Optimization Problems: Traveling Salesman

Other techniques that have been applied to the traveling salesman problem:

- Genetic Algorithms. Quite successful for solving TSP. Numerous applications to other types of problems. Software based approach.

- Neural Networks. Not so successful for solving TSP. Many applications to other types of problems. Both software and hardware based approaches.

Genetic Algorithms

Genetic algorithms try to model evolution by natural selection. In nature the *genetic code* is stored in DNA molecules as sequences of bases: adenine (A) which pairs with thymine (T), and cytosine (C) which pairs with guanine (G).

The analog of DNA in a digital genetic algorithm is a sequence of binary digits (0) and (1).

In nature, the genetic code describes a *genotype*, which is translated into an organism, a *phenotype*, by the process of cell division.

Digital genetic algorithms can be used to solve a problem, such as finding the global minimum of a complicated energy landscape. The phenotype in a genetic algorithm is some state of the model: strings of binary digits are mapped to the states of the model to be solved.

Evolution by natural selection is driven in part by changes to the genetic code:

Mutations: Random changes can occur, for example caused by radioactivity or cosmic rays damaging a DNA molecule. Mutations of the digital genotype can be modeled by choosing a random bit in the string and changing it $1 \to 0$ or $0 \to 1$.

Recombination or Crossover: During sexual reproduction the offspring inherit DNA from each of the parents. This can be simulated by taking two strings and exchanging two substrings.

Survival of the Fittest: There is some criterion of *fitness* such that when mutations or recombinations take place, the mutants or offspring either survive and reproduce or die out.

These simple ingredients can be used to construct a very wide variety of genetic algorithms. A simple algorithm which can be applied to an energy landscape problem is illustrated by the random Ising model:

$$E = -\sum_{\langle ij \rangle} T_{ij} s_i s_j \ ,$$

where $s_i = \pm 1$ are Ising spins, and the coupling constants T_{ij} between nearest neighbors are chosen randomly to be ± 1. This is a model of a *spin glass* which has a very complicated energy landscape with numerous local minima.

What is a genotype for this model? Suppose we have a 2-D lattice of spins with $i, j = 0, 2, \ldots, (L-1)$, then we can order the spins linearly using the formula $n = iL + j = 0, 1, \ldots, (L^2 - 1)$ for example. A configuration is of spins is mapped to a genotype of L^2 bits by setting the bit with index n to 0 or 1 if $s_{ij} = \pm 1$.

Since we are seeking the global energy minimum, the *fitness* of a particular genotype can be taken to be $2L^2 - E$, since the minimum and maximum possible values for the energy are $\mp 2L^2$ for a 2-D square lattice and periodic boundary conditions. (Recall that the number of bonds is then twice the number of spins.)

The following is one possible evolution protocol:

- Start with a population of a fixed number N_0 of strings initialized in some way, for example by setting the string bits randomly.

- Repeat the following "generations":
  - Allow some number of mutations. For example, choose 20% of the strings at random, and mutate a random bit (flip a random spin) in each string.
  - Choose some number of pairs of strings at random and have them "reproduce" as follows: each pair produces two offspring which differ from the parents by exchange of a randomly chosen substring.

- The size of the population has now increased from N_0 to N due to reproduction, and the parents and children are competing for the same limited natural resources. Select N_0 fittest survivors as follows:
  * Construct a cumulative histogram

$$H_k = \sum_{i=1}^{k} (2L^2 - E_i) , \qquad k = 1, 2, \ldots, N ,$$

    where k labels the strings in the population.
  * Repeat N_0 times:
    · Choose a random H between 0 and the maximum H_N.
    · Select the smallest k such that $H_k > H$.

After many generations the population should converge to the global energy minimum configuration!

Neural Network Models

Genetic algorithms are modeled on evolution due to natural selection. Neural network algorithms are modeled on the working of nerve cells or *neurons* in the brain.

A crude binary model of a neuron is that it can be in one of two states, a resting state which can be represented by binary 0, and an active or *firing* state in which an impulse or signal is transmitted along the *axon* which is a long fiber extending from the cell body or *soma*.

The axon of a neuron branches multiply and connects to other neurons via *synapses*, which are essentially chemical junctions.

What determines the state of a neuron? A simple model is that the neuron sums all of the input signals from other neurons which synapse to it: if this sum is larger than a threshold value, then it fires, and otherwise it does not.

Hopfield introduced a simple model based on these ideas in *Proc. Natl. Acad. Sci. USA* **79**, 2554 (1982) which simulates the storage and retrieval of memories. Consider a network of N neurons. The state of the network is defined by specifying a binary valued *potential* $V_i = 1$ or 0 at each neuron: if $V_i = 1$ then neuron i is firing, while if $V_i = 0$ it is not. The *synaptic strength* between neurons i and j is denoted T_{ij}. The integrated signal at neuron i is

$$S_i = \sum_{j \neq i} T_{ij} V_j \; .$$

The state of this neuron is set according to the criterion

$$V_i = \begin{cases} 1, & \text{if } S_i > 0 \\ 0, & \text{if } S_i \leq 0 \end{cases} \; .$$

The network is operated by updating the neurons according to some protocol, for example by choosing neurons at random or sequentially (which is usually what is done in software networks), or by updating the whole network synchronously (which is more natural for a hardwired network controlled by a clock).

Hopfield showed that the network tends to the global minimum of the function

$$E = -\sum_{\text{pairs}} T_{ij} V_i V_j \ ,$$

which represents the energy of a random spin glass with spin variables $s_i = 1 - 2V_i = \pm 1$.

The energy landscape depends on the the synaptic strengths of the network T_{ij}. It turns out that these strengths can be used to store patterns represented by states of the network according to *Hebb's Rule*:

$$T_{ij} = \sum_{p=1}^{P} (1 - 2V_i^{(p)})(1 - 2V_j^{(p)}) \ ,$$

where P is the number of patterns stored and $V_i^{(p)}$ is the state of neuron i in pattern p.

Hopfield showed that

- The network dynamics decreases the energy of the network This implies that if the network is started in an arbitrary state, then it will evolve to the nearest local energy minimum.

- The stored states are local minima of the energy function. So if the initial state happens to be in the *basin of attraction* of one of the stored minima, the that pattern will be recalled!

A network with N neurons has a huge number 2^N states. The network works best if the stored memories partition the space of network states into well defined basins. The storage capacity of the network is found to be $\sim 0.13N$. If too many memories are stored, then the minima are not well defined and memories may not be perfectly recalled.