Simulated Annealing

Drawbacks of Hill Climbing

- Locally Optimal Solution
  - Solution: Simulated Annealing
  - “Greedy” optimization methods
    - Can get trapped at local minima
    - Outcome might depend on starting point

- Most minimization strategies find the nearest local minimum
  - Standard strategy
    - Generate trial point based on current estimates
    - Evaluate function at proposed location
    - Accept new value if it improves solution

Simulated Annealing

- It is an algorithm which helps us reach a global optimum of a multi-dimensional function despite the presence of several hills and valleys
- It is evolutionary in the sense that unlike traditional optimization techniques like random walk or hill climbing it will not get stuck at a local maximum
- Imagine you are standing in a place and you need to reach the highest point. you can look around you and if you see a place with a higher height than the ground you are standing, you move to that place, keep repeating this and you eventually reach a point where you cannot see a higher point. This technique is called hill climbing

Simulated Annealing

- Problem: You might not be standing at the highest point, and you might not be able to see a higher point from where you are standing. you might have been able to observe a higher point if you where standing at some other point
- Simulated annealing takes the risk of occasionally moving to a lower point so that the chances of missing out on seeing a higher point are reduced
- Initially it is willing to take risks of jumping to lower points, however as time passes, it is less willing to jump to lower point and eventually terminates
- This idea lower the chances of getting caught at local maxima, plateau or ridge

Annealing

- One manner in which crystals are formed
- Gradual cooling of liquid …
  - At high temperatures, molecules move freely
  - At low temperatures, molecules are "stuck"
- If cooling is slow
  - Low energy, organized crystal lattice formed

Simulated Annealing

- Analogy with thermodynamics
- Incorporate a temperature parameter into the minimization procedure
- At high temperatures, explore parameter space
- At lower temperatures, restrict exploration

Relationship between Physical Annealing and Simulated Annealing

Thermodynamic Simulation	**Combinatorial Optimisation**
System States	Feasible Solutions
Energy	Cost
Change of State	Neighbouring Solutions
Temperature	Control Parameter
Frozen State	Heuristic Solution

Simulated Annealing

- Change in terminology:

  heuristic function→ objective function

  Goal: Minimize objective function value

- Process of valley descending rather than hill climbing

Annealing

- Physical substances such as metals are melted (raised to high temperature-energy levels) and then gradually cooled until some solid state is reached

  Goal: produce minimal energy state

- Physical substances usually move from higher energy configurations to lower ones, so the valley descending occurs naturally

Simulated Annealing

- But, there is some probability that a transition to a higher energy state will occur

$$P = e^{-(-\delta E / kt)}$$

Where,

δE: Change in energy level

T : Temperature

k : Boltzmann's constant

- Probability that an uphill move will be made, decreases as the temp decreases

Simulated Annealing

T: 10,000	99.99990%
1000	99.9905%
100	99.99000%
10	90.04%
5	81.87%
3	71.89%
2	60.65%
1	36.78%

- Moves are more likely during the beginning of the process when the temperature is high and they become less likely at the end, as the temperature becomes less

  Downhill moves→ anytime

  Upward moves→ during initial time

Simulated Annealing

- **Annealing Schedule:**
  - **Rate at which the system is cooled**
- **Metal heated to a sufficiently high temperature and all we want to do is cool the metal down**
- **Cooling can be done by two approaches**
  - **Hastily approach**
  - **Gradual approach**
- **These properties of physical annealing can be used to define an analogous process of simulated annealing**

- **In analogous process…..**
- **δE is the change in the value of objective function**
- **k describes the correspondence between the units of temperature & unit of energy E & T are artificial, we drop 'k'**

$$p' = e^{-(-\delta E / kt)}$$

- **Need to choose a schedule of value T**

- In actual physical annealing process, the temperature is well defined notion measured in standard units. The variable k describes the correspondence between the unit of temperature and unit of energy. Here in this optimization theory we may omit k and combine it in T itself.

- **How Simulated Annealing is different than Hill Climbing?**
  - **The annealing schedule must be maintained**
  - **Move to worst/bad/poor states may be accepted**
  - **Maintains state ‘best state found so far’**

In order to select an annealing schedule

- Initial value of T
- When to reduce the temperature of the system (After how many iterations)
- How much amount the temperature should be reduced
- The end point of T.

Algorithm

- Algorithm:

1. Evaluate the initial state: If it is also a goal state, return it and quit. Otherwise continue with initial state as current state

2. Initialize BEST-SO-FAR to the current state

3. Initialize ‘T’ according to annealing schedule

4. Loop until a solution is found or until there are no new operators left to be applied in the current state

a) Select an operator that has not yet been applied to the current state and apply it to produce a new state

b) Evaluate the new state

δE = (Value of current) – (Value of new state)

Cont.

- If the new state is goal state, return it & quit

  - else if it is better than current state, make it the current state also set BEST-SO-FAR to this new state

  - If it is not better than current state, then make it the current state with probability p'. This step is usually implemented by invoking a random number generator to produce a random number in the range of [0,1]. If the number is less than p', then the move is accepted otherwise rejected. Otherwise, do nothing.

c) Revise T as necessary according to annealing schedule

5. Return BEST-SO-FAR as the answer

Metropolis Algorithm

- In 1958 Metropolis et al. introduced a simple algorithm for simulating the evolution of a solid in a heat bath to **thermal equilibrium**.
- Their algorithm is based on Monte Carlo techniques, and generates a sequence of states of the solid in the following way.
- Given a current state i of the solid with energy E_i, a subsequent state j is generated by applying a **perturbation mechanism** that transforms the current state into a next state by a small distortion, for instance, by a displacement of a single particle.

Metropolis Algorithm

- The energy of the next state is E_j. (Ej)
- If the energy difference, $E_j - E_i$, is less than or equal to 0, the state j is accepted as the current state.
- If the energy difference is greater than 0, then state j is accepted with a probability given by

$$\exp\left(\frac{E_i - E_j}{k_B T}\right)$$

  - where T denotes the temperature of the heat bath and
  - k_B a physical constant known as the Boltzmann constant.

(c) The following table shows six evaluations of a simulated annealing algorithm. For each evaluation give the probability of the next state being accepted. Assume the objective function is being maximised.

No.	Current State (Evaluation)	Potential New State (Evaluation)	Temperature
1	120	50	20
2	120	50	500
3	120	100	20
4	120	100	500
5	120	150	20
6	120	150	500

No.	Current State (Evaluation)	Potential New State (Evaluation)	Temperature	Probability of Acceptance
1	120	50	20	0.030197383
2	120	50	500	0.869358235
3	120	100	20	0.367879441
4	120	100	500	0.960789439
7	120	150	20	4.48168907
8	120	150	500	1.061836547

- In this part of the discussion:
- If the proposed solution is an improved solution then it is given a probability greater than
- 1. Therefore, it will always be accepted.
- See rows 5 & 6.
- When the temperature is zero, we have a hill climbing algorithm (i.e. only better solutions are accepted).
- It is usual to check if the proposed solution is better than the current solution and accept it anyway. Although this achieves the same result it does save having to execute the computationally expensive exp function.
- By looking at the rows where the change in the evaluation is the same but the temperature is different (for example, compare row 1 and 2) it shows that when the temperature higher the probability of the worse solution being accepted is higher, even when the change in the evaluation function is the same.

Simulated Annealing, e.g.

Let us maximize a continuous function:

$$f(x) = x^3 - 60x^2 + 900x + 100$$

- A solution x is represented as a string of 5 bits.
- The neighborhood consists in flipping randomly a bit.
- The global maximum of this function is 01010 ($x = 10$, $f(x) = 4100$).

For an initial solution of 10011 ($f(19) = 2399$)

1. $p = e^{(-112/500)} = 0.80$
2. $p = e^{(-247/405)} = 0.54$
3. $p = e^{(-16/295.2)} = 0.95$
4. ...

$T = 500$ and Initial Solution (10011)

T	Move	Solution	f	Δf	Move?	New Neighbor Solution
500	1	00011	2287	112	Yes	00011
450	3	00111	3803	<0	Yes	00111
405	5	00110	3556	247	Yes	00110
364.5	2	01110	3684	<0	Yes	01110
328	4	01100	3998	<0	Yes	01100
295.2	3	01000	3972	16	Yes	01000
265.7	4	01010	**4100**	<0	Yes	01010
239.1	5	01011	4071	29	Yes	01011
215.2	1	11011	343	3728	No	01011

The initial temperature is not high enough and the algorithm gets stuck by local optima.

T = 100 and Initial Solution (10011). When Temperature is not High Enough, Algorithm Gets Stuck

T	Move	Solution	f	Δf	Move?	New Neighbor Solution
100	1	00011	2287	112	No	10011
90	3	10111	1227	1172	No	10011
81	5	10010	2692	<0	Yes	10010
72.9	2	11010	516	2176	No	10010
65.6	4	10000	**3236**	<0	Yes	10000
59	3	10100	2100	1136	Yes	10000

Drawbacks of Simulated Annealing

- Can take a long time to run if the annealing schedule is very long.
- There is a lot of tuneable parameter in this algorithm.

Application

- TSP