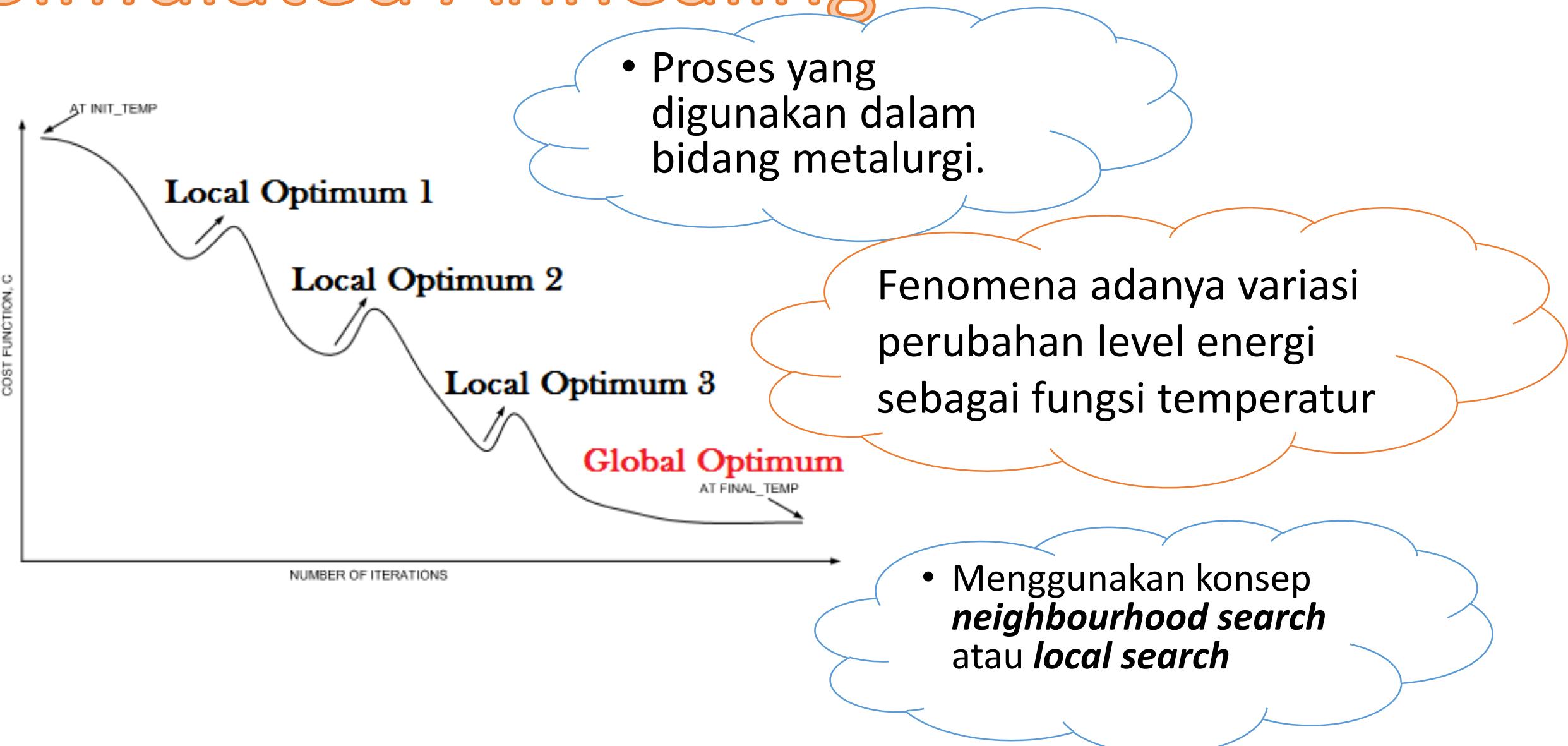


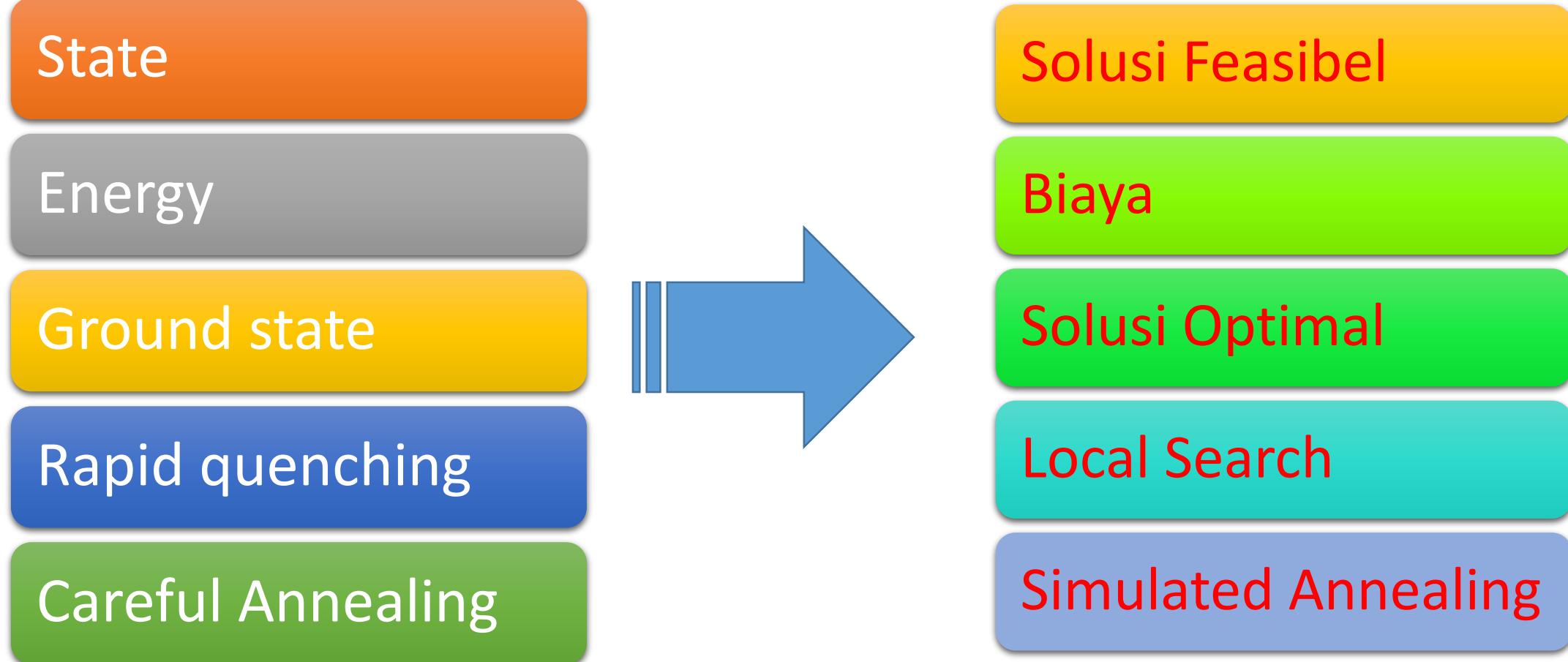


SIMULATED ANNEALING

Simulated Annealing



Analogi Fisis - SA

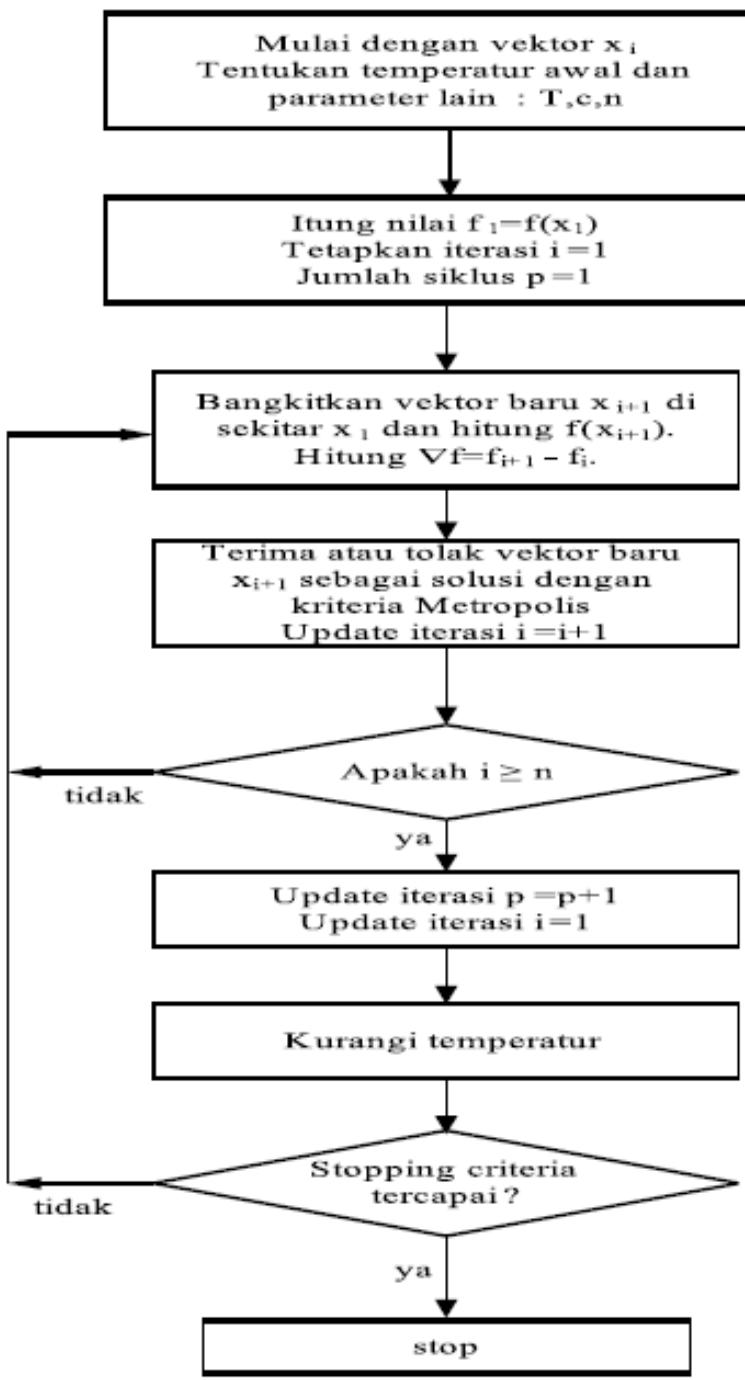


Simulated Annealing

Memperbaharui Solusi

Mengevaluasi kualitas solusi

Menerima solusi baru jika lebih baik dari sebelumnya



Algoritma *simulated annealing*

Input: $T_0, c, itmax$

Output: x_{opt}, f_{best}

Bangkitkan x_0

$f_{best} = f(x_0)$

$T = T_0$

while kriteria penghentian belum dicapai do

 bangkitkan Δx berdistribusi Gaussian

 if $(f(x_0 + \Delta x) < f(x_0))$ then

$f_{best} = f(x_0 + \Delta x)$

$x_0 = x_0 + \Delta x$

 else

$\Delta x = f(x_0 + \Delta x) - f(x_0)$

 Bangkitkan bilangan random r

 if $r < \exp(-\Delta x * \beta/T)$ then

$x_0 = x_0 + \Delta x$

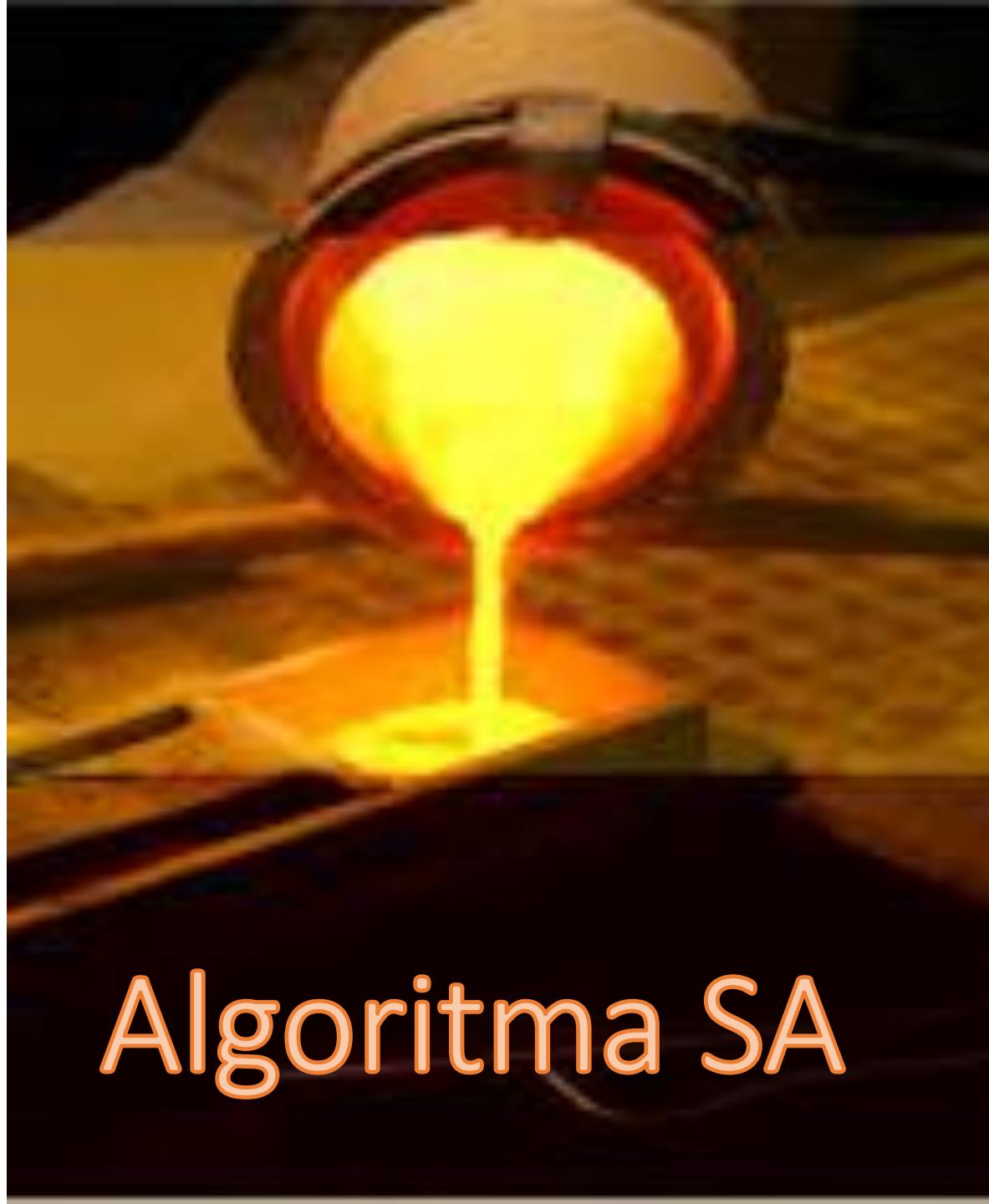
 titik dasar baru diturunkan dari uphill move

 end

 end

$T = T^* \text{faktor reduksi}$

end



Algoritma SA

Contoh minimasi fungsi:

$$f(x) = 500 - 20x_1 - 26x_2 - 4x_1x_2 + 4x_1^2 + 3x_2^2$$

1. Langkah 1, Tetapkan nilai inisial untuk T dengan cara mengambil 4 nilai random untuk x , lalu ambil rata-rata dari 4 nilai $f(x)$ yang dihasilkan. Misalkan

$$x^{(1)} = [2, 0], \quad x^{(2)} = [5, 10], \quad x^{(3)} = [8, 5], \quad x^{(4)} = [10, 10]$$

dengan nilai f yang berhubungan adalah

$$f^{(1)} = 476, \quad f^{(2)} = 340, \quad f^{(3)} = 381, \quad f^{(4)} = 340.$$

Nilai rata-rata dari keempat nilai f ini adalah 384.25. Sehingga nilai awal $T = 384.25$. Faktor reduksi temperatur dipilih $c = 0.5$. Agar proses komputasinya pendek kita pilih jumlah iterasi untuk tiap nilai T , $n = 2$. Pilih nilai x awal $x^{(1)} = [4, 5]$.

2. Langkah 2, hitung nilai fungsi $f^{(1)} = 349$, dan tetapkan iterasi $i = 1$

3. Langkah 3, Temukan titik baru dengan cara mencari dalam selang tertentu dari nilai sekarang untuk setiap variabel. Di sini kita gunakan selang ± 6 dari nilai sekarang dikalikan dengan bilangan random. Selang untuk $x_1 = 4 \pm 6$ dan selang untuk $x_2 = 5 \pm 6$. Misalkan bilangan random $u_1 = 0.31$ dan $u_2 = 0.57$.

$$r_1 = -2 + u_1[10 - (-2)] = -2 + 0.31(12) = 1.72$$

$$r_2 = -1 + u_2[11 - (-1)] = -1 + 0.57(12) = 5.84$$

$x^{(2)} = [r_1, r_2] = [1.72, 5.84]$. Karena nilai fungsi tujuan $f^{(2)} = f(x^{(2)}) = 387.7312$, nilai Δf didapatkan sebagai

$$\Delta f = f^{(2)} - f^{(1)} = 387.7312 - 349.0 = 38.7312$$

$x^{(2)} = [r_1, r_2] = [1.72, 5.84]$. Karena nilai fungsi tujuan $f^{(2)} = f(x^{(2)}) = 387.7312$, nilai Δf didapatkan sebagai

$$\Delta f = f^{(2)} - f^{(1)} = 387.7312 - 349.0 = 38.7312$$

Karena Δf positif, maka kita akan gunakan kriteria Metropolis untuk menentukan menerima atau menolak titik yang baru. Untuk itu perlu dibangkitkan bilangan random $r = 0.83$. Misalkan $k = 1$, maka

$$P(x^{(2)}) = e^{-\Delta E/kT} = e^{-38.73/384.25} = 0.9041$$

Karena $0.83 \leq 0.9041$ maka nilai $x^{(2)}$ kita terima. Walaupun $f^{(2)}$ lebih besar dari $f^{(1)}$ titik $x^{(2)}$ kita terima karena ini masih dalam tahap awal dimana temperatur masih tinggi.

4. Langkah 4, masuk iterasi $i = 2$,
5. Bangkitkan nilai x baru berdasar nilai $x^{(2)} \pm 6$ dengan bilangan random $u_1 = 0.92$ dan $u_2 = 0.73$

$$r_1 = -4.28 + u_1[7.72 - (-4.28)] = -4.28 + 0.92(12) = 6.76$$

$$r_2 = -0.16 + u_2[11.84 - (-0.16)] = -0.16 + 0.73(12) = 8.60$$

Sehingga nilai $x^{(3)} = [6.76, 8.60]$ dan nilai $f^3 = 313.3264$. Kita lihat bahwa nilai f^3 lebih kecil dari f^2 dengan $\Delta f = f^{(3)} - f^{(2)} = 313.3264 - 387.7312 = -74.4048$.

6. Langkah 4, Karena nilai $\Delta f < 0$, maka kita terima $x^{(3)}$ sebagai solusi baru. Kita masuk iterasi baru i=3. Kita ke langkah 5
7. Langkah 5, karena $i > n$ maka satu siklus iterasi untuk nilai T yang sekarang sudah selesai ($n = 2$), maka nilai T perlu direduksi menjadi $T = 0.5(384.25) = 192.125$

Kode matlab

```
function [x,fx] = sa(f,maxit,xL,xU)

% sa minimasi fungsi dengan simulated annealing
% f adalah nama fungsi yang akan dicari minimumnya
% maxit - maksimum jumlah iterasi
% xL- batas bawah nilai x
% xU- batas atas nilai x
%
N=10;
n=length(xL);
x=zeros(N,n);
```

Kode matlab

```
fx=zeros(N,1);
for i=1:N
    x(i,:)=xL+(xU-xL).*rand;
    fx(i)=feval(f,x(i,:));
end
init=mean(fx);%temperatur awal
parent=xL+(xU-xL).*rand;%nilai inisial x
fp=feval(f,parent);
k = 1; % boltzmann constant
c=0.9; %faktor reduksi
it=1;
deltaf=10;
```

```
while abs(deltaf)>1e-8
    while it<maxit
        newsol = (parent +(randperm(n))==n)*randn/100;
        fnew=feval(f ,newsol);
        deltaf=fnew-fp;
        if deltaf<0
            parent=newsol;
            fp=feval(f ,parent);
        else %gunakan kriteria metropolis
            if rand < exp(-deltaf/(k*init))
                %jika nilai rand < dari probabilitas
                parent=newsol;
                fp=feval(f ,parent);
            end
        end
        it=it+1;
    end
```

Kode matlab

```
init=c*init;
    it=1;
if init<1e-8
    break
end
end
x=parent;
fx=feval(f,x);
```



Thank You
Thank You
Thank You!!!!