Simulated Annealing

PhD. Msc. David C. Baldears S.

Intuitions

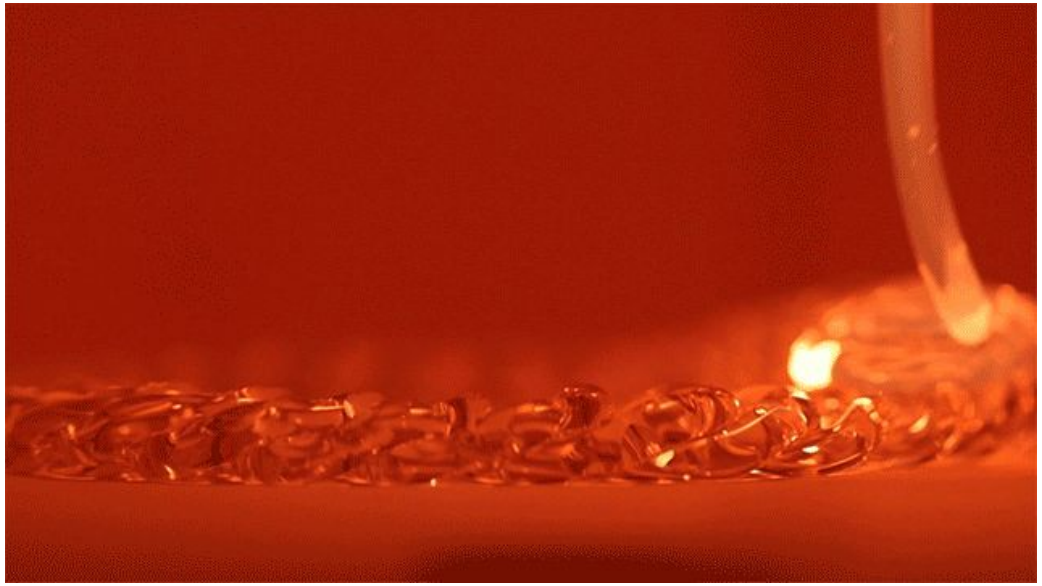
- Hill-climbing is incomplete
- Pure random walk, keeping track of the best state found so far, is complete but very inefficient

Combine the ideas: add some randomness to hill-climbing to allow the possibility of escape from a local optimum

Industrial Annealing

Annealing: harden metals and glass by heating them to a high temperature and then gradually cooling them

At the start, make lots of moves and then gradually slow down

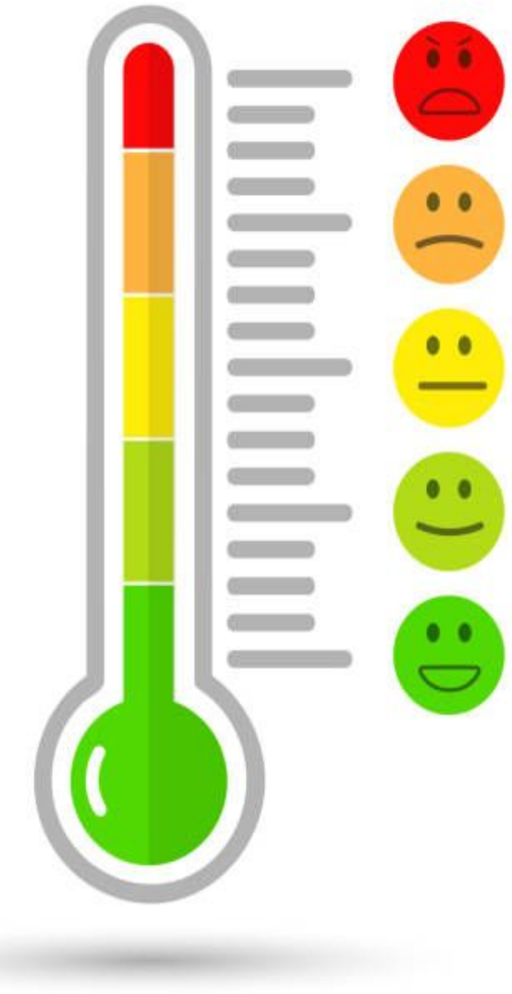


Here you can see the effect of not annealing your beads. By using polarising filters to show the stress. The bead on the right has not been annealed, you can see the stress in the bead as the blue and orange coloured parts, the bead on the left **has** been annealed and shows no coloured sections (note that the dark areas are just shadows and reflections)

Simulated Annealing

Based on a metallurgical metaphor

- Start with a temperature set very high and slowly reduce it.
  - Increase the temperature of the heat bath to a value where the metal melts
  - Decrease the temperature of the heat bath carefully while the particles arrange themselves in orderly way, until it reaches a minimum state
- Run hill climbing with the twist that you can occasionally replace the current state with a worse state based on the current temperature and how much worse the new state is.



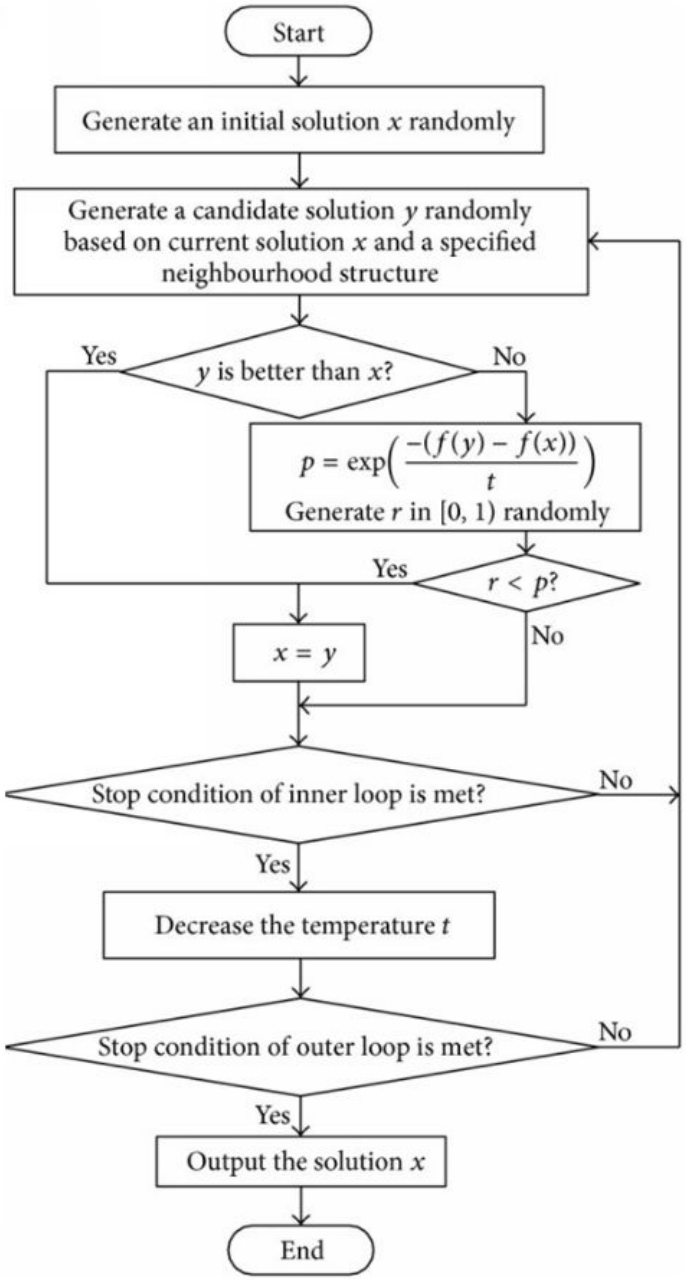
Control of Annealing Process

Acceptance of a search step (Metropolis Criterion):

- Assume the performance change in the search direction is Δ.
- Always accept a descending step, i.e. Δ ≤ 0.
- Accept a ascending step only if it passes a random test,

$$e^{(-\Delta/T)} > \text{random}(0,1)$$

- Stopping Criterion:
    - A given minimum value of temperature has been reached
    - A certain number of iterations has passed without acceptance of a new solution
    - A specified number of total iterations has been executed



Simulated Annealing - Theory

- Decrease the temperature slowly, accepting less bad moves at each temperature level until at very low temperatures the algorithm becomes a greedy hill-climbing algorithm.
  - Probability of a move decreases with the amount ΔE by which the evaluation is worsened
    - The distribution used to decide if we accept a bad movement is known as Boltzmann distribution.
    - This distribution is very well known is in solid physics and plays a central role in simulated annealing. Where γ is the current configuration of the system, E γ is the energy related with it, and Z is a normalization constant.

$$P(\gamma) = \frac{e^{-E_\gamma/T}}{Z(T)},$$

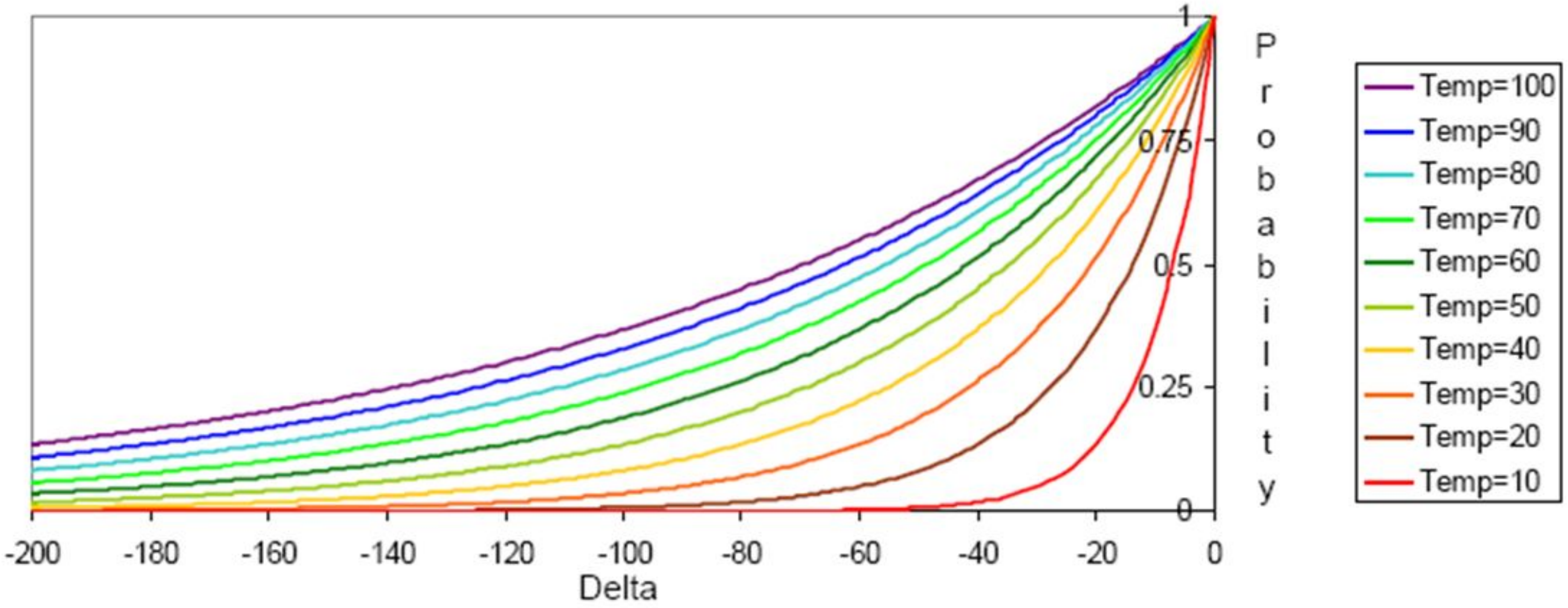
$$Z(T) = \sum_{\gamma'} e^{-E_{\gamma'}/T},$$

  - A second parameter T is also used to determine the probability: high T allows more worse moves, T close to zero results in few or no bad moves
  - Schedule input determines the value of T as a function of the completed cycles

Acceptance criterion and cooling schedule

if (delta>=0) accept

else if ($random < e^{delta/Temp}$) accept, else reject /* 0<=random<=1 */



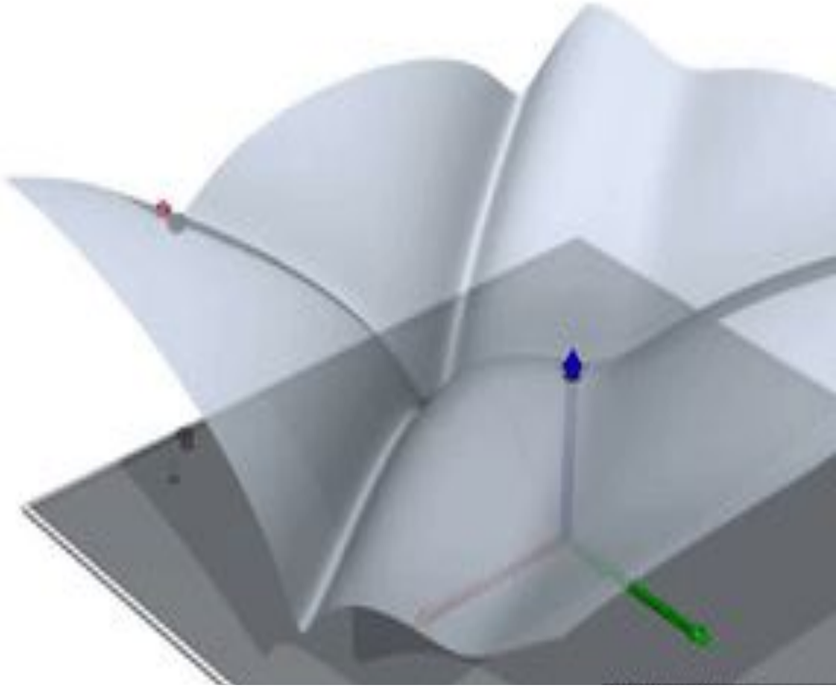
Initially temperature is very high (most bad moves accepted)

Temp slowly goes to 0, with multiple moves attempted at each temperature

Final runs with temp=0 (always reject bad moves) greedily “quench” the system

Layman's terms

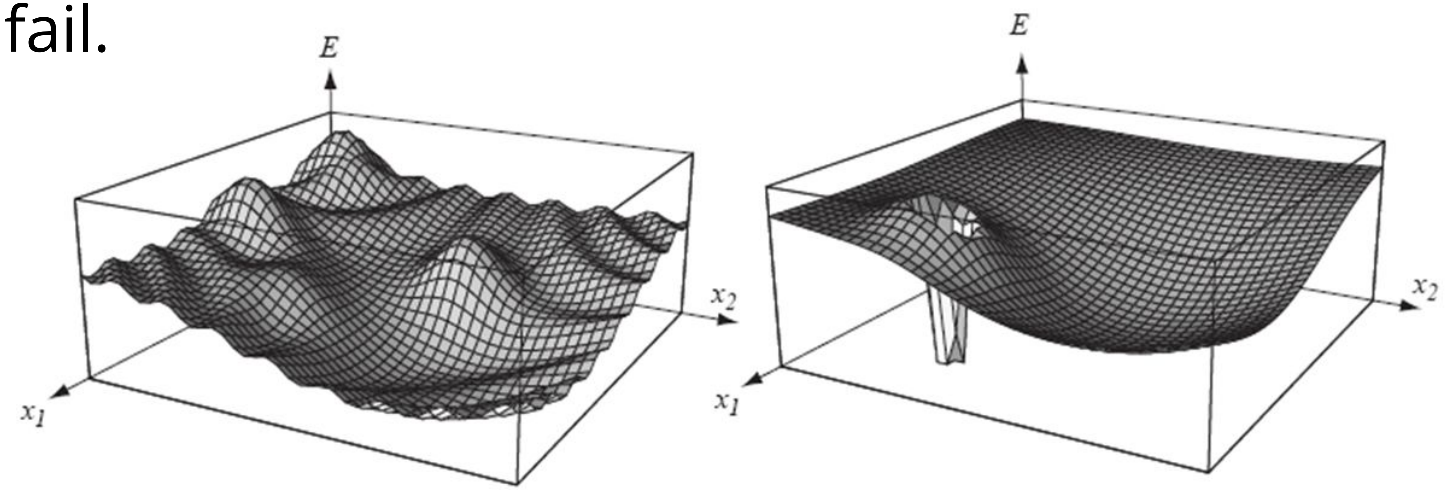
- Generate a random new neighbor from current state.
- If it's better take it.
- If it's worse then take it with some probability proportional to the temperature and the delta between the new and old states.
- The algorithm wanders around during the early parts of the search, hopefully toward a good general region of the state space
- Toward the end, the algorithm does a more focused search, making few bad moves



High vs Low Temperatures

Practical Issues with simulated annealing

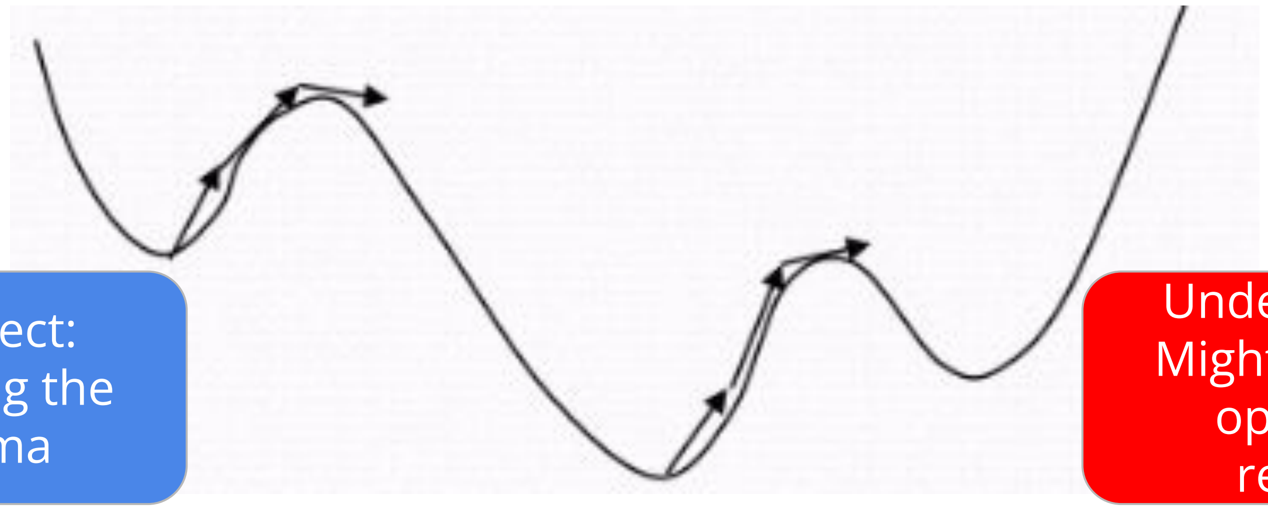
- Cost function must be carefully developed, it has to be “fractal and smooth”.
- The energy function of the left would work with SA while the one of the right would fail.



- The cost function should be fast it is going to be called “millions” of times.

Theoretical Completeness

- There is a proof that if the schedule lowers T slowly enough, simulated annealing will find a global optimum with probability approaching 1
- In practice, that may be way too many iterations
- In practice, though, SA can be effective at finding good solutions
- Consequences of Occasional Ascents

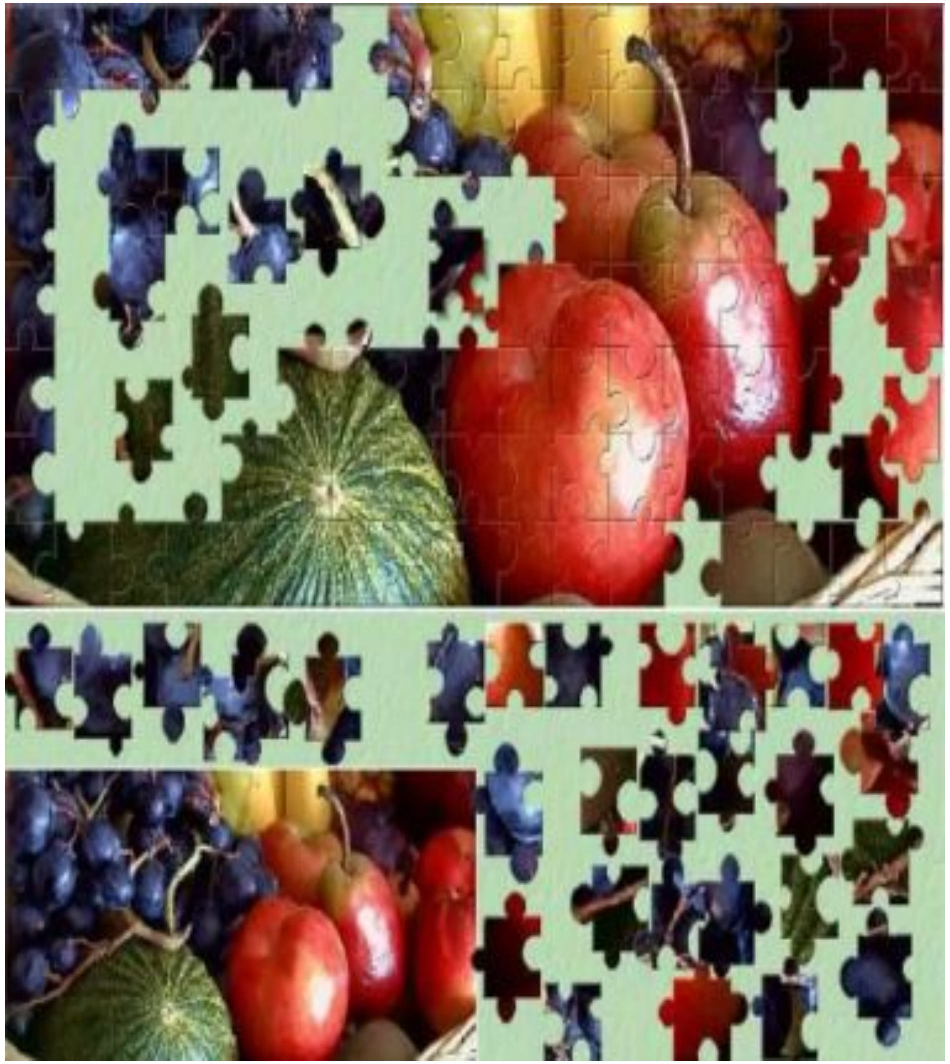


Desired Effect:
Help escaping the local optima

Undesired Effect:
Might pass Global optima after reaching it

Intuitive Usage

- Given a jigsaw puzzle such that one has to obtain the final shape using all pieces together
- Starting with a random configuration, the human brain unconditionally chooses certain moves that tend to the solution.
- However, certain moves that may or may not lead to the solution are accepted or rejected with a certain small probability.
- The final shape is obtained as a result of a large number of iterations.



Simulated Annealing - Pseudocode

Input: Insert optimization function $f(\mathbf{x})$
Initialize system configuration
  Randomize x_0
  Initialize T with a large value
  Initialize ΔT
repeat
  repeat
    Apply random permutation Δx to the state: $x_{i+1} = x_i + \Delta x$
    Evaluate $\Delta E(x) = E(x + \Delta x) - E(x)$:
    if $\Delta E(x) < 0$ **then**
      keep the new state,
    else
      accept the new state with probability $P = e^{-\frac{\Delta E}{T}}$
    end if
  until the number of accepted iterations is below a threshold level
  Set $T = T - \Delta T$
until T is small enough.

Summary

- Improvement of Hill Climbing
- This is a stochastic algorithm
- Convergence can only be realized in asymptotic sense
- It is a Controlled greed algorithm
  - Starts Controlled→ Ends Greedily

References

- Aarst, “Simulated annealing and Boltzman machines”, Wiley, 1989.
- Duda Hart Stork, “Pattern Classification”, Wiley Interscience, 2001.
- Otten, “The Annealing Algorithm”, Kluwer Academic Publishers, 1989.
- Sherwani, “Algorithms for VLSI Physical Design Automation”, Kluwer Academic Publishers, 1999