

**Hendrik Christian Janifer Simanjuntak**

**1301150021**

**IF 39 - 07**

LAPORAN TUGAS PROGRAM KECERDASAN BUATAN 1

SIMULATED ANNEALING

|  |
| --- |
| **CCH3F3**  **ARTIFICIAL INTELLIGENCE**  **SEMESTER GANJIL 2017/2018** |
| **S1 TEKNIK INFORMATIKA**  **FAKULTAS INFORMATIKA** |

**Deskripsi Masalah**

***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 dari suatu permasalahan. Masalah yang membutuhkan pendekatan SA adalah masalah yang menggunakan rumus probabilitas yang memungkinkan bisa keluar dari local minimum, ketika new state tidak lebih baik dari current state,maka new state tersebut masih mungkin dipilih dengan probabilitas.

***Bagaimana menentukan nilai minimum dari sebuah fungsi menggunakan algoritma Simulated Annealing ?***

**Rancangan Metode**

Pada masalah ini, program akan mencari nilai minimum dari fungsi dengan menggunakan algoritma Simulated Annealing. Funsi minimum yang harus dicari adalah :

Dengan batasan

Simulated Annealing adalah metode yang biasa digunakan dalam bidang metalurgi atau proses pembentukan Kristal dalam suatu materi, untuk mendapatkan susunan Kristal yang sempurna maka dibutuhkan suhu yang tinggi dan menurunkan suhu secara perlahan dan terkendali. Metode ini digunakan pada proses SA, dengan membandingkan nilai state sekarnag dengan state baru dan mencari state terkecil, jika temperature annealing masih tinggi maka SA masih dapat menerima state yang lebih besar dari state yang lebih baik, dan semakin rendah temperature akan semakin sempit kemungkinan pergantian state besar.

Logika dalam membuat Simulated Annealing :

* Evaluasi *Initial State.* Jika state ini adalah *goal state,* maka kembalikan fungsi state tersebut sebagai solusi akhir dan *exit* dari program. Jika tidak, maka masukkan *initial state* ke *current state*.
* Inisialisasikan BEST\_SO\_FAR dengan *current state.*
* Inisialisasi T sesuai dengan *annealing schedule.*
* Ulangi sampai solusi ditemukan atau tidak ada lagi aturan produksi yang bisa diterapkan ke *current state.*
* Pilih sebuah aturan produksi yang belum pernah diterapkan ke current state dan terapkan aturan produksi tersebut untuk menghasilkan new state.
* Evaluasi nilai new state. Hitung :

= f(new state )– f(current state)

* Jika new state adalah goal, maka kembalikan state ini sebagai solusi setelah itu keluar dari program.
* Jika new state bukan goal tetapi lebih baik daripada current state, maka masukkan new state ke current state dan masukkan BEST-SO-FAR ke new state.
* Jika new state tidak lebih baik dari current state, maka assign current state dengan new state dengan rumus probabilitas berikut :

Langkah ini juga diimplementasikan dengan menggunakan nilai bilangan acak (R) dengan range dari 0 sampai 1. Jika untuk mendapatkan nilai yang lebih baik, maka range-nya antara nilai 0.8 sampai dengan 0.99.

Jika nilai probabilitas tersebut kurang dari R, maka masukkan *current state* ke *new state*. Tetapi jika nilai probabilitas lebih dari nilai R, maka abaikan saja.

* Tampilkan nilai dari BEST\_SO\_FAR sebagai solusi.

**Hasil Dari Program yang dibuat**

Dengan menggunakan program yang telah saya buat dan dengan menggunakan model akurasi yang ditentukan.

adalah nilai minimum yang didapatkan.

adalah nilai minimum yang diasumsikan paling realistis oleh dosen.

Dengan menggunakan model diatas program akan menghasilkan outputan sebagai berikut:

=== Hasil Simulated Annealing ===

Nilai x1 = -6.441644983574114

NIlai x2 = -8.718310594418778

Fx(x1,x2) = 0.09182926337445219

Akurasi Model = 193.87804910836985%

**Screenshot Output Program**

