Laporan tugas besar

*Simulated Annaeling* (SA)



Disusun oleh :

Ivana Azharyani (1301154497)

IF 39-07

Kecerdasan Buatan

Semester Ganjil 2017/2018

1. **Deskripsi masalah (studi kasus)**

Gunakan algoritma ***Simulated Annealing*** (SA) untuk menemukan nilai minimum dari fungsi

dengan batasan dan

* setiap mahasiswa hanya boleh melakukan satu-dan-hanya-satu kali *running* program dengan nilai-nilai parameter SA yang (berdasarkan sejumlah eksperimen) dianggap paling optimal. Hasil *running* ini digunakan untuk menghitung **Akurasi Model** berdasarkan rumus

di mana *f*A adalah nilai minimum yang Anda dapatkan dan *f*R adalah nilai minimum yang diasumsikan paling realistis oleh dosen. Sebagai contoh, jika Anda mendapatkan *f*A = 1,8 sedangkan *f*R= 1,5, maka Akurasi = 80%. Bagaimanapun, Anda mungkin saja mendapatkan Akurasi lebih dari 100%. Misalnya, jika Anda mendapatkan *f*A = 1,2, maka Akurasi = 120%;

* Jika Anda mendapatkan Akurasi > 100%, maka Akurasi\_Model = 100;

1. **Rancangan metode yang digunakan dan nilai minimum yang dihasilkan**

Simulated Anneling (SA) memanfaatkan analogi pendingan dan pembekuan metal menjadi sebuah struktur Kristal dengan energi yang minimal. Tidak seperti pendekatan HC(Hill Climbing), dengan probabilitas tertentu SA mungkin bias keluar dari *jebakan local minimum.* SA menggunakan sebuah rumus probabilitas yang memungkinkannya bias keluar dari *local minimum.* Ketika new state tiddak lebih baik dari current state, maka new state tersebut masih mungkin dipilih dengan probabilitas sebagai berikut:

*P*(∆E)=

Algoritma Simulated Annealing

1. Evaluasi initial state. Jika state ini adalah goal state, maka kembalikan state ini sebagai solusi dan keluar dari program. Jika bukan, lanjutkan dengan initial state sebagai current state.
2. Inisialisasi BEST-SO-FAR dengan current state
3. Inisialisasi T sesuai dengan annealing schedule
4. Ulangi sampai solusi ditemukan atau tidak ada lagi aturan produksi yang bias diaplikasikan ke current state:
5. Pilih sebuah aturan produksi yang belum pernah diaplikasikan ke current state dan aplikasikan aturan produksi tersebut untuk menghasilkan new state.
6. Evaluasi new state. Hitung:

∆E =

* Jika new state adalah goal, maka kembalikan state ini sebagai solusi dan kkeluar dari program.
* Jika new state bukan goal tetapi lebih baik dari pada current state(∆E>0), maka set current state ke new state. Juga set BEST-SO-FAR ke new state.
* Jika new state tidak lebih baik dari pada current state (∆e≤0), maka set current state ke new state dengan probabilitas *p* seperti didefinisikan oleh persamaan.
* Langkah ini biasanya diimplementasikan dengan membangkitkan sebuah bilangan acak dalam interval [0,1]. Jika bilangan tersebut kurang dari *p* ,maka set current state ke new state. Jika bilangan tersebut lebih besar dari pada *p*, maka jangan mengerjakan apapun.

1. Jika diperlukan, revisi nilai T berdasarkan *annealing schedule*
2. Kembalikan *BEST-SO-FAR* sebagai solusi.

Referensi: Buku Artifical Intelligence,suyanto,2014

1. **Screenshot yang menunjukkan output program ketika dijalankan**

