ME 620: Fundamentals of Artificial Intelligence

Lecture 9: Informed Search Strategies - III

Shyamanta M Hazarika
Biomimetic Robotics and Artificial Intelligence Lab
Mechanical Engineering and M F School of Data Sc. & AI
IIT Guwahati

Simulated Annealing

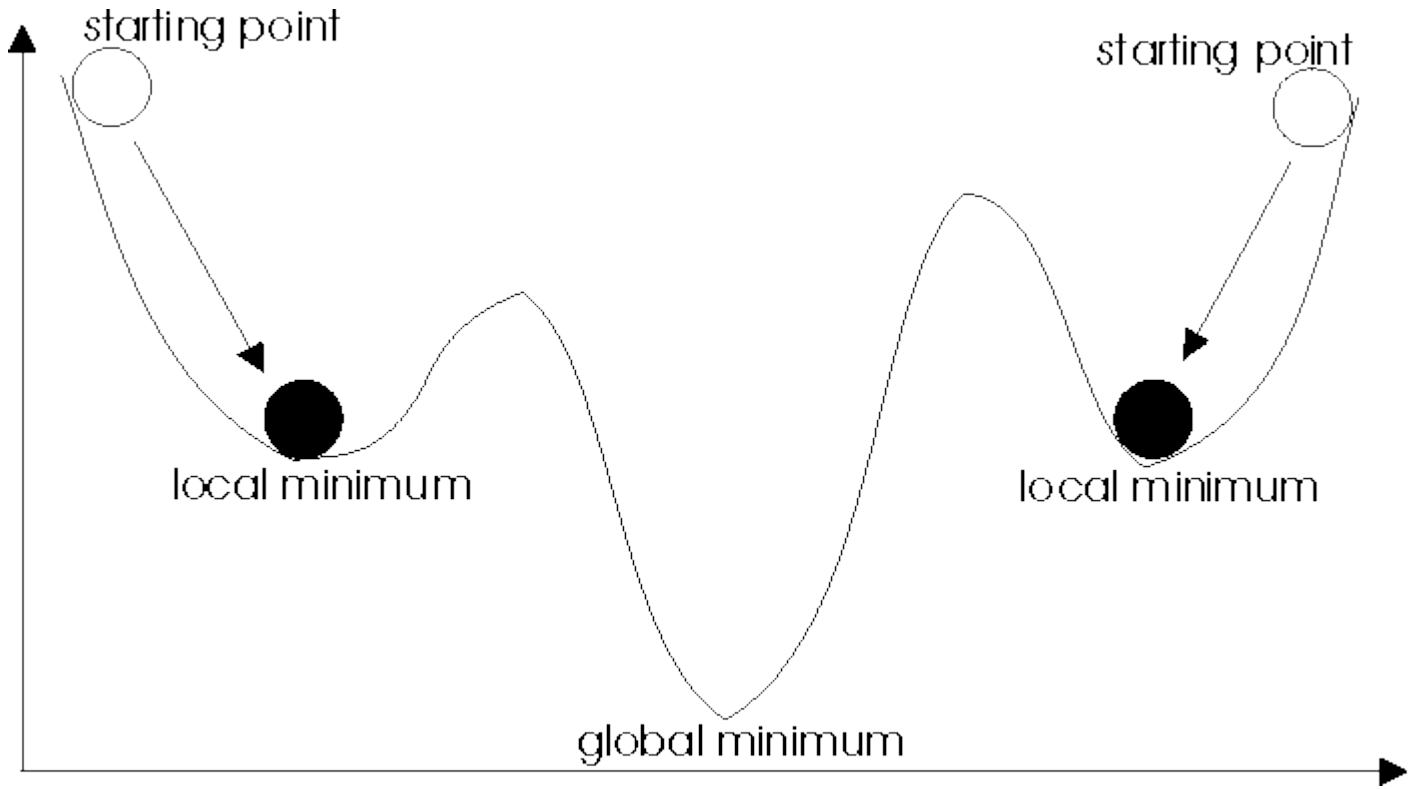
- Hill-climbing algorithm that never makes a downhill move is guaranteed to be incomplete, because it can get stuck in a local maximum.
- In contrast, a purely random walk - that is, moving to a successor chosen uniformly at random from the set of successors is complete but extremely inefficient.
- Combine hill-climbing with a random walk in some way to get both efficiency and completeness. Simulated annealing is one such algorithm.

Simulated Annealing

- It is a stochastic hill-climbing algorithm:
  - A successor is selected among all possible successors according to a probability distribution.
  - The successor can be worse than the current state.
- Random steps are taken in the state space.
- It is inspired by the physical process of controlled cooling (crystallization, metal annealing):
  - A metal is heated up to a high temperature and then is progressively cooled in a controlled way.
  - If the cooling is adequate, the minimum-energy structure (a global minimum) is obtained.

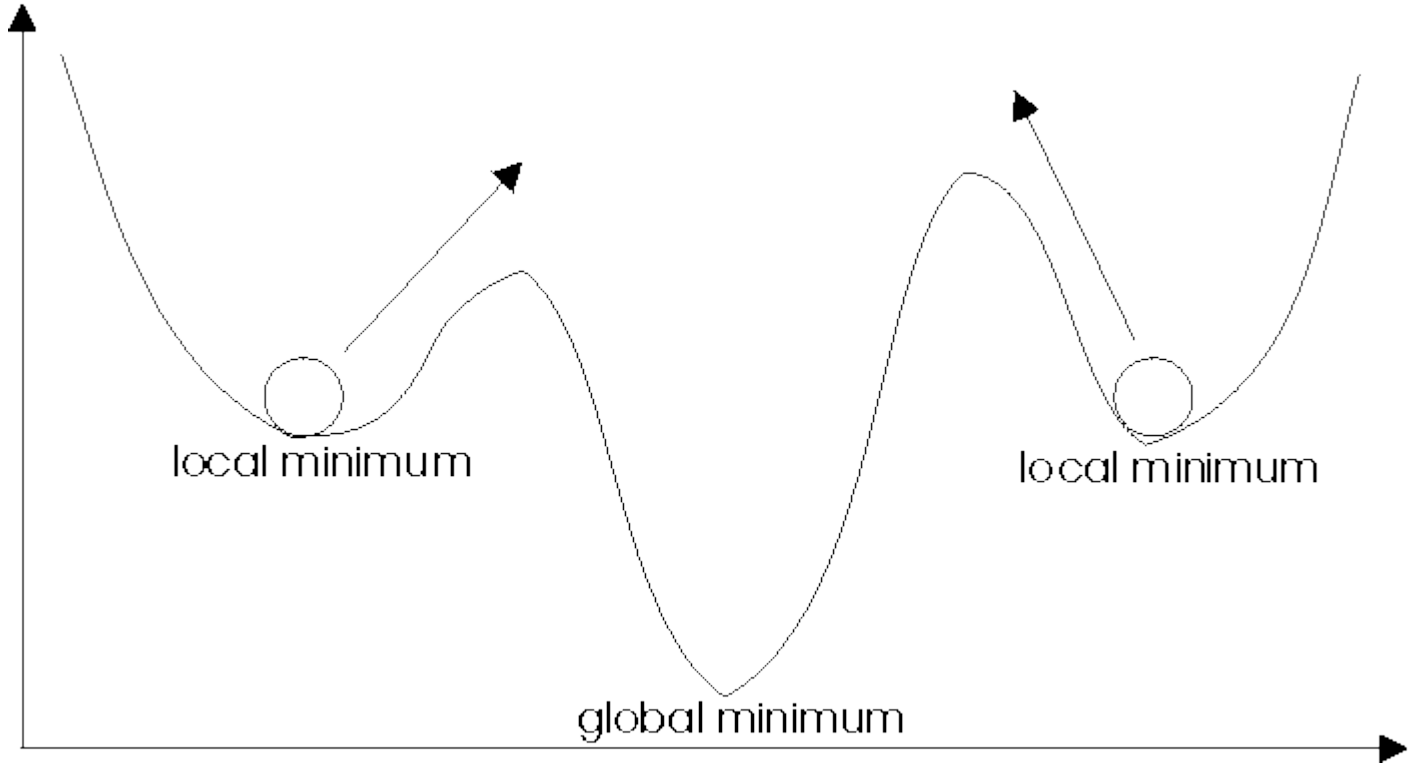
Simulated Annealing

Aim: to avoid local optima, which represent a problem in hill climbing.



Simulated Annealing

Solution: to take, **occasionally, steps in a different direction** from the one in which the increase (or decrease) of energy is maximum.



Local search (or Monte Carlo) Algorithms

- Before describing the simulated annealing algorithm for optimization, we need to introduce the principles of **local search optimization algorithms**, of which simulated annealing is an extension.

- These algorithms **optimize the cost function by exploring the neighborhood** of the current point in the solution space.

Local search (or Monte Carlo) Algorithms

Local search

1. **Draw an initial solution i;**
2. **Generate a solution j from the neighborhood S_i of the current solution i;**
3. **If $f(j) < f(i)$ then j becomes the current solution;**
4. **If $f(j) \geq f(i)$ for all $j \in S_i$ then END;**
5. **Go to step 2;**

Metropolis Algorithm

Starting from an initial state i of energy E_i, a new state j of energy E_j is generated by modifying the position of one particle.

If the energy difference, $E_i - E_j$, is positive (the new state features lower energy), the state j becomes the new current state. If the energy difference is less than or equal to zero, then the probability that the state j becomes the current state is given by:

$$Pr\{\text{Current state} = j\} = e^{\left(\frac{E_i - E_j}{k_b . T}\right)}$$

where T represents the temperature of the solid and k_B is the Boltzmann constant ($k_B = 1.38 \times 10^{-23}$ joule/Kelvin).

Simulated Annealing

- The state-space points represent the possible states of the solid;
- The function to be minimized represents the energy of the solid.

A control parameter c, acting as a temperature, is then introduced. This parameter is expressed with the same units as the objective that is optimized.

It is also assumed that the user provides for each point of the state space, a neighborhood and a mechanism for generating a solution in this neighborhood. We then define the acceptance principle:

$$Pr\{\ accept\ j\} = \begin{cases} 1 & if\ f(j) < f(i) \\ e^{\left(\frac{f(i)-f(j)}{c}\right)} & else. \end{cases}$$

Simulated Annealing

1. **Initialization** $i := i_{start}, k := 0, c_k = c_0, L_k := L_0$);
2. **Repeat**
3. **For** $l = 0$ **to** L_k **do**
   - **Generate a solution** j **from the neighborhood** S_i **of the current solution** i**;**
   - **If** $f(j) < f(i)$ **then** $i := j$ (j **becomes the current solution**);
   - **Else,** j **becomes the current solution with probability** $e^{\left(\frac{f(i)-f(j)}{c_k}\right)}$;
4. $k := k + 1$;
5. **Compute**(L_k, c_k);
6. **Until** $c_k \simeq 0$

Simulated Annealing

- Best way to select an annealing schedule
  - Try several schedules and observing the effect on both the quality of the solution that is found and the rate at which the process converges.
- It is important to realize that
  - As T approaches zero, the probability of accepting a move to worse goes to zero and simulated annealing becomes identical to simple hill climbing.
  - What matters in computing the probability of accepting a move is the ratio $\frac{\Delta E}{T}$. Thus it is important that value of T is scaled so that this ratio is meaningful.

Simulated Annealing

- It is suitable for problems in which the global optimum is surrounded by many local optima.
- It is suitable for problems in which it is difficult to find a good heuristic function.
- Determining the values of the parameters can be a quite difficult and often requires experimentation.