(Year of 2004-2005)

Possible Topics for Semester Projects

proposed by Akira Imada

(last modified on)

October 12, 2004

Abstract

Possible topics for undergraduate semester project are listed bellow. The first four are explorations of some test-functions whose solutions are already known. The purpose is to know how Evolutionary Computations work. The last two are topics from *Biologically Inspired Approach*, in general, Genetic Algorithms, in particular, to NP-hard Combinatorial Problems. Pick up one item from the list and then read the text to learn a little more in detail.

Index

- Test with an already known solution:
  1. Multi-dimensional Schwefel Function

     Let's search for the global minimum among infinite number of local minima.
  2. A multi-peak 2-D function

     If we have multiple solutions, how randomly created chromosomes in the 1st generation evolve to find those solutions in a run?
  3. Jeep Problem

     How deep can a Jeep penetrate in the desert with a limited amount of gasoline given? — An optimization problem.
  4. A Needle in a Haystack Problem

     How evolutionary computations find a needle in a haystack — Baldwin Effect.
- A challange to an NP-complete problem:
  5. Traveling Sales-person Problem (TSP)

     To look for near optimum (almost shortest) tour to visit N cities which is an NP-complete problem.
  6. Knap-sack Problem

     Yet another NP-complete problem. We have a limitied capacity of a knapsack and we have to pick up items from list of items in which the price and size of each item are given. The goal is maximize the total price in our knapsack.

1 Re-visit to an already known solution

Bellow listed a couple of testbeds whose solutions were already known, and therefore it is useful to explore this test function to learn how evolution works inside the PC.

1.1 Shcwefel function

The function bellow is called Schwefel function and enormous amount of local minimum and the only unique global minimum at the Origin. Search for the global minimum when the function is difined on, say, $x_i \in [-500, 500]$.

$$y = \sum_{i=1}^{n} (x_i \sin(|x_i|)) \tag{1}$$

You might try to explore this hyper-surface by setting n to 20, for example. Then the surface is defiened on 20-dimensional Eucledean space. If, however, you want to know the image of the hyper-surface, see a two dimensional version of this function. The graph of

$$y = x \sin(|x|) \tag{2}$$

is shown in Figure 1

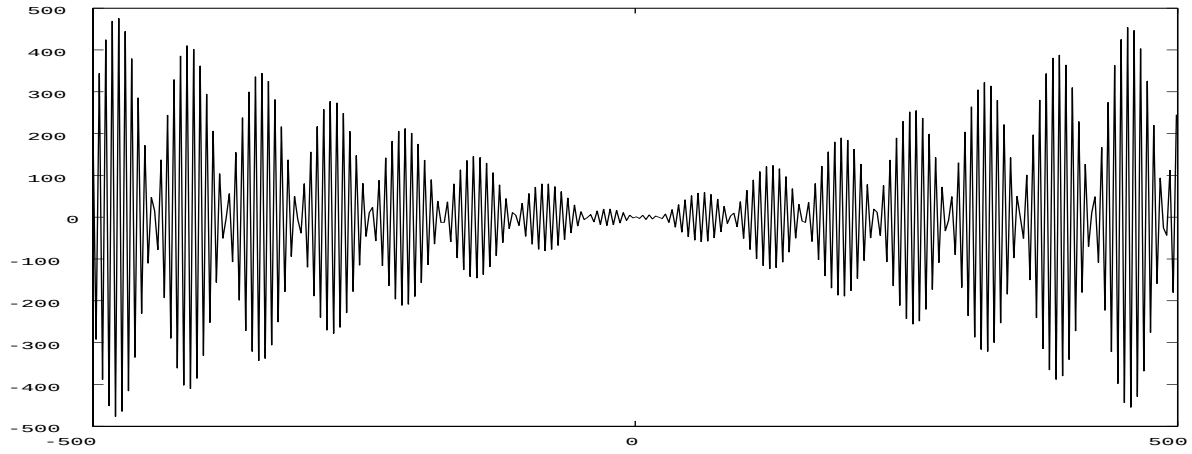

Figure 1: A two-dimensional version of the Schwefel's Function's graph.

1.2 Yet another Testfunction — A 2-D function but multi peaks

The test-function of the previous subsection is the one which evolutionary computations are especially good at, since we can treat whatever large dimension simply by setting the number of genes in a chromosome to the dimensionality.

Then what should we do, if we are interested in a 2-D function? For example,

$$y = \sin^6(5\pi x) \tag{3}$$

or

$$y = -2((x - 0.2)/0.8)^2 \sin^6(5\pi x) \tag{4}$$

are interesting functions in order for us to observe how randomly created chromosomes in the 1st generation evolve to find peaks. See Figure 2.

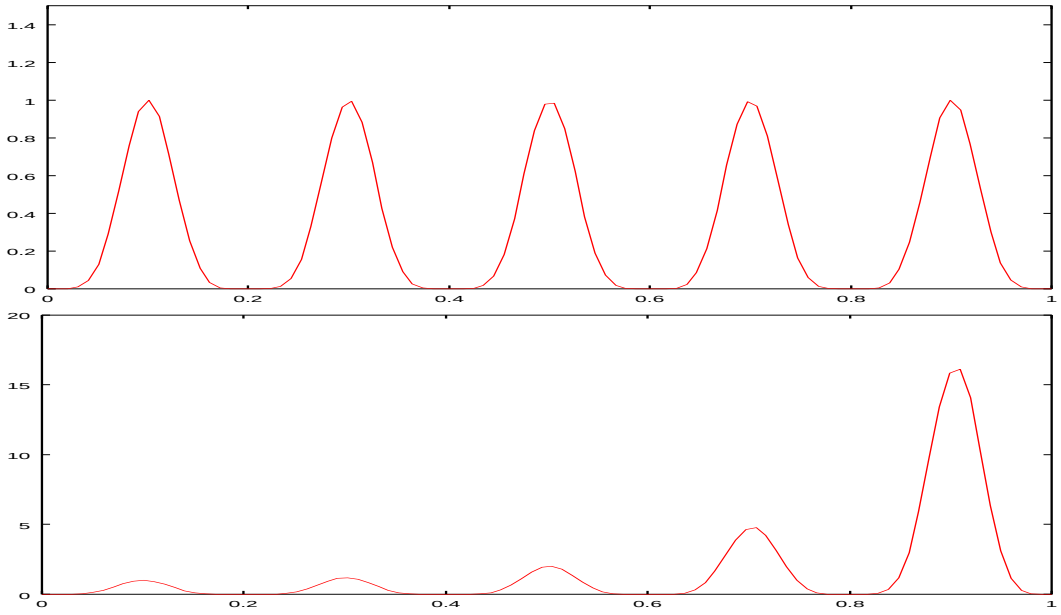

Figure 2: A multi-peak 2-D function and its variation

The question is how we design our chromosomes. In the multi-dimensional function $y = f(x_1, x_2, \cdots, x_n)$ our genes might be continuos value each of which corresponds to the independent variable x_i $(i = 1, 2, \cdots n)$, that is, our chromosomes are made up of n genes. On the other hand how should we design our chromosomes when the number of independent variable is only one. A chromosome with only one gene? How we crossover two parents?

O.K. we usually use binary chromosome in this situation. Any (decimal) real-valued variable $x_i \in [a, b]$ could be encoded by n-bit binary strings where a and b is represented by $(00\cdots0)$ and $(11\cdots1)$, respectivley, and therfore accuracy (or granularity) is $(b-a)/(2^n-1)$. For example, if our concern is the above $x \in [0, 1]$ then 10 bit of binary strings from 0000000000 to 1111111111 are expresses decimals with the precision of 1/1024. Or you might use and comapre *Gray Code* where gray-code $a_1a_2\cdots a_n$ is tlanslated from binary number $b_1b_2\cdots b_n$ as

$$a_i = \begin{cases} b_i & \text{if} \quad i = 1 \\ b_{i-1} \oplus p_i & \text{otherwise} \end{cases} \tag{5}$$

where $\oplus$ is addition modulo 2. In gray code a pair of adjusent decimals are different only with Hamming distance 1, while in the standard binary encode this does not hold.

1.3 A Needle in a Haystack.

As shown in Figure 3 — a conceptual diagram of fitness landscape plotted on a fictitious 2-d space — we usually have a gradient information when we speak of peaks in a fitness landscape.

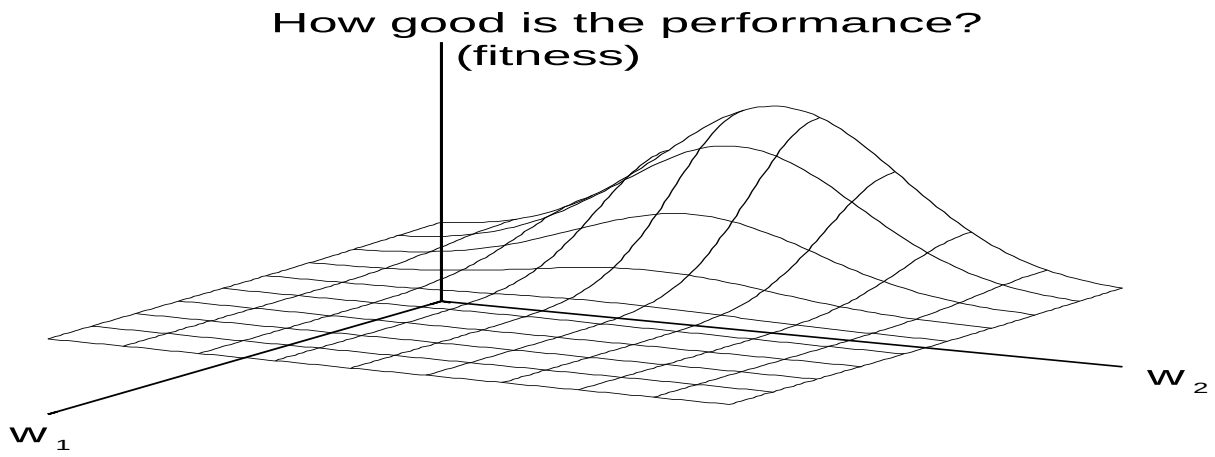

Figure 3: A conceptual diagram of fitness landscape.

But sometimes the shape of a peak is something like in the Figure 4 (Top), and the most difficult case, above all, is like in the Figure 4 (Bottom) which is widely known as *a-needle-in-a-haystack* problem.

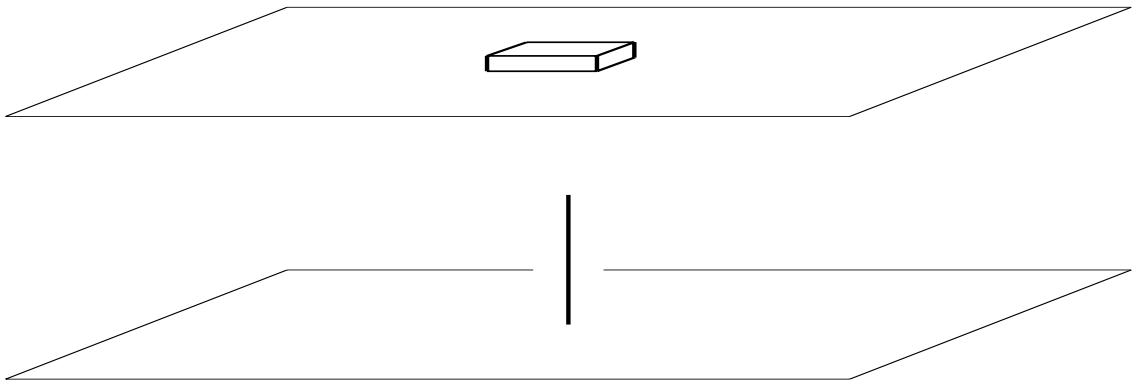

Figure 4: Two special fitness landscapes: A-tiny-flat-island-in-a-huge-lake (top) and A-needle-in-a-haystack (bottom).

One most famous example of a-needle-in-a-haystack problem is only one configuration of 20 binary bits is assigned fitness one and all the other configuration is assigned fitness zero. For example, (11111111110000000000) — a needle — is assigned fitness one, and all the other $2^{20} - 1$ points are assigned fitness zero — a haystack. Our task is to design a needle detector conceptually shown in Figure 5.

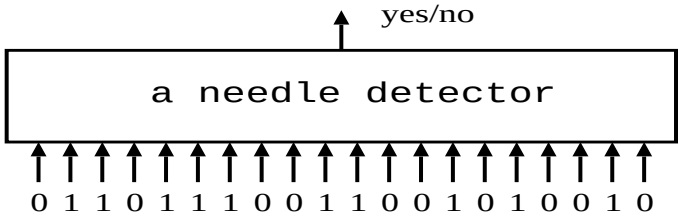

Figure 5: Conceptual black-box to detect a needle. Only the input of one specific configuration retures YES, otherwise NO.

1.4 Jeep Problem

Assume we have a Jeep at the base which locates at the edge of a desert. The Jeep has a tank which can be filled with a maximum of one-unit of gasoline. With one unit of gasoline, the jeep can move one unit of distance. The Jeep can only fill gasoline at the base.The jeep can carry containers to put its gasoline on the desert for a future use, Assume the tour should be on the strait line in the desert.

The question is "How far the jeep can penetrate in to the desert on the straight road when n unit of gasoline is available at the base.

For example when $n = 2$, the best strategy is to start the base with one unit of gasoline in its tank and go 1/3 unit distance (it has spent 1/3 unit of gasoline to reach the point, then put 1/3 unit of gasoline in the container there (now jeep has 1/3 unit of gasoline in the tank) and go back to the base. Exactly when the jeep arrive to base all gasoline filled at the start was spent. Then geep fill the 2nd unit of gasoline given, go 1/3 unit fill the gasoline he had put before and the tank is again full, then go forward until all the gasoline in the tank will be spent. Therefore the maximum distance the jeep can go is 4/3 unit of distance.

Guess the maximum distance when $n = 2$. We already know the maximum distance with n unit ofgasoline is D_n is expressed as the recursive relation $D_n = D_{n-1} + 1/(2n - 1)$.

Our interest is on whether an evolutionary computation can find a almost best strategy, say, $n = 5$ in which maximum distance is 1323/945=1.4. (If my calcuration is correct. Try it by yourself).

2 Approach to NP-hard Combinatorial Problems.

2.1 Traveling Salse-person Problem by a Genetic Algorithm.

Asuuming N cities all of whose cordinate are given. Traveling Salse-person Problem (TSP) is a problem in which a sales-person should visit all of these cities once but only once and objective is to look for the shortest tour.

I found that 13,509 real cities in USA are given with their cordinates in the web-page
http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95.
Why don't we try this very challenging problem by ourselves. The plotted these cities are

shown in Figure 6.

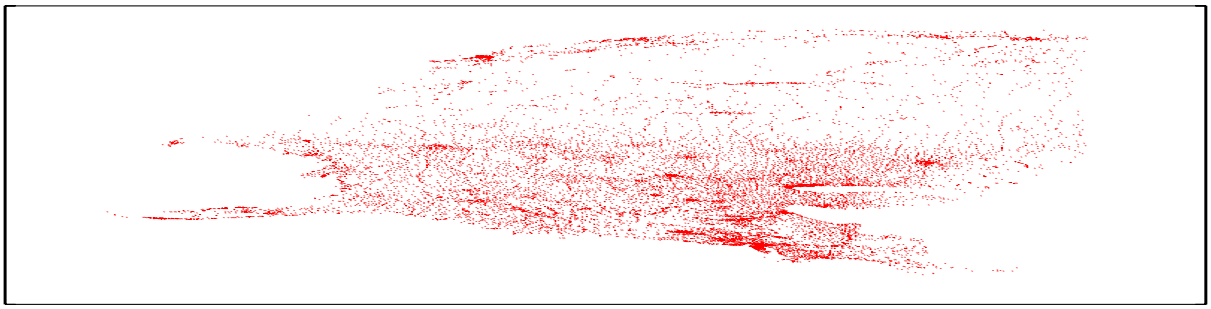

Figure 6: An example of 13509 real cities' location in US. Ploted with the data taken from http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95.

2.2 Also TSP but by Ant Colony Optimization.

Ant Colony Optimization (ACO) is an optimization technique we borrowed the metaphore of an intelligence of ant society as their collective behaviour. Ants are good at seeking a shortest path from their nest to a food when one of their colleague find it by using a chemical called pheromon. It seems more direct way of computation comparing to by GA.

2.3 Approach to Knap-sack Problem by GA.

Knap-sack Problem is also an NP-hard Combinatorial Problems. Assume we have N items all of whose price and size are given. We have one knap-sack which has a limited capacity. We pick up items to carry with the knap-sack such that the total price should be maximize. Don't forget our knap-sack has a limited capacity.

3 If you are interested...

Then come to me, and I'll give you more information about the topic. Or you might collect information via http://google.com for example.

Have fun!