CAD Algorithms

Simulated Annealing (SA)

Mohammad Tehranipoor

ECE Department

Overview

- Assume that the simulation studies a collection of *M helium atoms* in a cube.
- The position of each atom is described by three parameters that give its coordinates within the cube.
- The energy of this system is given by the sum of all pair-wise interaction energies.
- To calculate the average energy of this system, a simple Monte Carlo simulation should not be used.
  - A method that estimates possible outcomes from a set of random variables by simulating a process a large number of times and observing the outcomes.
- This is because a random placement of the M atoms may place two of the atoms so close together that their interaction energy is virtually infinite. This adds an infinite energy to the ensemble of atom distributions and produces an infinite average energy.

Cont.

- In the real world, two helium atoms would never get that close together.
  - Thus, such placement will be considered a solution.
- A modification to the simple Monte Carlo simulation needs to be made so that *unrealistic samples* are not placed into the ensemble.
- Such a modification was proposed in 1953 by Nicholas Metropolis and coworkers.
- This modified procedure is known as a Metropolis Monte Carlo Simulation.

Metropolis Monte Carlo Simulation

- In contrast with the simple Monte Carlo simulation, a new point in search space is sampled by making a slight change to the current point.
  - For example, a new orientation of the helium atoms is created by making a *random, small change* to each atom's coordinates.
- If the energy of this new orientation is less than that of the old, this orientation is added to the ensemble. If the energy rises, a Boltzmann acceptance criteria is used.
  - If the energy rise is small enough, the new orientation is added to the ensemble.
  - If the energy rise is too large, the new orientation is rejected and the old orientation is again added to the ensemble.

Simulated Annealing

- In 1983, Kirkpatrick and coworkers proposed a method of using a Metropolis Monte Carlo simulation to find the *lowest energy* (most stable) orientation of a system.
- The method is based upon the procedure used to make the *strongest possible glass*.
- This procedure heats the glass to a high temperature so that the glass is a liquid and the atoms can move relatively freely.

Simulated Annealing

- The temperature of the glass is slowly lowered so that at each temperature the atoms can move enough to begin adopting the most stable orientation.
- If the glass is cooled slowly enough, the atoms are able to "relax" into the most stable orientation.
- This slow cooling process is known as *annealing*, and so their method is known as ***Simulated Annealing***.

SA (Cont.)

- Simulated Annealing (SA) is commonly said to be the oldest among the metaheuristics and surely one of the first algorithms that had an explicit strategy to avoid *local minima*.
- The fundamental idea is to allow moves resulting in solutions of worse quality than the current solution (uphill moves) in order to escape from local minima.
- The probability of doing such a move is decreased during the search.

SA Basic Steps

- The algorithm starts by generating an initial solution (either randomly or heuristically constructed) and by initializing the so-called temperature parameter *T*.
- The following is repeated until the termination condition is satisfied:
  - A solution *s1* from the neighborhood *N(s)* of the solution *s* is randomly sampled and it is accepted as new current solution depending on *f(s)*, *f(s1)* and *T*.
  - *s1* replaces *s* if *f(s1)* < *f(s)* or, in case *f(s1)* >= *f(s)*, with a probability which is a function of *T* and *f(s1)* - *f(s)*.
  - The probability is generally computed following the Boltzmann distribution $e^{-(f(s1) - f(s))/T}$.

SA Basic Steps (Cont.)

- The temperature *T* is decreased during the search process, thus at the beginning of the search the probability of accepting uphill moves is high and it gradually decreases, converging to a *simple iterative improvement* algorithm.
- This process is similar to the annealing process of metals and glass, which assume a low energy configuration when cooled with an appropriate cooling schedule.
- The search process shows that the algorithm is the result of two combined strategies: random walk and iterative improvement.

SA

- In the first phase of the search, the bias toward improvements is low and it permits the exploration of the search space; this erratic component is slowly decreased thus leading the search to converge to a (local) minimum.
- The probability of accepting uphill moves is controlled by two factors:
  - The difference of the objective functions
  - The temperature.
- At fixed temperature, the higher the difference *f(s1)- f(s)*, the lower the probability to accept a move from *s* to *s1*.
- The higher *T*, the higher the probability of uphill moves.

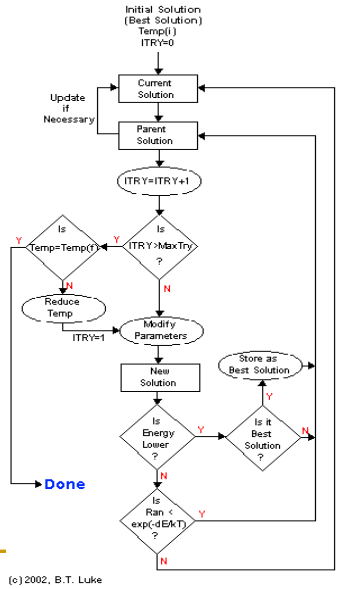
Basic Simulated Annealing

- *s = Generate_Initial_Solution()*
- *T = T_0*
- *WHILE termination conditions not met*
  - *s1 = Pick_At_Random(N(s))*
  - *IF f(s1) < f(s)*
    - *s = s1*
  - *ELSE*
    - *Accept s1 as new solution with probability p(T,s1,s)*
  - *ENDIF*
- *Update(T)*
- *ENDWHILE*

SA Algorithm: With More Details

- *Select an initial temperature t_0 (a large number)*
- *Select an initial solution s*
- *Select a cost function f*
- *Select a neighborhood structure for the solution space*
- *Select a temperature reduction function alpha(t)*
- *Repeat*
  - *Repeat*
    - *randomly select s1 in N(s)*
    - *diff = f(s1) - f(s)*
    - *if diff< 0 then s = s1*
    - *else generate random x in (0, 1) and if x < $e^{-diff/t}$ then s = s1*
  - *until iteration count = max_number_iteration*
  - *t = alpha(t)*
- *until stopping condition*
- *s is the approximation solution*

Simulated Annealing Flow Chart



(c) 2002, B.T. Luke

Advantages

- The main problem with Hill Climbing is getting stuck in a local minimum. Simulated annealing works as a hill-climbing search method that allows moves in less good goal directions once in a while to escape local minima.
- Simulated annealing is proven to converge to the optimum solution of a problem.

Disadvantages

- Although it is proven to converge to the optimum, it converges in infinite time. Not only for this reason, but also since you have to cool down slowly, the algorithm is usually not faster than its cotemporaries.

Cooling Schedules

- Various functions can be used:
- *alpha(t)* = *a*t* with *a* between 0.8 and 0.99
- *alpha(t)* = *t / (1 + b*t)*, with *b* small (around 0)