukaran yang dilakukan pada kasus TSP

adalah dengan menukarkan dua titik berlainan yang dipilih secara random dengan mengambil bilangan bulat, I_1 dan I_2 yang berharga 1 sampai N_t dengan persamaan sebagai berikut:

$$I_1 = 1 + INT(N_p R_1) \quad (5)$$

$$I_2 = 1 + INT(N_p R_2) \quad (6)$$

dimana R_1 dan R_2 merupakan bilangan random yang mempunyai Distribusi Seragam dalam daerah $[0,1]$ dan INT mununjukkan hanya bagian bilangan bulatnya yang diambil.

Dalam menentukan titik yang akan ditukar ada 2 (dua) kondisi yang harus dipenuhi, yaitu:

1. Titik-titik yang akan ditukar mempunyai selisih jarak yang lebih besar dari suatu harga minimum tertentu, sehingga proses simulasi berlangsung efisien.
2. Memenuhi kondisi Metropolis.

Kondisi Metropolis adalah suatu kondisi yang memenuhi persamaan sebagai berikut:

$$P(\Delta E, T) = \begin{cases} 1 & \text{bila } \Delta E \leq 0 \\ \exp\left(-\frac{\Delta E}{KT}\right) & \text{bila } \Delta E > 0 \end{cases} \quad (7)$$

dimana ΔE : Perubahan Energi, T : Temperatur dan K : Konstanta Boltzmann ($1.3860 \times 10^{-23} JK^{-1}$) atau ($8.6174 \times 10^{-5} eV K^{-1}$)

Proses simulasi terdiri dari beberapa langkah dimana setiap langkahnya terdiri atas beberapa iterasi. Proses simulasi tersebut dapat diterangkan dengan algoritma dibawah ini:

1. Membangkitkan kondisi awal simulasi
2. Menghitung Energi awal yang dihasilkan pada kondisi awal
3. Update state awal dengan menggunakan aturan yang bersesuaian dengan permasalahan.
4. Hitung kembali energi yang dihasilkan pada setiap updating.
5. Bangkitkan bilangan random berdistribusi Uniform $[0,1]$.
6. Uji kriteria, bila $p < \exp\left(-\frac{\Delta E}{KT}\right)$ update state

diterima atau pertukaran diterima, lain dari kondisi ini di tolak. Dengan Energi E untuk masalah TSP didefinisikan sebagai berikut

$$E = \sum_{i=1}^{M-1} |x(s_{i+1}) - x(s_i)| + |x(s_1) - x(s_M)| \quad (8)$$

dimana E : Fungsi energi setelah iterasi,
 $x(s_i)$: Posisi dari titik ke-i $|x(s_i) - x(s_j)|$: Jarak dari titik ke-i menuju titik ke-j, M : Jumlah Kota yang dikunjungi.

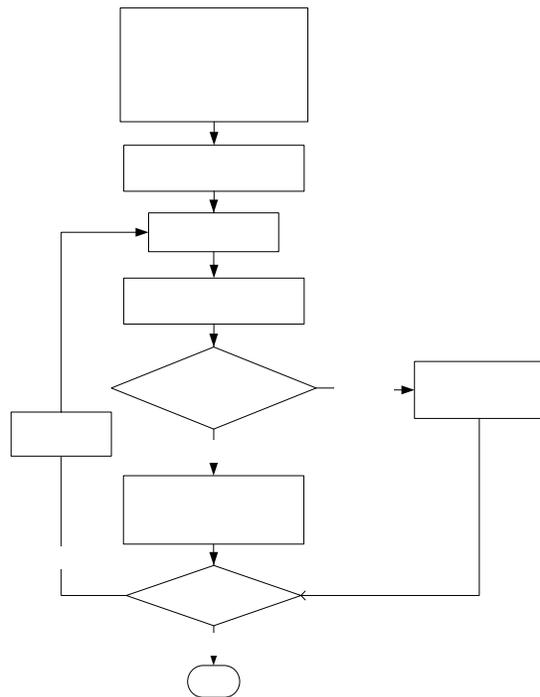
7. Turunkan T (temperatur) dengan fungsi *cooling schedule*. Untuk kasus TSP ini, fungsi *cooling schedule* yang digunakan

$T_i = T_0 \left(\frac{T_n}{T_0} \right)^{\frac{i}{N}}$ dimana

T_i : Temperatur *Cooling Schedule* ke-I, T_0 : Temperatur Awal, T_n : Temperatur *Cooling Schedule*, N : Jumlah Iterasi, i : iterasi ke-i

8. Ulangi langkah ke-3 sampai mencapai kriteria berhenti.

Berikut ini tampilan bagan alur dari Algoritma Simulasi Annealing:



Gambar 1. Diagram Alur Simulasi Annealing pada TSP

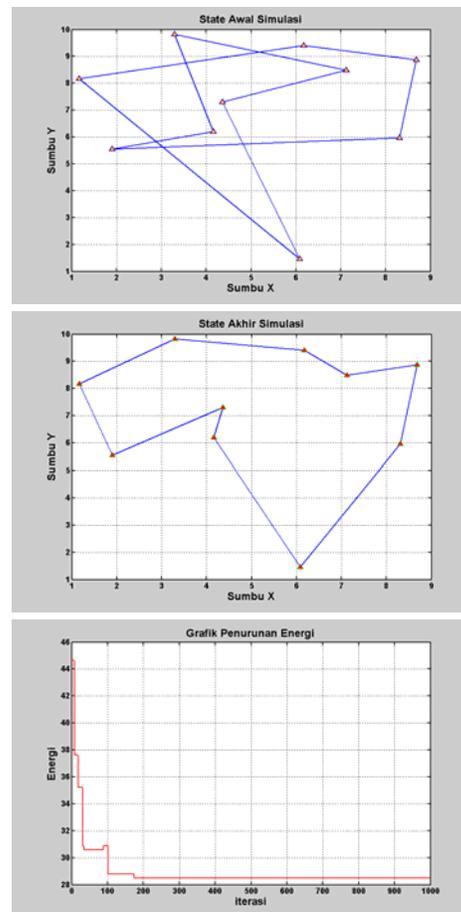
3. Hasil dan Pembahasan

a. Contoh Aplikasi Simulasi Annealing pada kasus TSP

Sebagai contoh penerapan Simulasi Annealing pada kasus TSP, akan dilakukan simulasi pada lokasi-lokasi yang harus dikunjungi dalam satu kali perjalanan. Simulasi dilakukan pada 10 titik lokasi, 20 titik lokasi dan 50 titik lokasi. Dalam setiap simulasinya diberikan parameter-parameter masukan yang sama seperti: temperatur awal ($T_0 = 0.5$), *Cooling Schedule* ($T_n = 0.0001$), dan iterasi ($N=1000$)

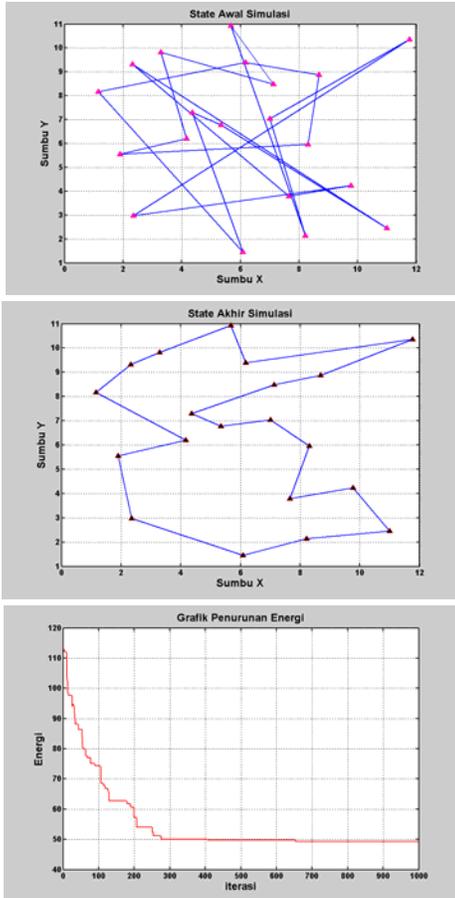
Berikut Hasil Output dari Matlab:

1. Hasil simulasi untuk 10 titik lokasi dengan keluaran antara lain: state awal simulasi, state akhir simulasi dan penurunan energi



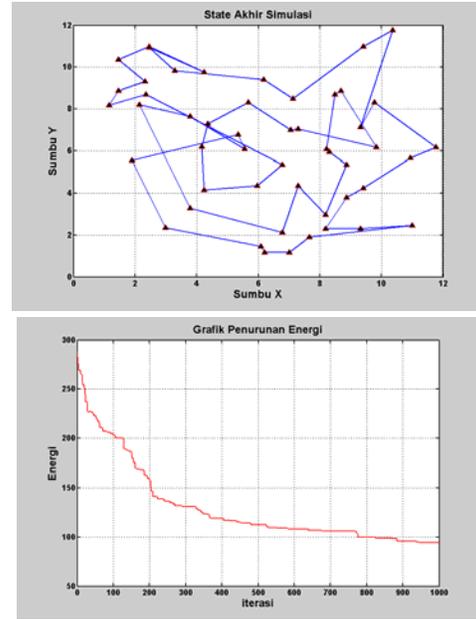
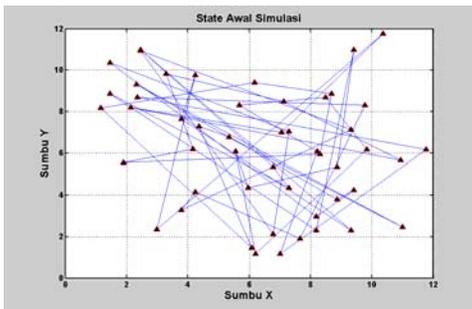
Gambar 2. (atas) State Awal Iterasi (tengah) State Akhir Iterasi (bawah) Penurunan Energi

2. Hasil simulasi untuk 20 titik lokasi dengan keluaran antara lain: state awal simulasi, state akhir simulasi dan penurunan energi



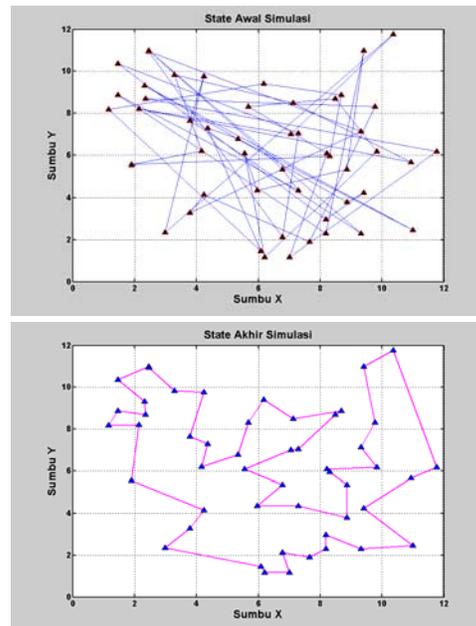
Gambar 3. (atas) State Awal Iterasi (tengah) State Akhir Iterasi (bawah) Penurunan Energi

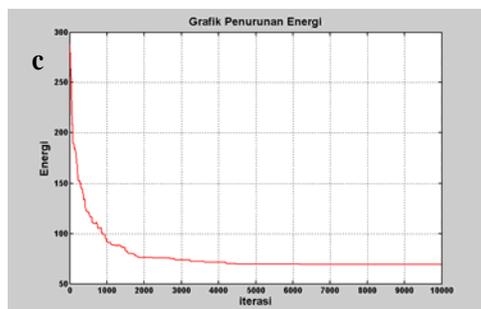
3. Hasil simulasi untuk 50 titik lokasi dengan keluaran antara lain: state awal simulasi, state akhir simulasi dan penurunan energi



Gambar 4. (atas) State Awal Iterasi, (tengah) State Akhir Iterasi, (bawah) Penurunan Energi

3. Hasil simulasi untuk 50 titik lokasi dengan keluaran antara lain: state awal simulasi, state akhir simulasi dan penurunan energi dengan temperatur awal ($T_0 = 0.5$), *cooling schedule* ($T_n = 0.0001$), dan iterasi ($N=10000$)





Gambar 5. (atas) State Awal Iterasi (tengah) State Akhir Iterasi (bawah) Penurunan Energi

4. Kesimpulan

Untuk kasus 10 titik lokasi dan 20 titik lokasi, cukup dengan melakukan 1000 iterasi diperoleh hasil yang optimal. Dapat dilihat dari grafik penurunan energi untuk masing-masing titik lokasi dimana untuk 10 titik lokasi cukup dilakukan dengan 200 iterasi. Sedangkan untuk 20 titik lokasi dibutuhkan 700 iterasi.

Untuk kasus 50 titik, iterasi sebanyak 1000 kali belum menghasilkan state akhir yang optimal, dapat dilihat dari tampilan akhir simulasi, dimana masih banyak rute perjalanan yang bersilangan. Selain itu dari grafik penurunan energi terlihat belum stasioner.

Iterasi yang dibutuhkan untuk mencapai kondisi optimal untuk kasus 50 titik adalah sebanyak 7000 iterasi. Ini ditandai dengan kecendrungan grafik setelah iterasi ditas 7000 cenderung stasioner (flat).

Jumlah iterasi yang menghasilkan kondisi/solusi optimal bergantung pada banyaknya titik lokasi yang disimulasikan. Semakin banyak titik lokasi yang disimulasikan, diperlukan jumlah iterasi yang banyak.

Hasil simulasi dari 10, 20, dan 50 titik lokasi menunjukkan, rute perjalanan yang optimal yaitu dengan mengambil rute terluar dan tidak ada rute yang saling bersilangan.

Daftar Pustaka

- [1] Anton, H., *Aljabar Linier Elementer*, Edisi kelima, **1994**, Erlangga, Jakarta
- [2] Basuki, A, Budi, S.T, dan Huda, M., *Modeling dan Simulasi*, Edisi pertama, **2004**, IPTAQ Mulia Media, Jakarta
- [3] Buckham, B.J and Lambert, C., *Simulated Annealing Application*, **1999**, Mechanical Engineering Department University of Victoria, 1-16
- [4] Budinich, M and Rosario, B.A., *Neural Network for The Traveling Salesman Problem with a Well Behaved Energy Function*, **1995**, MANNA Conference, 1-8
- [5] Diaby, M., *A Linear Programming Formulation of the Traveling Salesman Problem*, **2007**, WSEAS International Conference on Applied Mathematics, 6:97-102
- [6] Gee, A.H and Prager, R.W., *Limitations of Neural Networks For Solving Traveling Salesman Problems*, **1995**, IEEE Transactions on, 6:280-282.
- [7] Hogg, R.V and Craig, A.T., *Introduction to Mathematical Statistics* 5th Ed., **1978**, Prentice-Hall International, Inc. New Jersey. 48p.
- [8] Kirkpatrick, S.S, Gellat, D and Vecchi, M.P., *Optimization by Simulated Annealing*, **1983**, SCIENCE, 220:671-680
- [9] Peres, A.G, and Martin, O.C., *The Stochastic Traveling Salesman Problem: Finite Size Scaling and the Cavity Prediction*, **1999**, Springer Netherlands, 94:739-758
- [10] Potvin, J., *Genetic Algorithms for the Traveling Salesman Problem*, **1996**, Springer Netherlands, 63:337-370
- [11] Rizal, J., *Prediksi Penempatan Sumur Baru Pada Lapangan Panas Bumi Kamojang Menggunakan Metode Simulasi Annealing*, **2005**, Tesis, Institut Teknologi Bandung, Bandung.
- [12] Ross, S.M., *Simulation* 2nd Ed., **1997**, HP Harcourt Academic Press, San Diego California. 236p