MTRN4030

Optimisation Methods for Engineering Systems

Laboratory Exercise 2 – weeks 2, 3, 5

Classical Optimization Methods

Summary

This laboratory exercise is scheduled for three weeks. The first week is a tutorial meeting that tutors will provide a revision on the topic and guidelines to complete this exercise.

The exercise includes a program based implementation of line search and gradient optimization methods including Newton's method and the steepest descent method. The implementation details of these methods will be investigated and their characteristics and/or limitations are to be revealed and compared. This exercise also contains the development of a graphical search method to obtain the optimal setting of simple mechanical system parameters. The last part is about the use of the Matlab optimization toolbox to solve the design of the simple mechanical system.

An individual written report is required for this exercise, which carries 8% (of the total course marks), in additional to the marks for the practical work which is 5%. Report submission is to be uploaded to Moodle. Details of the requirements are given in the Assessment section.

Part 1 – Line Search and Gradient based optimization

Exhaustive search is a simultaneous search method in which all the experiments are conducted before any judgment is made regarding the location of the optimum point. This method is very inefficient. The dichotomous search method, as well as the Fibonacci and the golden section methods, are sequential search methods in which the result of any experiment influences the location of the subsequent experiment.

In the dichotomous search, two experiments are placed as close as possible at the centre of the interval of uncertainty. Based on the relative values of the objective function at the two points, almost half of the interval of uncertainty is eliminated.

The principle of gradient-based optimization methods is based on finding the root of the gradient of the objective function. This is because by setting the gradient to zero is a condition for a function to be at its minimum or maximum.

When the objective function is continuous, it can be approximated in a Taylor series, which contains terms of the derivatives of the function. The gradient terms are then extracted, set to zero and solved for the variables in the objective function. The Hessian matrix can be formulated from the gradients and Newton's method produces the optimum solution in an iterative loop.

While calculating the Hessian may be complicated, the steepest descent method makes use of only the gradient information. In particular, the adjustment on the iterative temporary solution is based on assigning the change as the negative of the gradient.

Task 1

1. Let the reduction ratio in the golden search method be ρ. Find the specific choice of the golden section ratio in order to reduce the number of objective function evaluations. For a given search range Δ and the accepted tolerance τ, show how the number of iterations N can be obtained.

2. Given the objective function $f(\boldsymbol{x})$ of an optimization problem, assume it is continuous where derivatives exist, give a Taylor series expansion about a point $\boldsymbol{x}_n$ up to the quadratic (2-nd derivative) term.

3. Obtain the derivative of the series with respect to the infinitesimal change $\Delta \mathrm{x} = \boldsymbol{x} - \boldsymbol{x}_{\mathrm{n}}$, and give the expression in terms of the Hessian matrix.

4. Give the iterative equation to obtain the optimum solution $\boldsymbol{x}^*$ in the form of Newton's method.

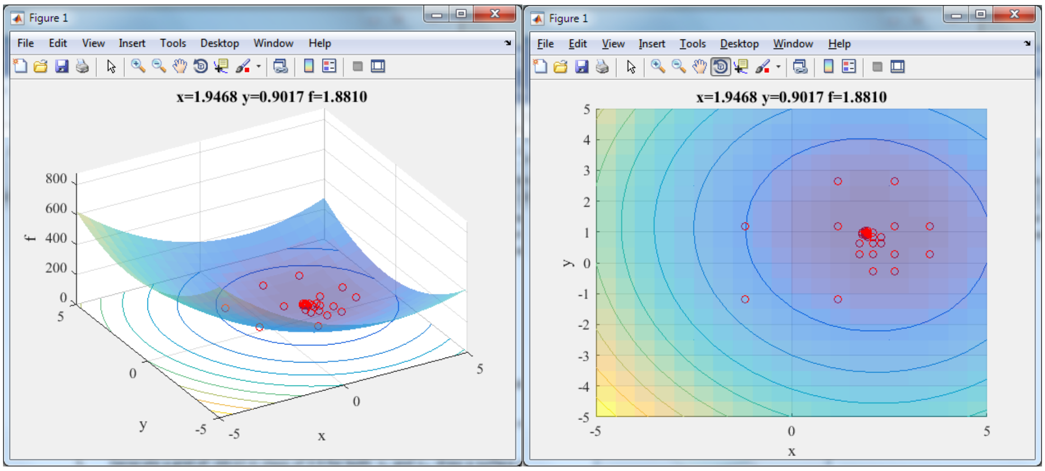
5. Formulate the steepest descent optimization algorithm in an iterative expression including a control coefficient α on the step size.

   Put your answers in the written report.

Task 2

1. Given an objective function $f(x, y) = 10(x-2)^2 + xy + 10(y-1)^2$, write a Matlab program to show the function landscape. Find the minimum, with a user-specified tolerance of $\tau = 0.01$, by using the Golden section method. You can set the search range as $x \in [-5,5]$, $y \in [-5,5]$.

2. You can illustrate your program output as:



3. Verify your results by calculating the gradients, $\partial f/\partial x, \partial f/\partial y$; then solve for the optimum design variables x^*, y^*, based on the first-order necessary condition, $\nabla f = 0$. Verify that the solution is a minimum by the second-order necessary condition $\nabla^2 > 0$, i.e., the Hessian is positive definite. Put your answer in the written report.

Task 3

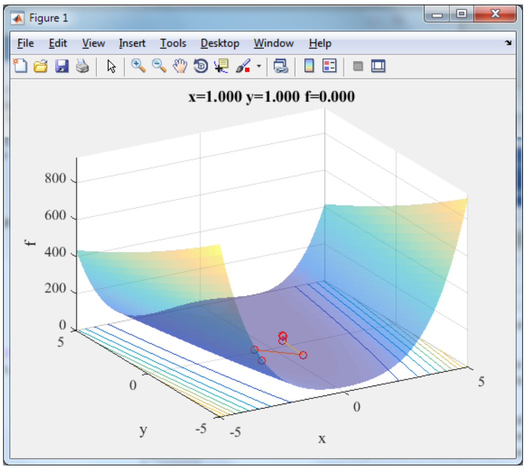
1. Given an objective function $f(x, y) = (y - x^2)^2 + (1 - x)^2$, write a Matlab program to perform the following. (Hint: the min. is at $x = 1, y = 1$ and $f(x^*, y^*) = 0$).

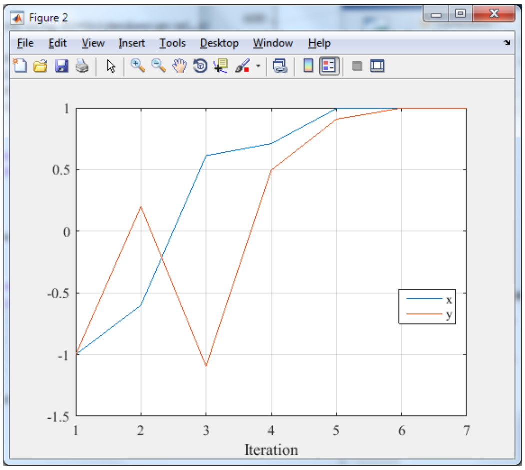
2. Use symbolic commands to calculate the gradient vector $\boldsymbol{G}$ and the Hessian matrix ($\boldsymbol{H}$), and its inverse $\boldsymbol{H}^{-1}$.

3. Generate a grid of $x \in [-5,5], y \in [-5,5]$ in steps of 0.1, draw a surface plot of the function. Determine if the function is convex or not, verify it with the Hessian matrix characteristics.

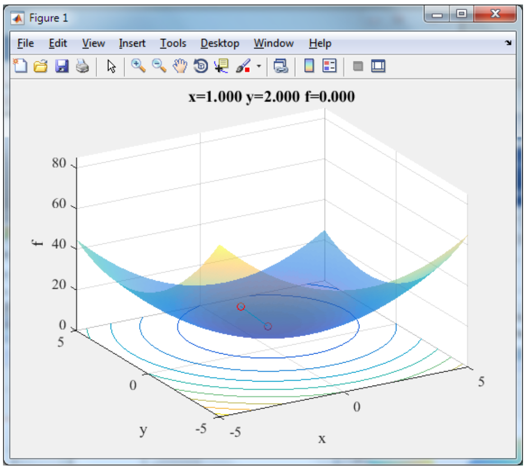
4. Use the initial guess of the solution $\boldsymbol{x}_0 = [-1, -1]^T$, construct an iterative loop for Newton's method and terminates when the change of solution (with respect to the previous iteration) is less than a user-specified tolerance, e.g. $\tau = 0.01$.

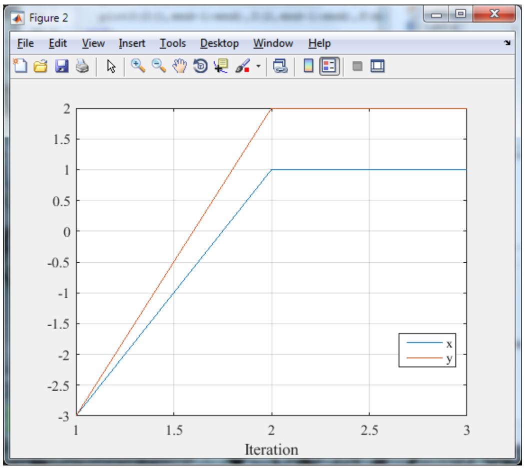
5. Plot the solutions on the objective function surface and a plot against the iteration count. Your plots may look like the figures below.





6. Change the objective function to $f(x, y) = (x - 1)^2 + (y - 2)^2$ and search for the optimum solution again. The resultant plots, from an initial guess at $x = -3, y = -3$, may be as shown below.





7. Observe that the solution is obtained after only one iteration (excluding the initial solution), i.e., iteration 2 as shown). Explain why it is possible. (Hint: inspect the form of the objective function, gradient matrix and/or the Hessian matrix.) Put your answer in the written report.

Part 2

Mechanical System Design - 1

The shear stress induced along the z-axis when two spheres are in contact with each other is given by

$$\frac{\tau_{zx}}{p_{max}} = \frac{1}{2}\left[\frac{3}{2\left\{1+\left(\frac{z}{a}\right)^2\right\}} - (1+\nu)\left\{1-\frac{z}{a}\tan^{-1}\left(\frac{1}{z/a}\right)\right\}\right]$$

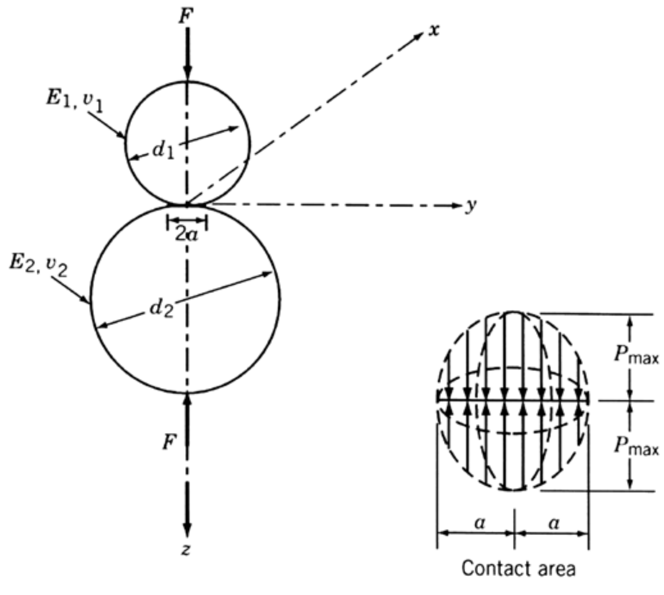
where a is the radius of the contact area and p_{max} is the maximum pressure developed at the centre of the contact area, and

$$a = \left\{\frac{3F\left[\frac{1-\nu_1^2}{E_1}+\frac{1-\nu_2^2}{E_2}\right]}{8\left(\frac{1}{d_1}+\frac{1}{d_2}\right)}\right\}^{1/3},\ p_{max} = \frac{3F}{\pi a^2}$$

where F is the contact force, E_1 and E_2 are Young's moduli of the two spheres, ν_1 and ν_2 are Poisson's ratios of the two spheres, and d_1 and d_2 the diameters of the two spheres. In many practical applications, such as ball bearings, when the contact load F is large, a crack originates at the point of maximum shear stress and propagates to the surface, leading to a fatigue failure. To locate the origin of a crack, it is necessary to find the point at which the shear stress attains its maximum value. Formulate the problem of finding the location of maximum shear stress for $\nu = \nu_1 = \nu_2 = 0.3$. Then we have

$$f(\lambda) = \frac{0.75}{1+\lambda^2} + 0.65\lambda\tan^{-1}\frac{1}{\lambda} - 0.65$$

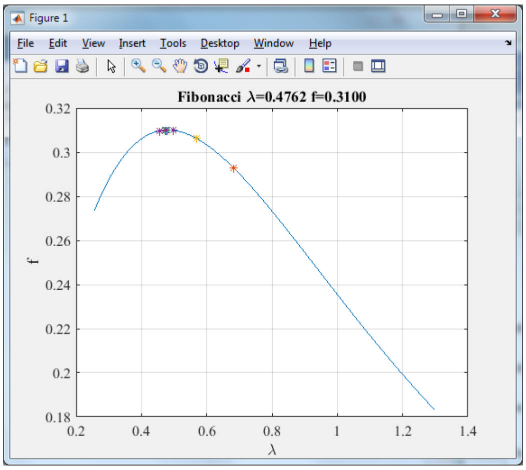
where $f = \tau_{zx}/p_{max}$ and $\lambda = z/a$, and the objective is to maximize $f(\lambda)$.

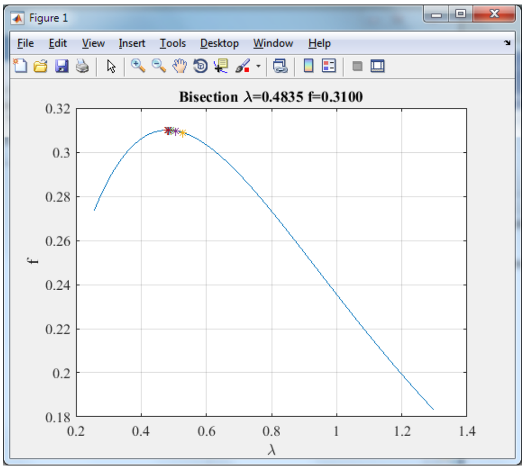


Task 4

Write a Matlab program to implement the following.

1. Use the 'bracketing' method to find the search ranges for Golden Section and Fibonacci methods. (Hing: start with a small λ, e.g. 0.01).
2. Implement the Fibonacci Section Search method to obtain λ where it attains the maximum $f(\lambda)$, use a tolerance of $\tau = 0.01$.
3. Implement the Bisection Search method to obtain λ where it attains the maximum $f(\lambda)$, use a tolerance of $\tau = 0.01$.
4. Your program may produce the following figures.

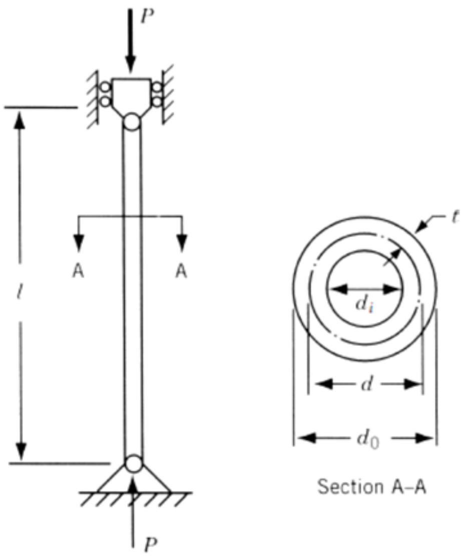




5. Compare the number of iterations needed for these methods, and the maximum values obtained. Comment on their overall implementation complexities. Put your discussion in the written report.

Mechanical System Design - 2

Consider the design a uniform column of tubular section, with hinge joints at both ends, to carry a compressive load P=2500 kg_f for minimum cost. The column is made up of a material that has a yield stress (σ_y) of 500 kg_f/cm^2, modulus of elasticity (E) of 0.85×10^6 kg_f/cm^2, and weight density ρ of 0.0025 kg_f/cm^3. The length of the column is 250 cm. The stress induced in the column should be less than the buckling stress as well as the yield stress. The mean diameter of the column is restricted to lie between 2 and 14 cm, and columns with thicknesses outside the range 0.2 to 0.8 cm are not available in the market. The cost of the column includes material and construction costs and can be taken as $5W + 2d$, where W is the weight in kilograms force and d is the mean diameter of the column in centimetres.



The design variables are the mean diameter d and tube thickness t,

$$\boldsymbol{x} = [x_1, x_2]^T = [d, t]^T.$$

The objective function to be optimized is

$$f(\boldsymbol{x}) = 5W + 2d = 5\rho L\pi dt = 9.82x_1x_2 + 2x_1.$$

The behaviour constraints can be expressed as stress induced be less than yield stress, and stress induced be less than buckling stress. The induced stress is given by

$$\sigma_i = \frac{P}{\pi dt} = \frac{2500}{\pi x_1 x_2}$$

The buckling stress for a pin-connected column is given by

$$\sigma_b = \frac{\pi^2 EI}{l^2}\frac{1}{\pi dt}$$

where the second moment of area of the cross section of the column is

$$I = \tfrac{\pi}{8}x_1x_2(x_1^2 + x_2^2)$$

The side constraints are given by

$$2 \le d \le 14,\ 0.2 \le t \le 0.8$$

The design objective is to minimize the cost, $f(\boldsymbol{x})$, such that the constraints in stresses σ_i, σ_b, and sides d, t are satisfied. These include

$$g_1(\boldsymbol{x}) = \frac{2500}{\pi x_1 x_2} - 500 \le 0$$

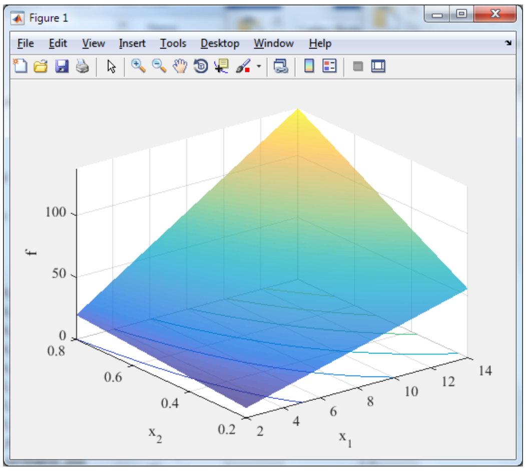
$$g_2(\boldsymbol{x}) = \frac{2500}{\pi x_1 x_2} - \frac{\pi^2(0.85 \times 10^6)(x_1^2 + x_2^2)}{8 \times 250^2} \le 0.$$

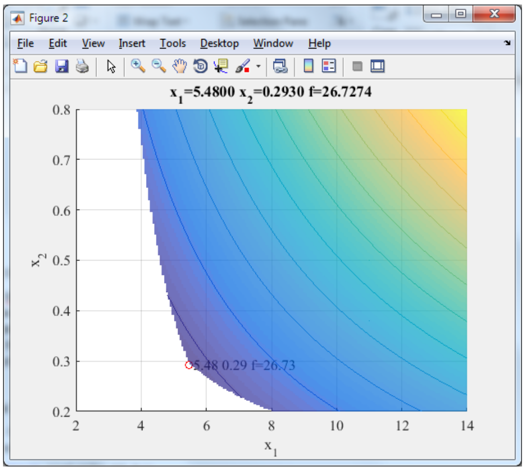
Task 5

Write a Matlab code to carry out an optimum design of the given mechanical system by use of a graphical/search approach.

The search method is based on calculating the objection function over a set of grid points. Then discard those points which do not satisfy the constraints. Finally, the solution is determined by identifying the grid point that gives the smallest objective function value.

1. Define the number of grid points, and specify the ranges according the size constraints.
2. Generate a set of grid points (use the 'meshgrid' command), calculate and plot the objective function surface with its contour.
3. Calculate the constraints impose by the limits on the stress.
4. Identify and discard the grid points that do not satisfy the constraints (use the 'find' command and set the identified grid point surface to non-numeric, i.e., 'NaN').
5. Re-plot the objective function surface on another graph.
6. Find the minimum among the evaluated grid point values (use the 'min' command with the minimum value and indices as outputs, and convert to grid points by the 'ind2sub' command).
7. Annotate the surface plot with the solution (use the 'text' and 'sprintf' commands).
8. Evaluate the effect of the number of grid points, hence resolution, on the accuracy of the result.





Assessment

This laboratory exercise carries 13% of the overall marks for the course. The practical work carries 5% and the written report carries 8%, to be submitted in Moodle.

The **practical work** (tasks 2 – 6) is assessed according to the following criteria:

Absent	0%
Work less than 1/4 completed	1%
Work less than 1/2 completed	2%
Work more than 1/2 completed	3%
Work more than 3/4 completed	4%
Work all completed	5%

At the end of the scheduled laboratory period, the work (code and graphs) has to be marked-off by tutors. A signed copy by tutors of the mark sheet (download from Moodle) should be kept by the student and upload to Moodle within the week when marks are obtained. Note: Time has to be allowed for tutors to mark the exercises. Students are expected to register with the tutor for their order in the marking process.

The **written report** is assessed according to the following criteria:

Task 1, derivation of equations leading to Golden Section Search, Newton's and steepest descent methods (a logical flow of reasoning/derivation/proofs is required) 2%

Task 2, verification of the solutions with respect to the characteristics of the gradient vector and the Hessian matrix. 2%

Task 3, discussion on form of the objective function, and how it affects the optimization process (evolution of the design variables through iterations). 2%

Task 4, comparison of the number of iterations needed, optimality of results obtained, and the implementation complexity. 1%

Task 5, comment on the effect of grid resolution (related to the number of grid points) on the solution accuracy (you can use a table to illustrate the relationship). 1%

In addition to the above requirements, the written report should be properly structured (use of section and sub-section headers, appropriate fonts – 12pt max, and proper layout of diagrams/graphs).

Submit your written report to Moodle **on/before midnight Mon week 6**. Note: If you have to use graphics/plots, press 'alt'+'prt sc' on your key board to capture a screen shot of the Matlab figures, then 'paste' on the MS Word file, and convert the whole document into a pdf file for submission. Make sure that the files size is within 5MB.