CSCI 4560/6560 Evolutionary Computation

Assignment Number 4: Due Tuesday 11/8/2005 **(in class)**

1. [**60 points**] Consider the following **continuous** variable optimization problem:

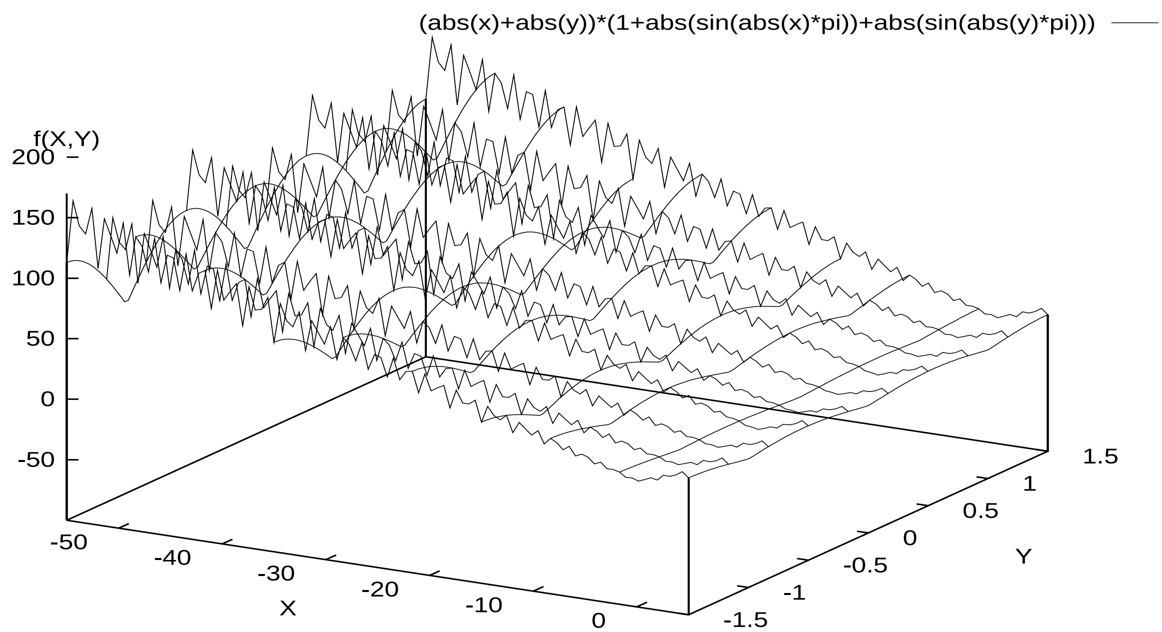
   $minimize \quad f(x, y) = (|x| + |y|) \cdot (1 + |sin(|x| \cdot \pi)| + |sin(|y| \cdot \pi)|)$

   $subject \quad to:$

   $-60 \leq x \leq 40$

   $-30 \leq y \leq 70$

   The objective function surface plot is shown in the figure below.



   (a) Use an Evolutionary Algorithm (Genetic Algorithm or Evolution Strategy or any suitable Evolutionary Computation method of your choice) to find the optimum. Your EA should use no more than **2000** fitness function evaluations (NOT 2000 generations!) every run. Run the EA 10 times (with different random initial populations) and report the best point found in each time. You may download and use an existing EA or implement your own.

   (b) Repeat the above experiment after you change the function to:
   $f(x, y) = (|x| + |y|) \cdot (1 + |sin(3 \cdot |x| \cdot \pi)| + |sin(3 \cdot |y| \cdot \pi)|)$
   This increases the number of local optima approximately 10 times. You should still use no more than **2000** evaluations in each optimization run.

(c) Discuss the results comparing the EA's performance before and after the increase in the number of local optima. How much did the perfomance suffer due to the increase?

2. [**40 points**]

Consider Ackley's function with **ten** dimensions. This function is described in the text book page 268. Use an Evolutionary Algorithm to find the optimum. Your EA should use no more than **10000** fitness function evaluations every run. Run the EA 10 times (with different random initial populations) and report the best point found in each time. You may download and use an existing EA or implement your own. You may also use the same EA for this problem that you used for the above problem (and I strongly recommend it because to do otherwise would be a waste of your time) but this is up to you.

You should try to experiment with your EAs to get the best results. In all problems, the global minimum is Zero.