**LAPORAN TUGAS PROGRAM**

**SIMULATED ANNEALING**

*Disusun untuk memenuhi tugas mata kuliah Kecerdasan Buatan yang*

*diampu oleh :*

*Untari Novia Wisesty, S.T., M.T*



**Disusun Oleh :**

**Fero Resyanto**

**1301154318**

**IF-39-10**

**JURUSAN S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**UNIVERSITAS TELKOM**

**BANDUNG**

**2017**

1. **Deskripsi Masalah**

Simulated Annealing adalah algoritma untuk optimasi yang mana biasa digunakan untuk penyelesaian masalah perubahan keadaan dari suatu kondisi ke kondisi yang lainnya. Pada kasus ini diketahui sebuah fungsi sebagai berikut :

Dimana nilai dan merupakan bilangan riil dengan Batasan dan . Dengan menggunakan algoritma pencarian Simulated Annealing, dilakukan optimasi dari fungsi tersebut untuk mencari niali minimum global.

1. **Rancangan Metode**

Pada kasus ini digunakan algoritma pencarian Simulated Annealing dengan beberapa parameter sebagai berikut :

* 1. Temperatur

Parameter temperatur pada kasus ini berfungsi agar solusi yang ditemukan tidak terjebak pada nilai minimum lokal. Pada algoritma Hill Climbing sering mengalami solusi yang terjebak di minimum lokal. Oleh karena itu, parameter temperatur digunakan untuk menjadikan solusi baru yang tidak lebih baik dari solusi terbaik mendapatkan peluang untuk dipilih kembali sebagai solusi terbaik. Tingginya nilai temperatur maka semakin besar peluang solusi baru terpilih. Pada kasus ini nilai temperatur awal adalah 100.000. Pada program temperatur menggunakan nama variabel t\_now.

* 1. Alpha

Alpha merupakan faktor penurun suhu yang berguna untuk digunakan pada proses perhitungan solusi berikutnya agar tidak terjebak pada solusi awal yang mungkin tidak optimal. Diasumsikan pada kasus ini nilai factor penurun suhu/alpha adalah 0,995. Pada program Alpha menggunakan nama variabel alpha.

* 1. Iterasi

Iterasi atau perulangan merupakan berapa banyak proses tersebut akan dilakukan. Pada kasus ini iterasi diset mulai dari 0. Sedangkan untuk maksimal iterasinya diset sebanyak 500.000. pada program iterasi awal menggunakan nama variabel i, sedangkan untuk iterasi maksimal menggunakan nama variabel iterasi\_maks.

1. **Algoritma**
   1. Cari nilai random untuk dan . Simpan nilai sebagai batasanBest
   2. Hitung nilai dari fungsi dengan memasukan nilai dan yang telah didapatkan sebelumnnya kedalam fungsi, simpan hasil sebagai nilai hasilBest.
   3. Selama nilai iterasi awal lebih kecil dari iterasi maksimalnya, maka lakukan :
      1. Cari nilai random baru untuk dan . simpan nilai sebagai batasanNext.
      2. Hitung kembali nilai dari fungsi dengan memasukan dan yang baru. Simpan hasil sebagai nilai hasilNext.
      3. Jika nilai next lebih kecil dari nilai best, maka lakukan :
         1. Jadikan nilai hasilNext menjadi nilai hasilBest.
         2. Jadikan nilai batasanNext menjadi batasanBest.
      4. Jika tidak, maka lakukan :
         1. Cari nilai random antara 0 hingga 1. Simpan dalam variabel Z.
         2. Hitung probabilitas penerimaan nilai hasilNext, yaitu : eksponen (( hasilBest – hasilNext)/t\_now). Simpan hasil kedalam variabel Y.
         3. Jika nilai Y lebih besar dari nilai Z , maka lakukan :
            1. Jadikan nilai hasilNext menjadi nilai hasilBest.
            2. Jadikan nilai batasanNext menjadi batasanBest.
      5. Kalikan nilai t\_now dengan alpha.
      6. Iterasi ditambah 1.
   4. Outputkan nilai hasilBest dan batasanBest yang merupakan hasil akhir pencarian.

Algoritma ini diimplementasikan menggunakan bahasa pemrograman PHP dibantu dengan software XAMPP sebagai server. Berikut hasil yang didapatkan setelah melakukan beberapa kali percobaan :

* = -0.088435178552359.
* = 0.71235132730708.
* Hasil = -1.031620402664.

1. **Output Program**

Berikut screenshot hasil yang didapat dari program yang dijalankan :

