

Metode Simulated Annealing

Oleh:

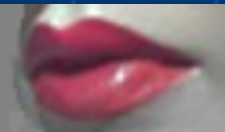
Tri Budi Santoso (Signal Processing Lab)

Achmad Basuki (Image Processing Lab)

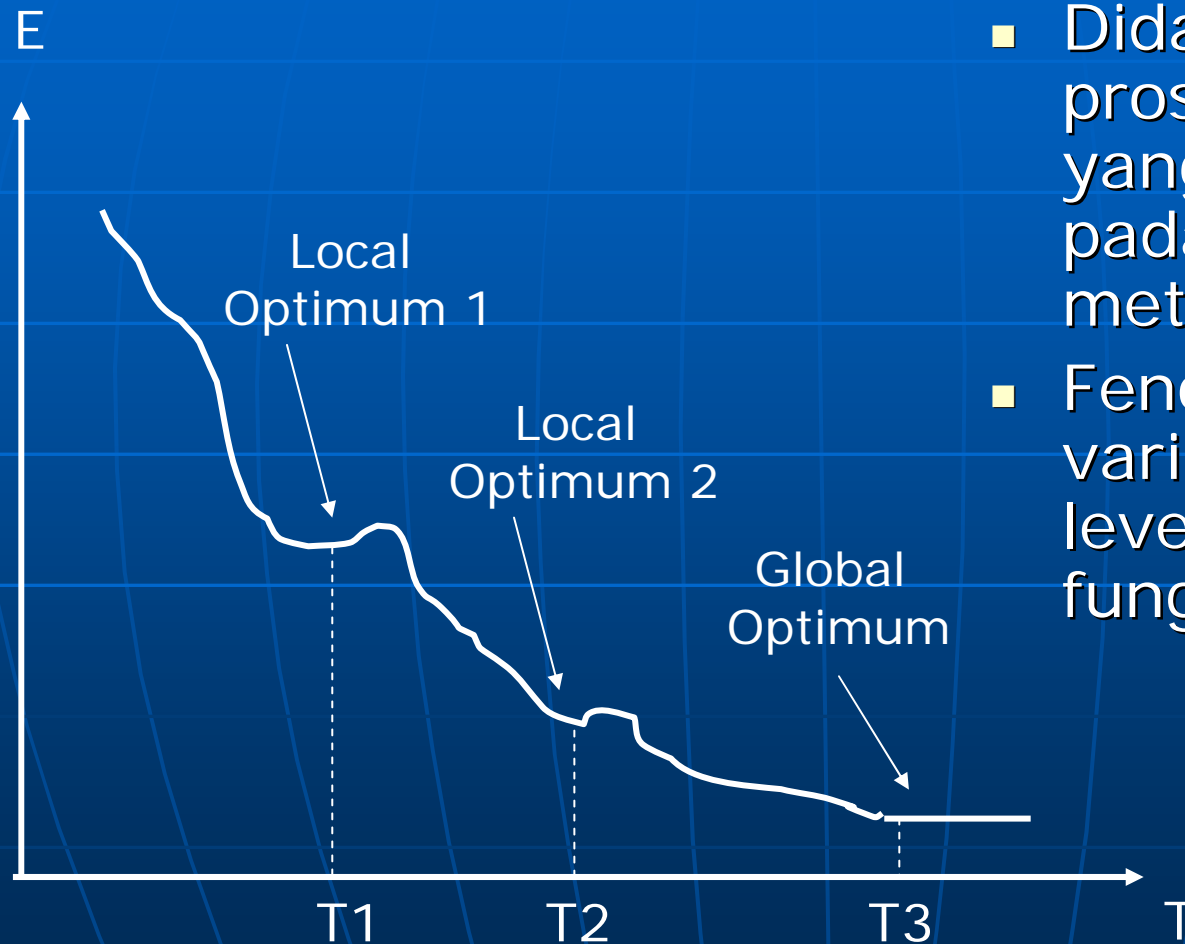
Miftahul Huda (Signal Processing Lab)

EEPIS-ITS

Speech Signal Processing



Fenomena Annealing



- Didasari pada logika proses annealing yang umum dipakai pada teknologi metalurgi
- Fenomena adanya variasi perubahan level energi sebagai fungsi temperatur

Konsep Berfikir Simulated Annealing

- Dalam grafik terlihat penurunan energi mengalami fenomena sedikit aneh. Pada saat temperatur T_1 , ada kecenderungan energi tidak mau turun. Tetapi dengan sedikit pergeseran dari nilai T_1 penurunan energi terjadi sampai dengan nilai T_2 , hal ini terjadi juga pada kasus T_2 , dst sampai terjadi suatu kondisi dimana nilai energi tidak lagi mengalami penurunan biarpun nilai T terus bertambah.
- Kondisi E pada T_1 dianalogikan sebagai titik local optimum 1, sedangkan pada saat T_2 dianalogikan sebagai local optimum 2.
- Konsep Optimasi dijalankan dengan cara merubah (menaikkan) nilai T untuk mendapatkan suatu kondisi dimana E menurun, dan dilakukan terus menerus sampai suatu kondisi dimana perubahan T tidak berpengaruh pada nilai E , atau dikatakan suatu global optimum sudah dicapai.

Logika Matematika Annealing

Energi pada suatu state = E_k

Energi pada state berikutnya = E_{k+1}

Keduanya memiliki hubungan berdasar proses annealing sebagai:

$$\begin{aligned}h(x) &= \frac{\exp(-E_{k+1}/T)}{\exp(-E_{k+1}/T) + \exp(-E_k/T)} \\ &= \frac{1}{1 + \exp(\Delta E_k/T)} \\ &\approx \exp(-\Delta E_k/T)\end{aligned}$$

Algorithma

- Mengacu pada proses Monte Carlo untuk mencari nilai π , simulated annealing bisa dilakukan dengan step berikut:
 1. Buat inisialisasi $m = 1$, $N = 1000$
 2. Bangkitkan nilai x secara random yang memiliki nilai $0 \sim 1$
 3. Bangkitkan nilai y secara random yang memiliki nilai $0 \sim 1$
 4. Hitung $r_2 = x^2 + y^2$
 5. Jika nilai $r < 1$, maka $m = m + 1$,
Jika tidak,
Bangkitkan suatu nilai random $a(0 \sim 1)$.
Tetapkan nilai $p = \exp(-\Delta E/T)$.
Jika $a \geq p$, maka $m = m + 1$
Jika tidak, maka $m = m$
 6. Lakukan proses looping ini sampai 1000 kali.
 7. Setelah proses looping selesai, hitung nilai $\pi = 4 * m/N$.

```

#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#define N 1001
int nn=N-1;
double pi,E_T=0.5;
void anneling(){
double a,p,R,x,y,m=0.0;
for (int i=1;i<=nn;i++) {
    x=(double)rand()/(RAND_MAX);
    y=(double)rand()/(RAND_MAX);
    R = x*x + y*y;
    if(R<1.0)      {
        m=m+1.0;
    }
    else{
        a=(double)rand()/RAND_MAX;
        p=exp(-E_T);
        if(a>p)
            m=m+1.;
        else
            m=m;
    }
}
pi=4*m/(nn);
printf("\nSimulated Annealing menghasilkan pi= %f",pi);
}

```

```

void main()
{
    anneling();
}

```