

PERBANDINGAN ALGORITMA STEEPEST ASCENT HILL CLIMBING DAN SIMULATED ANNEALING PADA PENYELESAIAN TSP

Haeni Budiati
Fakultas Sains dan Komputer, Universitas Kristen Immanuel Yogyakarta
h3nimu@gmail.com

ABSTRAK

Simulated annealing (SA) adalah salah satu algoritma untuk optimisasi yang bersifat generik. Berbasiskan probabilitas dan mekanika statistik, algoritma ini dapat digunakan untuk mencari pendekatan terhadap solusi optimum global suatu permasalahan.

Steepest Ascent Hill Climbing melakukan pencarian berdasarkan nilai *heuristic* terbaik. Dalam hal ini penggunaan operator tidak menentukan penemuan solusi. *Steepest ascent hill climbing* merupakan metode algoritma yang banyak digunakan untuk permasalahan optimasi. Metode *Hill Climbing* (Pendakian Bukit) adalah teknik *heuristic* yang digunakan untuk mempercepat pencarian solusi. Teknik *heuristic* digunakan untuk mengeliminasi beberapa kemungkinan solusi tanpa harus mengeksplorasinya secara penuh. Selain itu, teknik *heuristic* juga membantu memutuskan kemungkinan solusi mana yang pertama kali perlu dievaluasi.

Kesimpulan yang diperoleh dari penelitian ini menunjukkan bahwa *Simulated annealing* dan *Steepest Ascent Hill Climbing* berhasil memperoleh graph solusi terpendek. Waktu pemrosesan menggunakan metode *Steepest Ascent Hill Climbing* lebih lama dibandingkan dengan Metode *Simulated Annealing* tetapi lebih optimal untuk mencari graph solusi terpendek.

Kata kunci : *Simulated Annealing, Steepest Ascent Hill Climbing, Travelling Salesman Problem.*

1. PENDAHULUAN

Metode Hill Climbing (Pendakian Bukit) adalah teknik heuristik yang digunakan untuk mempercepat pencarian solusi. Teknik heuristik digunakan untuk mengeliminasi beberapa kemungkinan solusi tanpa harus mengeksplorasinya secara penuh. Teknik heuristik juga membantu memutuskan kemungkinan solusi mana yang pertama kali perlu dievaluasi. Ada beberapa metode pencarian heuristik salah satunya adalah metode Hill Climbing. Metode Hill Climbing sering digunakan jika terdapat fungsi heuristik yang baik untuk mengevaluasi state (Rich, 1991)

Hill climbing adalah metode yang dikenal untuk pencarian lokal. Gagasan untuk metode Hill Climbing adalah mulai secara acak state yang sudah ada; bergerak ke tetangga dengan nilai evaluasi yang terbaik; dan jika suatu minimum lokal telah dicapai lalu memulai lagi secara acak pada state yang berbeda. Pengulangan prosedur ini dilakukan hingga solusi ditemukan. Algoritma ini mengatur suatu parameter yang disebut Max-Flips, untuk membatasi jumlah maksimum langkah dalam setiap kali pengulangan dan membantu meninggalkan minimum lokal yang tidak dibatasi. Faktanya, algoritma Hill Climbing harus menyelidiki dan memperhitungkan semua tetangga dari state sekarang sebelum bergerak, dan ini cukup memakan waktu.

Simulated annealing (SA) adalah salah satu algoritma untuk optimisasi yang bersifat generik. Berbasiskan probabilitas dan mekanika statistik, algoritma ini dapat digunakan untuk mencari pendekatan terhadap solusi optimum global suatu permasalahan. Masalah yang membutuhkan pendekatan Simulated Annealing adalah masalah-masalah optimisasi kombinatorial, dimana ruang pencarian solusi yang ada terlalu besar, sehingga hampir tidak mungkin ditemukan solusi eksak terhadap permasalahan itu. Publikasi tentang pendekatan ini pertama kali dilakukan oleh S. Kirkpatrick, C. D. Gelatt dan M. P. Vecchi, diaplikasikan pada desain optimal hardware komputer, dan juga pada salah satu masalah klasik ilmu komputer yaitu Travelling Salesman Problem.

Annealing adalah satu teknik yang dikenal dalam bidang metalurgi, digunakan dalam mempelajari proses pembentukan kristal dalam suatu materi. Agar dapat terbentuk susunan kristal yang sempurna, diperlukan pemanasan sampai suatu tingkat tertentu, kemudian dilanjutkan dengan pendinginan yang perlahan-lahan dan terkendali terhadap materi tersebut. Pemanasan materi di awal proses Annealing, memberikan kesempatan pada atom-atom dalam materi itu untuk bergerak secara bebas, mengingat tingkat energi dalam kondisi panas ini cukup tinggi. Proses pendinginan yang perlahan-lahan memungkinkan

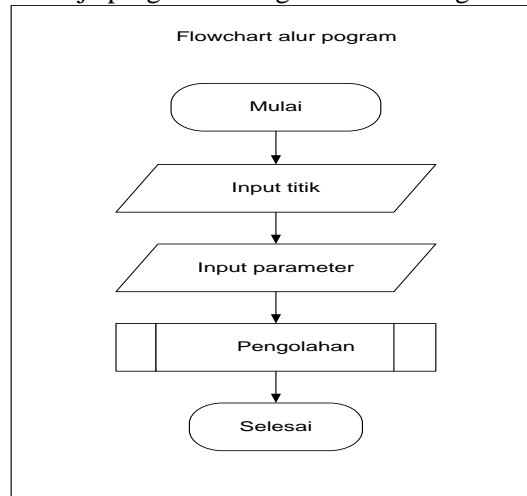
atom-atom yang tadinya bergerak bebas itu, pada akhirnya menemukan tempat yang optimum, dimana energi internal yang dibutuhkan atom itu untuk mempertahankan posisinya adalah minimum.

2. METODE PENELITIAN

2.1. Metode Penelitian

2.1.1. Perancangan Umum

Secara keseluruhan proses kerja program ini digambarkan dengan flowchart pada gambar 1.



Gambar 1. Perancangan umum program

2.1.2. Perancangan Form Utama

1. Gambar titik, area untuk menampilkan banyaknya jumlah titik yang diinputkan.
2. Toolbar adalah tool-tool untuk mengeksekusi perintah di aplikasi sesuai dengan fungsinya.
3. Jalur graf hasil perhitungan, area untuk menampilkan animasi jalur graf hasil perhitungan setelah dilakukannya pemrosesan dengan metode Simulated Annealing dan Steepest Ascent Hill Climbing.
4. Jalur terpendek, jalur yang paling minimum yang dilalui untuk mendapatkan graph jalur terpendek.

2.1.3. Menentukan Kebutuhan Sistem

Berikut ini adalah penjelasan tentang komponen-komponen apa saja yang digunakan dalam pembuatan aplikasi.

1. Timage berfungsi untuk menampilkan objek berupa gambar
2. TmenuFile adalah komponen berupa menu file pull-down.
3. Tpanel adalah komponen berbentuk panel untuk meletakkan komponen lain.
4. Tstringgrid adalah komponen berbentuk tabel.

2.1.4. Proses kerja algoritma

Proses kerja dibagi dalam dua metode yaitu menggunakan metode Simulated Annealing dan Steepest Ascent Hill Climbing untuk mencari jalur yang paling terpendek. Pada awal proses kerja dengan metode Simulated Annealing, User harus menentukan nilai awal parameter yang terbagi atas tiga input parameter yaitu temperature awal (T), temperature akhir (t), serta berapa besar temperature tersebut akan berkurang atau faktor penurunan nilai temperature (F). User juga dapat menginputkan jarak pada layar form utama, setelah mengaktifkan tombol tool edit atau menggunakan jarak yang sudah terandom oleh program. Selanjutnya User harus menginputkan bilangan random yang dibutuhkan untuk menentukan jalur yang akan dimodifikasi atau dapat menggunakan bilangan random yang sudah terandom oleh program, berdasarkan syarat penggunaan bilangan random serta perhitungan penerimaan nilai P tergantung dari besarnya nilai t, dimana P akan diterima apabila angka random lebih kecil dari nilai probabilitas(P).

Proses menggunakan Steepest Ascent Hill Climbing dilakukan dengan menentukan ruang keadaan yang berisi semua kemungkinan lintasan yang mungkin. Selanjutnya penggunaan operator untuk menukar posisi kota-kota yang bersebelahan sehingga kita dapat membandingkan jalur mana yang akan digunakan untuk dijadikan successor dan successor pertama yang lebih baik dan mendekati tujuan akan dijadikan next state.

2.1.4. Output

Proses yang paling terakhir dalam pembuatan aplikasi ini adalah proses output yang juga dibagi dalam dua proses output yaitu output jalur graph hasil perhitungan dan output jalur terpendek yang dibuat dengan komponen Tmemo dimana output jalur graph hasil perhitungan direpresentasikan dalam bentuk animasi urutan – urutan kota yang dilalui sedangkan output jalur terpendek adalah jalur yang paling minimum dari setiap kota setelah diproses dan direpresentasikan dalam bentuk angka.

2. 2. Landasan Teori

Algoritma Steepest Ascent hill climbing

1. Mulai dari keadaan awal, lakukan pengujian : jika merupakan tujuan maka akan berhenti; dan jika tidak, lanjutkan dengan keadaan sekarang sebagai keadaan awal
2. Kerjakan hingga tujuan tercapai atau hingga iterasi tidak memberikan perubahan pada keadaan sekarang.
 - a. Tentukan SUCC sebagai nilai heuristic terbaik dari successor- successor
 - b. Kerjakan untuk tiap operator yang digunakan oleh keadaan sekarang:
 - i. Gunakan operator tersebut dan bentuk keadaan baru
 - ii. Evaluasi keadaan baru tersebut. Jika merupakan tujuan, keluar. Jika bukan, bandingkan nilai heuristicnya dengan SUCC. Jika lebih baik, jadikan nilai keadaan heuristic baru tersebut sebagai SUCC. Namun, jika tidak lebih baik, nilai SUCC tidak berubah
 - iii. Jika SUCC lebih baik dari pada nilai heuristic keadaan sekarang, ubah node SUCC menjadi keadaan sekarang.

Pada Steepest Ascent Hill Climbing ada 3 masalah yang mungkin sebagai berikut :

1. Local Optimum : keadaan semua tetangga lebih buruk atau sama dengan keadaan dirinya.
2. Plateau : Keadaan semua tetangga sama dengan dirinya.
3. Ridge : Local optimum yang lebih disebabkan karena ketidakmampuan untuk menggunakan 2 operator sekaligus.

Algoritma: Simulated Annealing

1. Evaluasi keadaan awal. Jika start adalah tujuan maka pencarian solusi selesai. Jika tidak lanjutkan dengan keadaan awal sebagai keadaan sekarang
2. Inisialisasi BEST_SO_FAR untuk keadaan sekarang
3. Inisialisasi T (Temperature) sesuai dengan annealing schedule
4. Kerjakan hingga solusi ditemukan atau sudah tidak ada operator baru lagi akan diaplikasikan ke kondisi sekarang
 - b. Gunakan operator yang belum pernah digunakan untuk menghasilkan keadaan baru
 - c. Evaluasi kondisi baru dengan menghitung:

$$\Delta E = \text{nilai sekarang} - \text{nilai keadaan baru} \dots\dots\dots (\text{persamaan 1})$$
 - i. Jika keadaan baru adalah tujuan maka keadaan baru tersebut menjadi solusi akhir
 - ii. Jika keadaan baru bukan tujuan, namun nilainya lebih baik dari sekarang, maka jadikan keadaan tersebut sebagai keadaan sekarang
 - iii. Jika keadaan baru tidak lebih baik daripada keadaan sekarang, maka tetapkan keadaan baru sebagai keadaan sekarang dengan probabilitas $\dots\dots\dots$ (Persamaan 2)
dalam kondisi ini juga buat angka random dengan range yang sama dengan range P. Jika angka random lebih kecil dari P maka solusi diterima. Jika angka random lebih besar dari P maka solusi tidak diterima
 - d. Perbaiki T (Temperature) sesuai dengan annealing scheduling
5. BEST_SO_FAR adalah jawaban yang dimaksud

Tahapan proses penyelesaian Travelling Salesman Problem dengan Algoritma Simulated Annealing :

 1. Tentukan nilai awal parameter temperature (T) dan temperature akhir (T1): $0 < T1 < T$.
 2. Fungsi atau faktor penurun nilai temperature (F) : $0 < F < 1$.
 3. Solusi awal (urutan kota yang dikunjungi).
 4. Jarak total perjalanan (E).
 5. Jika $r < 0.5$ tengah di balik urutannya, jika $r \geq 0.5$ Gandeng muka belakang.
 6. Evaluasi solusi baru dengan rumus $\Delta E = E_{skrg} - E_{baru}$.
 7. Proses akan berhenti apabila nilai $T \leq T1$.

3. HASIL DAN PEMBAHASAN

3.1. Hasil Penelitian

Tabel 1. Perbandingan *Simulated Annealing* dan *Steepest Ascent Hill Climbing* pada *Sample*

No	Jumlah titik	Jarak <i>Simulated Annealing</i>	Jarak <i>Steepest Ascent</i>	Hasil terpendek
1	5	509,34	509,34	Sama
2	6	615,17	615,17	Sama
3	7	787,14	698,48	Sama
4	8	846,68	934,97	<i>Simulated Annealing</i>
5	9	1084,06	989,25	<i>Steepest Ascent</i>
6	10	1254,97	1084,65	<i>Steepest Ascent</i>
7	11	1699,61	1315,52	<i>Steepest Ascent</i>
8	12	1645,6	1455,99	<i>Steepest Ascent</i>
9	13	1924,34	1504,22	<i>Steepest Ascent</i>
10	14	2235,44	1530,57	<i>Steepest Ascent</i>

- Hasil penelitian menunjukkan metode *Simulated Annealing* dan *Steepest Ascent Hill Climbing* dapat memperoleh graph solusi.
- Hasil perbandingan jumlah *temperature* yang sedikit dalam metode *Simulated Annealing* menunjukkan bahwa metode *Simulated Annealing* kurang optimal jika dibandingkan dengan metode *Steepest Ascent Hill Climbing* dalam pencarian graph solusi.
- Graph solusi dengan metode *Simulated Annealing* menunjukkan dalam pengujian terdapat sama jalur dan jarak yang sama walaupun menggunakan pengujian *temperature* yang berbeda

3.2. Perbandingan Waktu pemrosesan

Waktu pemrosesan kedua metode *Simulated Annealing* dan *Steepest Ascent Hill Climbing* akan dibandingkan dengan perolehan tabel pengujian sebagai berikut.

Spesifikasi komputer :

Processor : Intel dual-core

Ram : 2 GB

VGA : 128 Mb

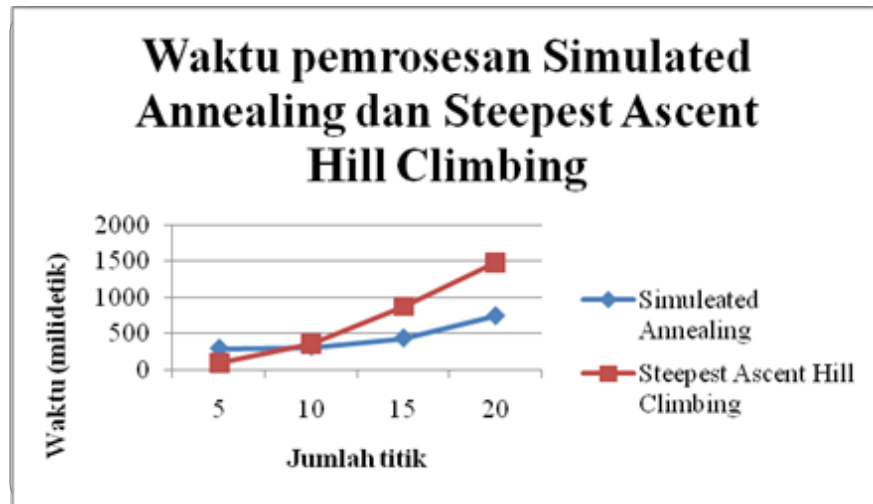
Hardisk : 120 GB

Simulated Annealing

T : 100000

T1 : 50

F : 0.8



Gambar 2. Grafik Waktu pemrosesan *Simulated Annealing* dan *Steepest Ascent Hill Climbing*

3.3. Pembahasan

1. Grafik pada gambar 2 terlihat bahwa semakin banyak titik pengujian sampel maka waktu pemrosesan yang digunakan untuk mencari solusi graph pada Metode *Steepest Ascent Hill Climbing* dan *Simulated Annealing* waktu proses akan semakin lama.
2. Metode *Steepest Ascent Hill Climbing* lebih lama dibandingkan dengan Metode *Simulated Annealing*.

4. KESIMPULAN

1. Metode *Simulated Annealing* dan Metode *Steepest Ascent Hill Climbing* berhasil memperoleh graph solusi.
2. Perbandingan Metode *Simulated Annealing* dan *Steepest Ascent Hill Climbing* dalam pemrosesan serta dalam memperoleh graph solusi adalah sebagai berikut :
 - a. Metode *Simulated Annealing* membutuhkan input parameter sedangkan *Steepest Ascent Hill Climbing* tidak membutuhkan input parameter.
 - b. Graph solusi dengan jarak terpendek dan jumlah temperature yang sedikit dalam metode *Simulated Annealing* membuat metode ini kurang optimal dibandingkan dengan metode *Steepest Ascent Hill Climbing* dalam pencarian solusi.
 - c. Bilangan random yang diinputkan oleh user dan beberapa temperature yang berbeda pada metode *Simulated Annealing* akan menghasilkan graph solusi yang sama dalam pemrosesan. Graph solusi yang diperoleh dalam pengujian bilangan random yang diinputkan oleh user menunjukkan bahwa metode *Simulated Annealing* kurang optimal dibandingkan metode *Steepest Ascent Hill Climbing*.
 - d. Hasil perbandingan graph solusi dengan data jarak yang sama menunjukkan bahwa graph solusi lebih menggunakan metode *Steepest Ascent Hill Climbing* lebih optimal dibandingkan metode *Simulated Annealing*.
3. Waktu pemrosesan dengan metode *Steepest Ascent Hill Climbing* lebih lama jika dibandingkan dengan metode *Simulated Annealing*.

5. SARAN

Setelah melakukan uji coba, metode-metode dalam mencari solusi graph terpendek dapat terus dikembangkan supaya proses dalam mencari graph solusi lebih optimal. Metode pada aplikasi sebaiknya dapat dikembangkan untuk menyelesaikan *Travelling Salesman Problem* dengan menggunakan metode Heuristik yang lainnya.

DAFTAR PUSTAKA

- [1]. Kusuma Dewi, Sri, *Artificial Inteligence : Teknik dan Aplikasinya*, Graha Ilmu, Yogyakarta, 2003.
- [2]. Kusuma Dewi, Sri, *Metode Heuristik*, Graha Ilmu, Yogyakarta, 2005
- [3]. Suyanto, ST. M.sc., *Artificial Inteligence : Searching, Reasoning, Planning and Learning*, 2007, Informatika, Bandung.
- [4]. <http://www.cs.cf.ac.uk/Dave/AI1/steep.html>
- [5]. <https://www.scribd.com/doc/60569625/4/Algorithm-Steepest-Ascent-Hill-Climbing>
- [6]. <http://katrinaeg.com/simulated-annealing.html>
- [7]. <http://mathworld.wolfram.com/SimulatedAnnealing.html>
- [8]. <http://mathworld.wolfram.com/TravelingSalesmanProblem.html>