Simulated Annealing (SA)

1 SA Concept

- Simulated Annealing is patterned after the physical process of annealing, in which physical substances such as metals are melted and then gradually cooled until some solid state is reached.
- SA is a variation of hill climbing in which, at the beginning of the process, some downhill moves may be made.
- SA should lower the chances of getting caught at a local maximum, a plateau, or a ridge.
- The term objective function is in place of the term heuristic function used in hill climbing.
- SA tries to minimize the objective function, valley descending rather than hill climbing.
- There is some probability that a transition to a higher energy state will occur. This probability is given by

$$p = e^{-\Delta E/kT}$$

ΔE = positive change in the energy level

T = temperature

k = Boltzmann's constant

- The rate at which the system is cooled is called the annealing schedule.
- If cooling occurs too rapidly, stable regions of high energy will form. A local but not global minimum might be reached. If too slow schedule is used, time is wasted.
- In SA, the revised probability formula

$$p = e^{-\Delta E/T}$$

ΔE = change of objective function

T = annealing schedule

2 Algorithm

- Evaluate the initial state. If it is also a goal state, then return it and quit. Otherwise continue with the initial state as the current state.
- Initialize *BEST-SO-FAR* to the current state.
- Initialize *T* according to the annealing schedule.
- Loop until a solution is found or until there are no new operators left to be applied in the current state.
    - Select an operator that has not yet been applied to the current state and apply it to produce a new state.
    - Evaluate the new state. Compute ΔE = (value of current)-(value of new state)
        - If the new state is a goal state, then return it and quit.
        - If it is not a goal state but is better than the current state, then make it the current state. Also set *BEST-SO-FAR* to this new state.
        - If it is not better than the current state, then make it the current state with probability $p = e^{-\Delta E/T}$. This step is usually implemented by invoking a random number generator to produce a number in the range [0, 1]. If the number is less than *p*, then the move is accepted. Otherwise, do nothing.
    - Revise *T* as necessary according to the annealing schedule.
- Return *BEST-SO-FAR*.