SIMULATED ANNEALING
(模擬退火法)

13 May 1983, Volume 220, Number 4598

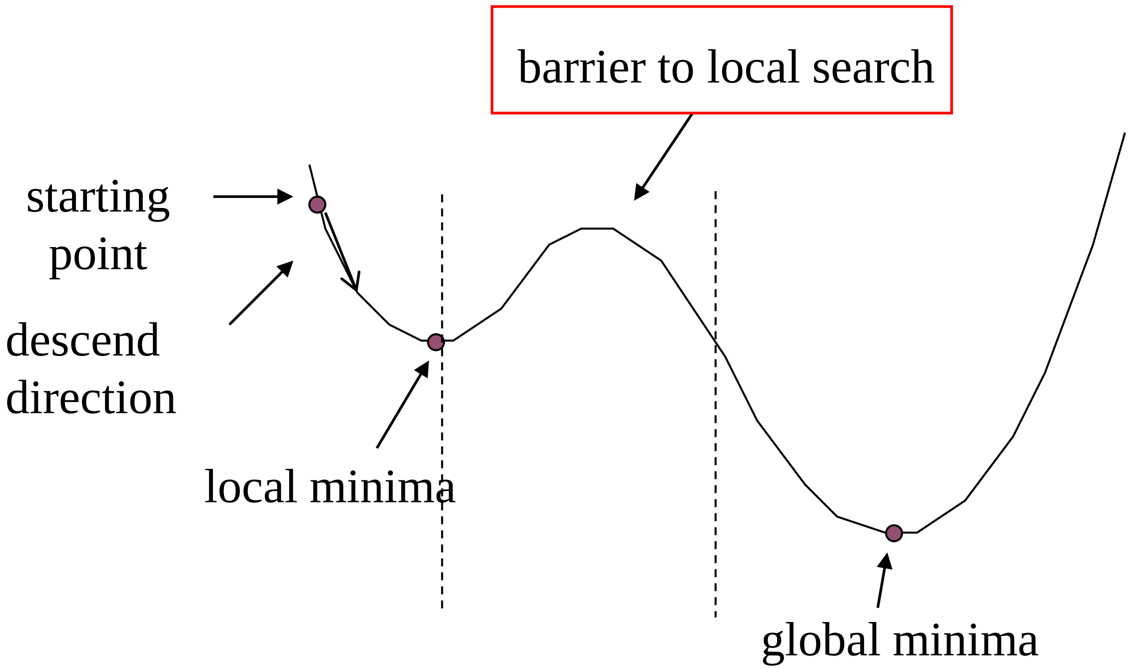
SCIENCE

Optimization by Simulated Annealing

S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi

with N, so that in practice exact solutions can be attempted only on problems involving a few hundred cities or less. The traveling salesman belongs to the large class of NP-complete (nondeterministic polynomial time complete) problems, which has received extensive study in the past 10 years (3). No method for exact solution with a computing effort bounded by a power of N has been found for any of these problems, but if

Difficulty in Searching Global Optima



Local search techniques, such as steepest descend method, are very good in finding local optima. However, difficulties arise when the global optima is different from the local optima. Since all the immediate neighboring points around a local optima is worse than it in the performance value, local search can not proceed once trapped in a local optima point. We need some mechanism that can help us escape (逃脫) the trap of local optima.

Annealing (退火)

- Annealing in metals
- Heat the solid state metal to a high temperature
- Cool it down very slowly according to a specific schedule.
- *If the heating temperature is sufficiently high to ensure random state and the cooling process is slow enough to ensure thermal equilibrium, then the atoms will place themselves in a pattern that corresponds to the global energy minimum of a perfect crystal.*

Intuition of Simulated Annealing

Origin:

Simulation of the annealing process of heated solids.

Intuition:

By allowing occasional ascent in the search process, we might be able to escape the trap of local minima. (allowing the search process to proceed in an unfavorable direction occasionally)

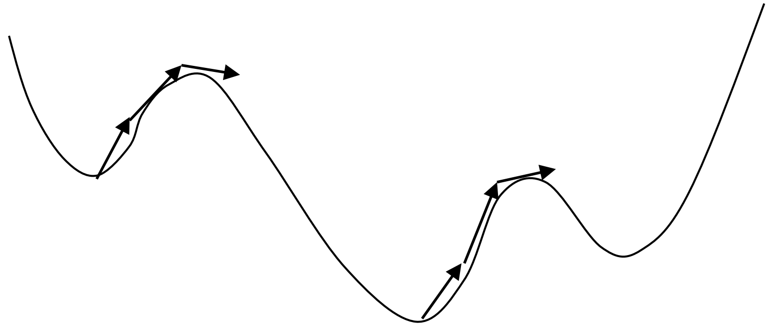
Analogy:

Solutions ~ states in a physical system
Cost of a solution ~ energy of a state

Consequences of the Occasional Ascents

desired effect

Help escaping the local optima.



adverse effect

Might pass global optima after reaching it

➔ It is essential to control the acceptance of occasional ascents (the heart of simulated annealing)

Fundamentals

- Motivation by an analogy to the statistical mechanics of annealing in solids.
- Coerce (強迫) a solid (i.e., in a poor, unordered state) into a low energy thermodynamic equilibrium (i.e., a highly ordered defect-free state) such as a crystal lattice.
- The material is annealed by heating the material to a temperature that permits many atomic rearrangements, then cooled slowly until the material freezes into a good crystal- Metropolis procedure.
- Different from greedy algorithm, SA allows perturbation to move uphill in a controlled fashion to escape from local minima.
- Temperature variable to simulate the heating process

Design Metaphor

- Simulated annealing offers a physical analogy for the solution of optimization problems.
- Boltzmann distribution: a probability distribution that gives the probability that a system will be in a certain state as a function of that state's energy and the temperature of the system

$$p_i \propto e^{\frac{-\varepsilon_i}{kT}}$$

p_i: probability of the system being in state i

ε_i is the energy of that state

kT is the product of Boltzmann's constant k and thermodynamic temperature T

- The probability of a uphill move of size ΔE at temperature T is

$$p(\text{accept}) = e^{\frac{-\Delta E}{kT}} \qquad \Delta E = f(x_{new}) - f(x)$$

- If ΔE<0, the new configuration is accepted.
- If ΔE>0, probability of accepting the worse configuration is calculated based on Boltzmann distribution. (uphill move)
- At higher temperatures T, the probability of large uphill move in energy is large (permits an aggressive, essentially random search of the configuration space)➔ random walk
- At lower temperatures T, the probability is small (few uphill moves are allowed) ➔ hill climbing
- By successfully lowering the temperature (cooling schedule) and running the algorithm, we can simulate the material coming into equilibrium at each newly reduced temperature.

Cooling Schedule

- A starting hot temperature and rule to determine when the temperature should be lowered and how much the temperature should be lowered and when annealing should be terminated.
- *Typically, T*= α*T*, α<1
- If α is very small, the temperature *T* reduces very fast and there is high possibility of being trapped in a local minimum
- If α is large, the energy decreases very slowly (slow convergence)
- Many schemes to reduce temperature
  - Fast Cauchy: $T_k = \frac{1}{k} T_0$
  - Geometric: $T_k = \alpha T_{k-1}$
  - Boltzmann: $T_k = \frac{1}{\ln(k) + 1} T_0$

  where T_0 is the initial temperature, and T_k is the temperature after the *k*'th temperature decrement

Pseudo Codes

- M = number of moves (perturbations) to attempt
- T = current temperature
- For $m = 1$ to M

  Generate a random move

  Evaluate the change in energy ΔE

  IF ($\Delta E < 0$)

      Accept this move and update configuration /* downhill move

  ELSE

      Accept with probability, $P = e^{-\frac{\Delta E}{T}}$ (larger than a random number)

      Update configuration if accepted

  ENDIF

  Update temperature $T = \alpha T$

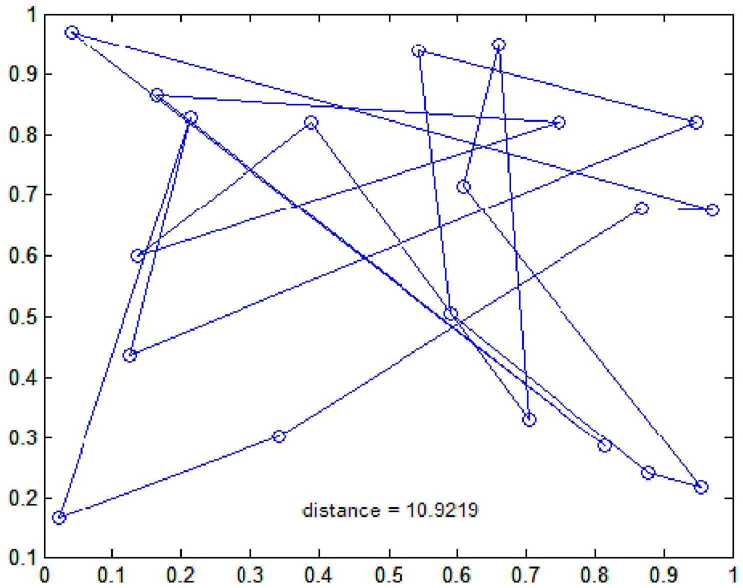
- ENDFOR

Simulated Annealing for TSP

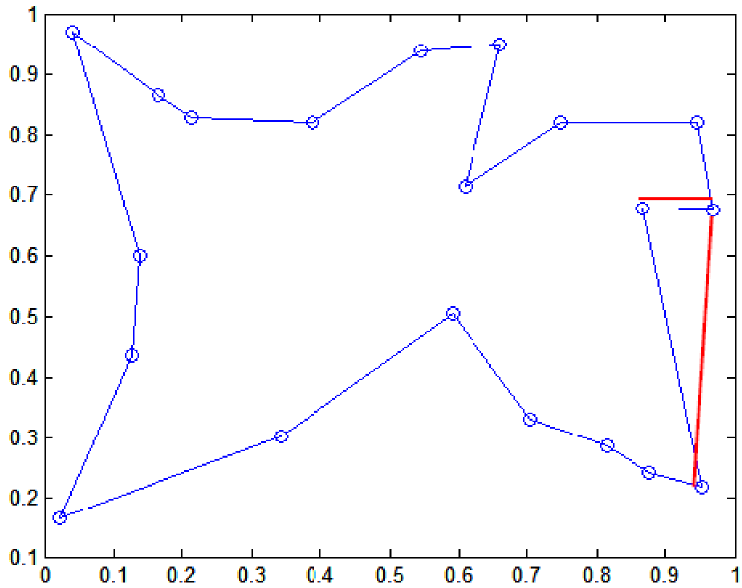
- Initial configuration: permutation =>1, 2, 3, 4, …, N
  - temperature $T = 2$
  - cooling rate $\alpha = 0.99$
  - energy = $d(1,2)+d(2,3)+\ldots+d(N,1)$
- Generate a new configuration from the current one at random
- Evaluate ΔE = current energy – previous energy
- If $\Delta E < 0$ accept the current configuration (downhill)

  Else accept configuration with probability $P = e^{-\frac{\Delta E}{T}}$
- $T = \alpha T$
- Stopping criteria if energy < threshold or number of iterations is reached.

Case Study: 20-city problem

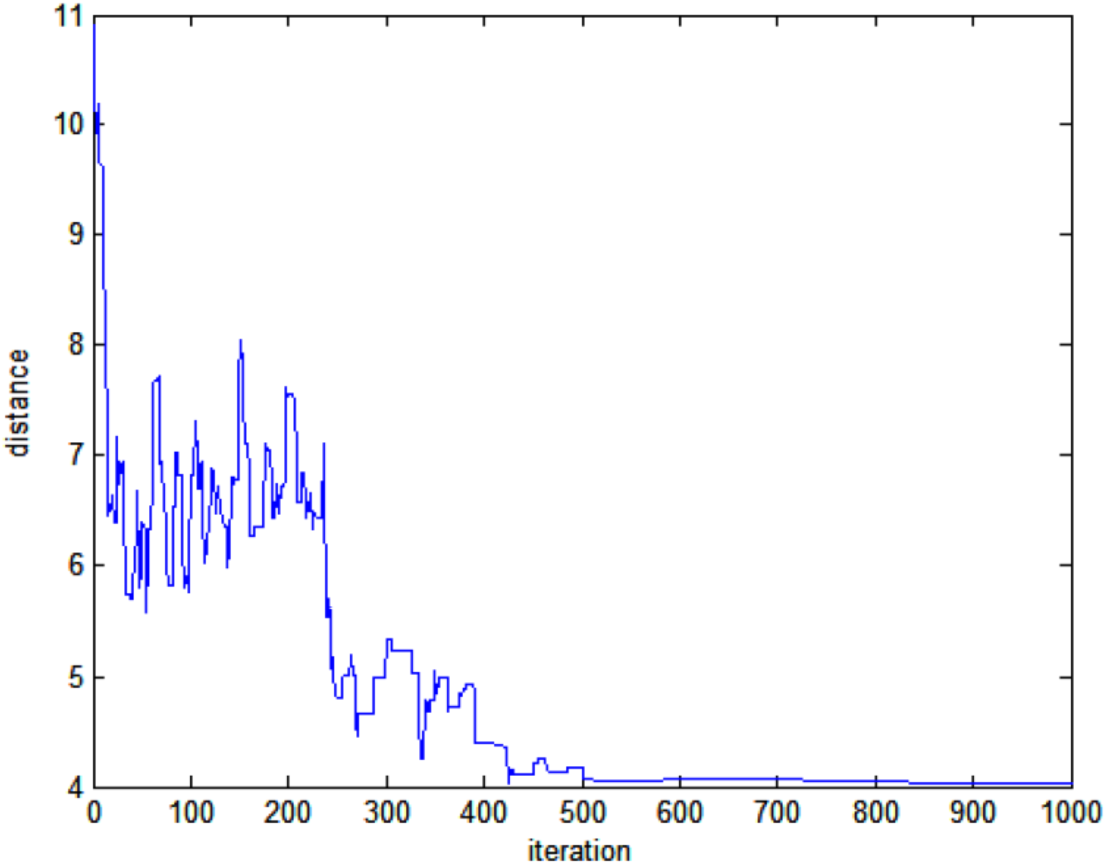


Initial random permutation



Best route found for 20-city problem
d=4.0185 (d=4.0192)

Initial temperature	2	2	2	2	2
Cooling rate	0.8	0.9	0.95	0.99	0.999
Converged iteration	87	155	298	$\boxed{851}$	$\boxed{5982}$
Minimal energy obtained	4.1056	4.0920	4.0427	4.0185	4.0185



$T_0 = 2$
$\alpha = 0.99$
Iteration = 851
d = 4.0185

Case Study: 101-city problem (ali101)

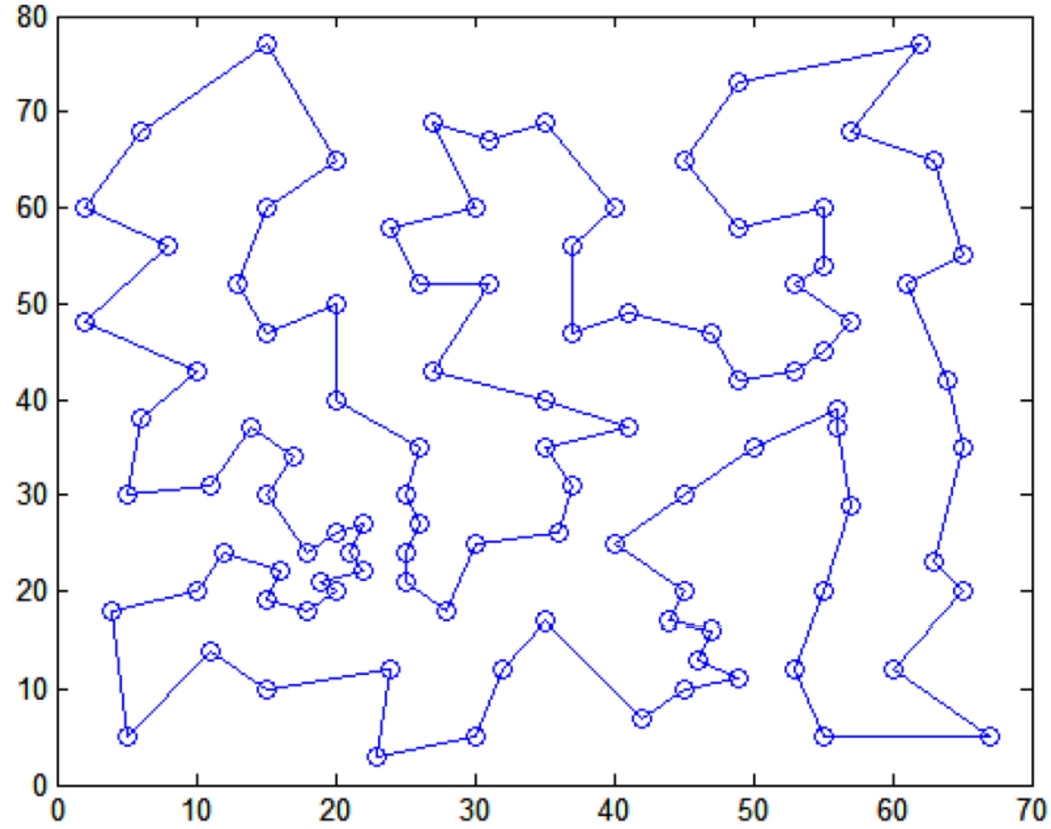
T_0=200

α= 0.999

Iterations = 100,000

d = $\boxed{661.25}$

Global minima = 629

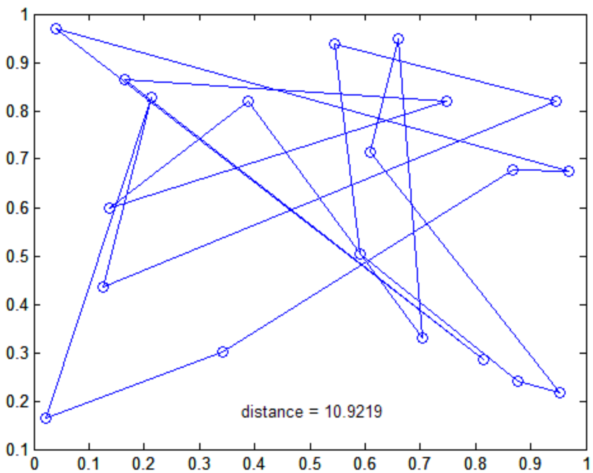


Research Issues

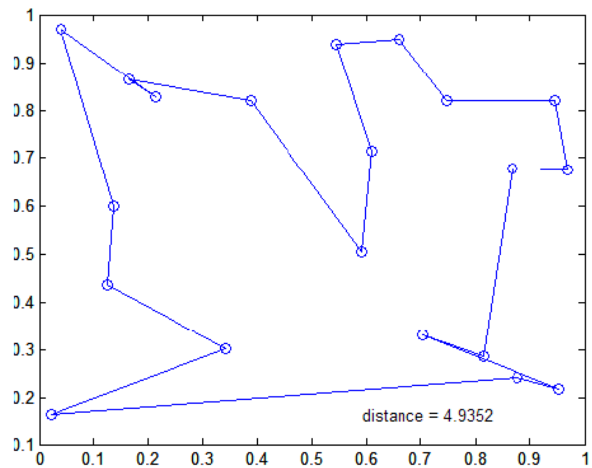
- Initial Temperature
- Initial Configuration
- Next Move (neighborhood size)
  - is it possible to design a truly local search for combinatorial optimization problem (as oppose to numerical optimization problem)?
- Cooling Schedule (cooling rate)
- Stopping Criteria
- Acceptance Probability

Effect on Initialization

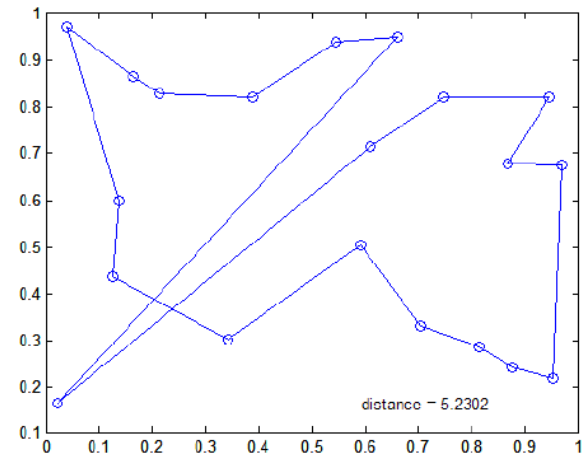
Random sequence



Hilbert space filling curve



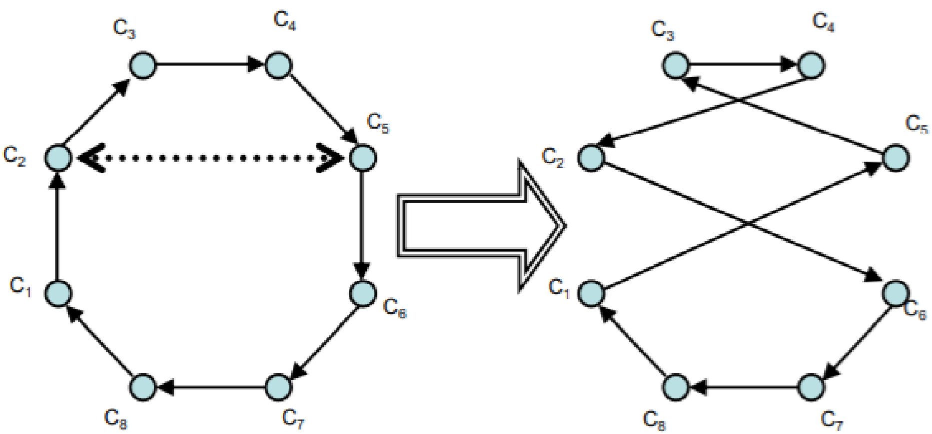
Fastest closest path



Experimental results indicate that the initialization method does not affect much the probability of the system to converge to the global optimal point.

Effect on Next Move Modification

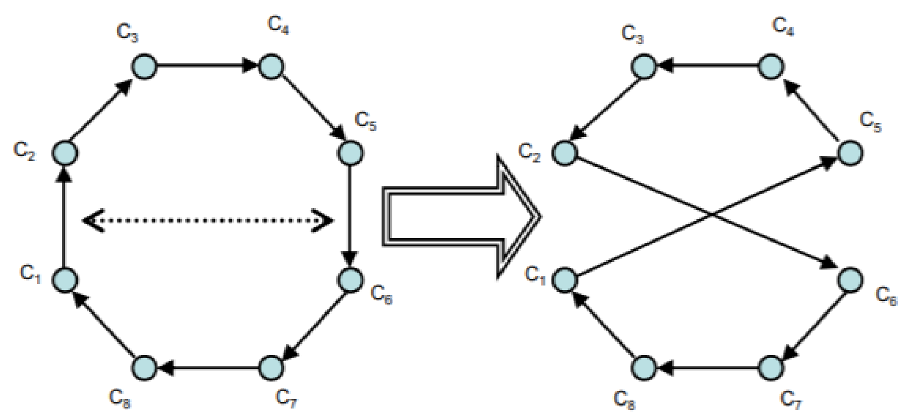
Random switch of two vertices



1 2 3 4 5 6 7 8

1 5 3 4 2 6 7 8

Random switch of two edges

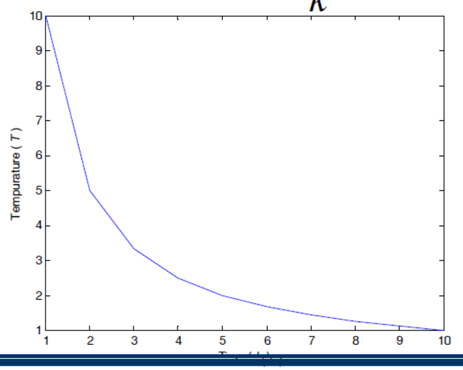


1 2 3 4 5 6 7 8

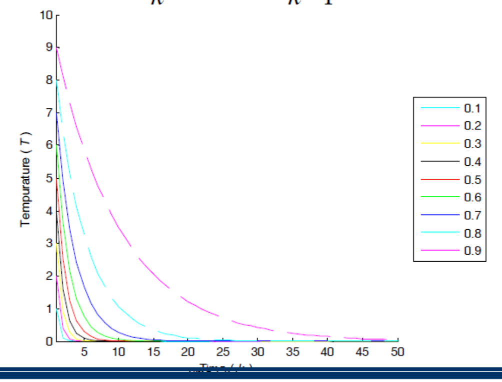
1 5 4 3 2 6 7 8

Effect on Cooling Schedule

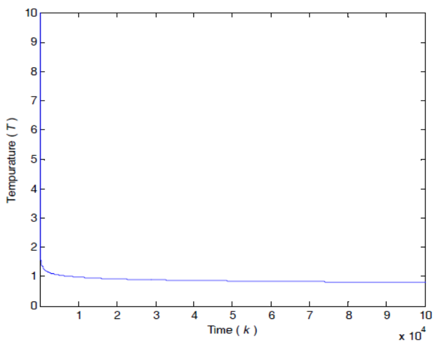
Fast Cauchy $T_k = \frac{1}{k} T_0$



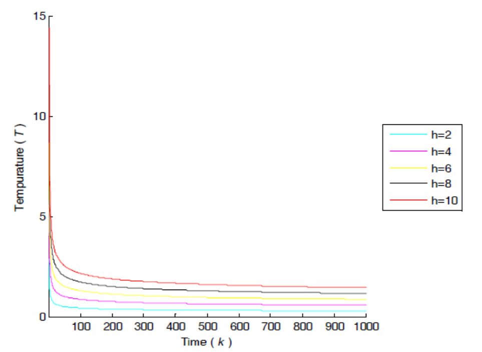
Geometric $T_k = \alpha\, T_{k-1}$



Boltzmann $T_k = \frac{1}{\ln(k) + 1} T_0$



Logarithmic $T_k = \frac{1}{\ln(k+1)} h$



- ***Conjecture***: In earlier stage, *raising the temperature* makes escaping from local minima easier. However, how much changes should be allowed was not addressed?
- ***Theorem***: SA with *larger neighborhoods* has a greater probability of arriving at a global optimum than generic SA has if the other conditions, i.e., the initial configuration, initial temperature and cooling rate are kept the same.
- ***Guideline***: In earlier stage, allow the next_state to be generated from a larger neighborhood of the current state. As the process continues, the neighborhood should be reduced accordingly.

Applications

- **Computer-aided VLSI design**
  – Simulated annealing for VLSI design, Kluwer Academic, 1988
- **Combinatorial optimization**
  – Simulated annealing: theory and application, Reidel Publishing, 1987
- **Neural network training**
  – A learning algorithm for Botzman machine, Cognitive Science, 9: 147-169, 1985
- **Image processing**
  – Image processing by simulated annealing, IBM Journal of Research and Development, 29: 569-579, 1985
- **Code design**
  – Using simulated annealing to design good codes, IEEE Trans Information Theory, 33: 116-123, 1987
- **Function optimization**
  – An empirical study of bit vector function optimization, in Genetic Algorithm and Simulated Annealing, Chapter 13, 170-204, 1987
- **and many more!**

Selected References

- **S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi, "Optimization by simulated annealing,"** ***Science*****, 220(4598), pp. 671-680, 1983.**
- B. Moon, H.V. Jagadish, C. Faloutsos, and J.H. Saltz, "Analysis of the clustering properties of the Hilbert space-filling curve," *IEEE Transactions on Knowledge and Data Engineering*, 13(1), pp. 124-141, 2001.
- S. Lin and B. Kernighan, "An effective heuristic algorithm for the traveling salesman problem," *Operations Research*, 21(2), pp. 498-516, 1973.

Homework #3A (by SA or NM Simplex, optional)

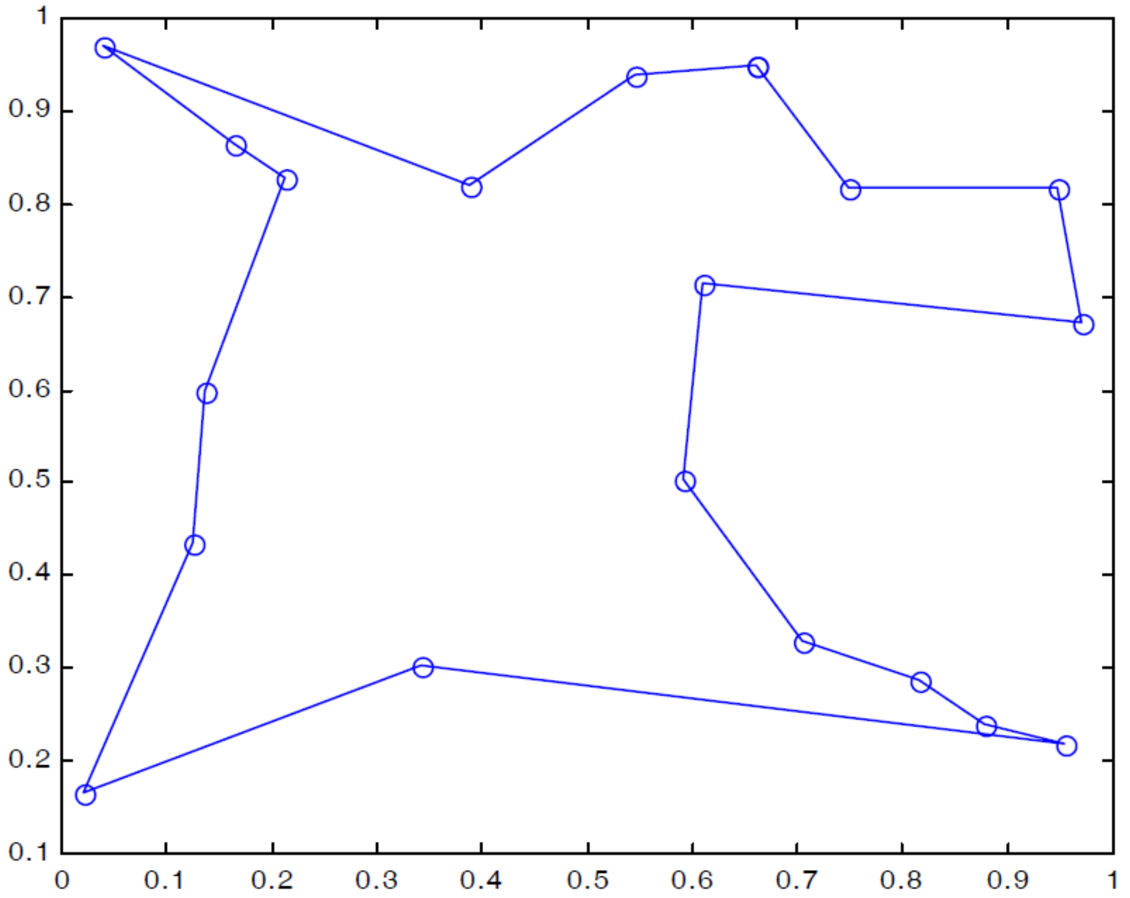
- **Problem #1 (*Combinatorial Optimization*)**

  Develop a *generic* simulated annealing algorithm to solve the traveling salesman problem with 20 cities that are uniformly distributed within a unit square in a 2-dimensional plane. The coordinates of 20 cities are given below in a matrix:

$$\text{cities} = \begin{bmatrix} 0.6606,0.9695,0.5906,0.2124,0.0398,0.1367,0.9536,0.6091,0.8767,0.8148 \\ 0.9500,0.6740,0.5029,0.8274,0.9697,0.5979,0.2184,0.7148,0.2395,0.2867 \end{bmatrix}$$

$$\begin{bmatrix} 0.3876,0.7041,0.0213,0.3429,0.7471,0.5449,0.9464,0.1247,0.1636,0.8668 \\ 0.8200,0.3296,0.1649,0.3025,0.8192,0.9392,0.8191,0.4351,0.8646,0.6768 \end{bmatrix}$$

- Show the “best” route you find and the associated distance with attached computer coding. An example is given below for reference.


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1