**LAPORAN TUGAS 1 – SIMULATED ANNEALING**

Nama: Muhammad Rizky Pahlawan

NIM: 1301178549

Kelas: IF-41-GAB05

**Soal**

Lakukan analisis, desain, dan implementasi algoritma ***Simulated Annealing*** (SA) ke dalam suatu program komputer untuk menemukan **nilai minimum** dari fungsi:

dengan batasan dan .

**Analisis**

Diketahui sebuah fungsi sebagai berikut:

Untuk mencari nilai minimum dari fungsi tersebut dengan batasan -10 ≤ x1 ≤ 10 dan -10 ≤x2 ≤ 10, digunakan salah satu algoritma dari *informed search*, yaitu *Simulated Annealing.* Singkatnya, *Simulated Annealing* merupakan modifikasi dari algoritma *Hill Climbing.* Modifikasi tersebut merupakan penambahan suatu probabilitas ‘coba-coba’ dengan harapan terhindar dari kemungkinan terjebak di dalam *local minimal.*

Terdapat 10 langkah untuk menjalankan algoritma *Simulated Annealing*. Di antara langkah-langkah tersebut, terdapat beberapa langkah yang dinilai sangat vital atau sangat menentukan hasil akhir yang ingin dicapai, yaitu *Best So Far* (BSF). Langkah-langkah tersebut antara lain: inisiasi nilai awal dari x1 dan x2, inisiasi nilai awal T, pemilihan teknik pengurangan T hingga T bernilai 0, serta pemilihan teknik untuk men-*generate* x1 baru dan x2 baru. Adapun rincian dari langkah-langkah tersebut akan dijabarkan pada bagian strategi.

Kemudian, untuk menyelesaikan permasalahan di atas, saya memilih menggunakan bahasa pemrograman *Phyton*.

**Strategi**

Sebagaimana yang telah disinggung sebelumnya, terdapat 4 langkah yang sangat menentukan dalam pemecahan masalah di atas. Adapun rinciannya sebagai berikut.

1. Inisiasi nilai awal dari x1 dan x2

Walaupun biasanya inisiasi awal suatu nilai adalah 0, namun pada kasus ini nilai awal dari x1 dan x2 dapat ditentukan secara bebas (*random*). Saya memilih untuk tidak banyak berspekulasi untuk nilai awal dengan memberi nilai 1 untuk x1 dan nilai -1 untuk x2.

1. Inisiasi nilai awal T

Nilai T merujuk pada nilai *Temperature*, dimana besaran nilai tersebut akan berkontribusi dalam proses pengulangan. Disini saya memilih untuk memberikan nilai T=1000

1. Pemilihan teknik pengurangan T

Awalnya saya menggunakan teknik pengurangan T=T-1 (T dikurangi 1 untuk setiap perulangan). Hal ini berarti program akan menjalankan perulangan sebanyak 1000 kali. Kemudian, saya mencoba untuk menggunakan teknik pengurangan T=T-0.1 (T dikurangi 0.1 untuk setiap perulangan). Hal tersebut mengakibatkan program akan menjalankan perulangan 10 kali lebih banyak. Walaupun dalam pengaplikasiannya teknik ini memakan waktu lebih lama dari teknik sebelumnya, namun keluaran yang dihasilkan pun lebih baik.

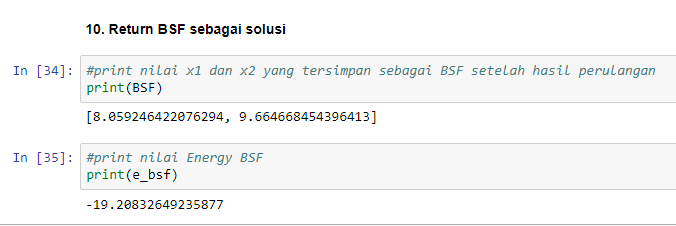
1. Pemilihan teknik untuk men-*generate* x1 baru dan x2 baru

Pada langkah ini, diperlukan kejelian dan kreativitas untuk mendapatkan suatu teknik untuk men-*generate*x1 baru dan x2 baru. Hal ini agar nilai yang dieksplorasi oleh program lebih beragam sehingga dapat mendekati kepada nilai minimum global dari fungsi. Disini saya memilih teknik eksplorasi sebagai berikut.

* Nilai x1 baru = x1 awal + x, dimana x merupakan nilai random dari (0<x< 1). Apabila hasil penjumlahannya bernilai lebih dari 10, maka nilai x1 baru = -10.
* Nilai x2 baru = x2 awal - x, dimana x merupakan nilai random dari (0<x< 1). Apabila hasil penjumlahannya bernilai kurang dari -10, maka nilai x2 baru = 10.

**Hasil *running* terbaik**

Setelah melakukan 10 langkah dalam algoritma *Simulated Annealing,* didapati hasil sebagai berikut.



Nilai BSF menyimpan nilai x1 = 8,059.. dan x2 = 9,664.., sehingga menghasilkan energi BSF sebesar -19,208...

Jika melihat kepada nilai E0 (Energi awal) yang sebesar -0,787.., maka dapat dihitung nilai Delta E akhir sebesar: ΔE = E\_BSF – E\_awal = -19,208.. – (-0,787..) = -18,420... Maka, dapat disimpulkan bahwa nilai ΔE bersifat bagus dan teknik yang digunakan cukup optimal.