COURSEWORK EXERCISE 3

Introduction

In this last 4M17 coursework exercise you will investigate the performance of some of the methods introduced in lectures on a problem that is easy to describe but harder to solve.

This assignment counts for 50% of your final grade for 4M17. You should expect to spend 16 hours (including the two timetabled sessions) on it. Time spent reading email or surfing the web while your code runs in the background does not count!

The Problem

The problem you will be attempting to solve is the so-called *Bird Function*. This is a 2-dimensional constrained optimization problem, defined as follows:

Minimize $f(\mathbf{x}) = \sin(x_1)\exp\left[\left(1-\cos(x_2)\right)^2\right] + \cos(x_2)\exp\left[\left(1-\sin(x_1)\right)^2\right] + (x_1 - x_2)^2$

subject to $-6 \le x_1 \le 6$

$-6 \le x_2 \le 6$

As shown below in Figure 1, this is a problem with multiple local maxima and minima and two equal-valued global minima. It is therefore quite a hard optimization problem to solve.

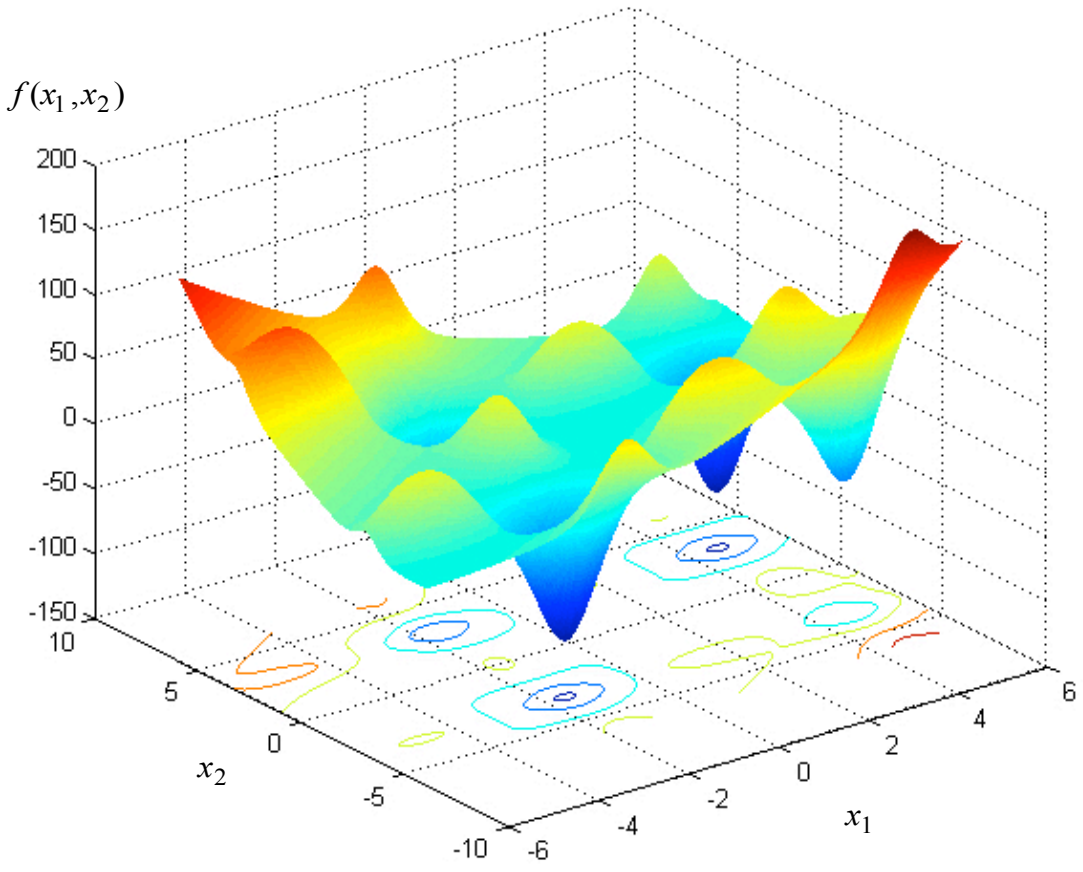

Figure 1: A 3D plot of the Bird Function.

On the plus side, however, all the control variables are of the same type (continuous variables) and have similar (in fact, identical) scales, all the constraints are inequality ones (and indeed just bounds) and the feasible space is not disjoint.

Coursework Tasks

The aim of this coursework exercise is to compare the performance of two of the optimization algorithms you have learnt about in 4M17 on the Bird Function.

One of the algorithms must be one of the methods covered in Dr Parks' lectures (i.e. a Genetic Algorithm, an Evolution Strategy or Tabu Search); the other algorithm can be another of these methods or Simulated Annealing or a direct search method (Nelder-Mead or Hooke & Jeeves).

You may **either** write your own optimization codes **or** find and use suitable (free) software on the world wide web.

You should only use software from the web for which you can obtain the source code, e.g. the uncompiled C++, to enable you to debug it if necessary and perhaps to modify it if it does not do exactly what is required.

Investigate the performance of the algorithms you have sourced or implemented on the Bird Function. Allow 1000 objective function evaluations in each algorithm run you execute. In analysing the performance of the algorithms, bear in mind the comments made on performance measures in the *Common Issues* section of the lecture notes.

Investigate the effects on the performance of your algorithms of varying some of the parameters or implementation options that control them; for instance, for a Genetic Algorithm investigate the effect of different population sizes/number of generations, or different crossover and mutation probabilities, or different selection schemes etc. An exhaustive exploration of all possibilities is not required, just some examination of the way in which the control parameters affect algorithm performance.

If you are using a code you have found, as opposed to one you have written, rather more investigations will be expected.

In these studies the same set of random number generator seeds should be used in each set of runs.

For population-based search methods (i.e. Genetic Algorithms and Evolution Strategies), the initial populations should be generated by random sampling; initial populations should not be "seeded" with solutions close to the optimum.

For search methods that start from a single solution, the initial solution should be randomly generated and varied from run to run.

Final Report

Write a report detailing and analyzing the results of your investigations. This report should be placed in the EIETL post-box by 5pm on the first day of Lent Full Term, **Tuesday 13th January 2015**.

Make sure that you put a coursework coversheet on your report. This can be downloaded from http://teaching.eng.cam.ac.uk/information/labs-coursework/part-iib/content.

Make sure that your name does not appear in the report itself or the code listings.

The report should include (in Appendices) listings of the source code of the algorithms you have implemented or found. The listings of codes you have written should be reasonably well commented. The source (e.g. the url) of any found codes should be clearly identified.

Detailed descriptions of the ways in which the algorithms work are not required. If you have implemented an idea of your own (which is by no means discouraged) or used a feature not described in the 4M17 lecture notes, perhaps one included in a code you found, the basic principles of this should be explained in your report.

Your report should discuss problem-specific implementation details (if any).

You should include in your report figures showing representative examples of the search patterns followed in (x_1, x_2) space by the two optimization methods you have tested.

The main focus of your report should be an evaluation of the performance of the two methods you have applied to the Bird Function, and a discussion of the effects on this performance of changing the algorithms' control parameters or implementation details.

Practical Issues

Demonstrator Help

Apart from the two scheduled coursework sessions you will be expected to work independently.

Questions related to the methods being investigated should be directed in the first instance to Dr Geoff Parks (email: gtp10@cam).

Questions about the final report or any other logistical matters should be directed to Dr Geoff Parks (email: gtp10@cam).

Computer Usage

As the objective function is a simple mathematical function, the only constraints are bounds and the number of iterations allowed in each run is quite small, the running of the algorithms you write or find should not be very expensive computationally and you should be able to do so on-line on the CUED Teaching System without offending anyone. Obviously, if you have access to good computing facilities of your own or through your College, it may be more convenient to use these.

When you come to investigate the effects of changing the control parameters of your algorithms and need to do series of runs where only the random number seeds are being changed, you may find it more convenient to run these using a script. For specific advice/ instructions you are referred to the relevant page on the CUED help system:

www-h.eng.cam.ac.uk/help/unix/LongRunningPrograms/

Summary

Key Dates

Wednesday 26 November 2014	10–11am	Coursework Introduction	LR5
Thursday 27 November 2014	1–2pm	Coursework Session	DPO
Wednesday 3 December 2014	10–11am	Coursework Session	DPO
Tuesday 13 January 2015	5pm	Report Submission Deadline	EIETL post-box