Norwegian University of Science and Technology
Department of Engineering Cybernetics

Exercise 9
Optimization Algorithms

Problem 1 (50 %) Unconstrained Optimization

This problem is about minimizing the Rosenbrock function

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \tag{1}$$

a Solve Problem 3.1 in the textbook using MATLAB. Comment on your results. What does the condition in the loop of Algorithm 3.1 (page 37) ensure?

b Modify your code to use the BFGS method (the most common Quasi-Newton method), instead of using the exact Newton direction. Use Algorithm 6.1 (p. 140). Compare your results to the two algorithms developed in a) and comment.

Attach plots showing $f(x_k)$ as a function of k, α_k as a function of k, and a plot that shows how x_k moves in the x_1-x_2 plane relative to isocurves for $f(x)$. You can use the MATLAB file plot_iter_rosenbrock.m published online for the isocurve plot. Also attach printouts of all three codes to your homework.

c Study the plots that show α_k at each iteration k for the Newton and BFGS algorithms. How common is a step lenght of $\alpha_k = 1$? In particular, what is the step length close to the solution? How does this agree with the convergence-rate theory in Chapter 3.3?

Problem 2 (5 %) Cholesky Factorization

The Newton direction can be modified using modified Cholesky factorization (p. 52 in the textbook). In which situations is it desirable to modify the search direction and what can be gained?

Problem 3 (15 %) Gradient Calculation

This problem is about calculation of gradients for

$$f(x) = 100(x_2 - x_1)^2 + (1 - x_1)^2 \tag{2}$$

Note that this is *not* the Rosenbrock function. See Chapter 8 in the textbook for more on approximating derivatives.

a Derive an approximation of the gradient of $f(x)$ using the forward-difference scheme (see equation (8.1) in the textbook).

b Calculate $\nabla f(x)$ analytically. Then, calculate $\nabla f(x)$ for $x = [0.5,\ 0.5]^\top$ and $x = [1,\ 1]^\top$ using three different values of ϵ for the approximation. Compare the approximations to the analytical gradient and discuss the results.

c Discuss the error associated with the forward-difference approximation of $\nabla f(x)$.

Problem 4 (30 %) The Nelder-Mead Method

In this problem, the Rosenbrock function (1) will be minimized using the Nelder-Mead algorithm (a derivative-free method, see Section 9.5 in the textbook). To get a more intuitive idea of the Nelder-Mead method, this animation, showing the algorithm used on the Rosenbrock function, might be worth a look (the animation can be found on the Wikipedia page about the Nelder-Mead algorithm).

a The Nelder-Mead method needs $n+1$ starting points for an n-dimensional function. What are the conditions on the $n+1$ initial points? Give an example of an invalid set of starting points. Describe in geometric terms the difference between valid and invalid sets of starting points for a function of two variables. Make sure you understand why this requirement is imposed.

b Study the code nelder_mead.m posted on Blackboard and make sure you understand how it is related to Procedure 9.5 on page 238. Run the code with different initial points, look at the output and try different values for the parameters `fun_tol` and `x_tol`. Depending on the starting point you give the function, you may have to change the parameter `iterlim` as well. The code can be called like this:

```
x0 = [2, -2]'; % initial point
[x, fval, x_iter] =  nelder_mead(x0, 'report');
```

c Use $x = [1.2,\ 1.2]^\top$ as the initial point (this point is close to the optimum $x^* = [1,\ 1]^\top$) and run the algorithm. Then start the algorithm from $x = [-1.2,\ 1]^\top$, which is a more difficult starting point. Study the output and the plots. Discuss the shape of the curve illustrating the average value of $f(x)$ at each iteration. Plot the iterates in the x_1-x_2 plane relative to isocurves for $f(x)$. (Again, you can use the MATLAB file plot_iter_rosenbrock.m published online.)

d Compare your results from c) with Problem 1. (You can measure the time an algorithm uses in MATLAB using the commands `tic` and `toc`, but this is not the most relevant means of comparison.)

e Solve the first half of Problem 9.12 on page 244 in the textbook. (That is, ignore the last sentence of the problem.)