T. Kärkkäinen
Course: Application Programming in MATLAB Environment 2000

Exercises 5–6

Illustration of the basics of optimization by using and monitoring built-in optimization routines of MATLAB.

Cost functionals for the following problems read as:

$$\begin{aligned} \mathcal{J}_1(x,y) &= 100*(y-x^2)^2+(1-x)^2 \quad \text{(so-called } \textit{Rosenbrock's} \text{ function)}, \\ \mathcal{J}_2(x,y,z) &= 100*(2*z-y^2-x^2)^2+(1-y)^2+(1-x)^2, \\ f(x) &= \frac{\sin(x^2)}{x}. \end{aligned}$$

You can probably figure out by yourself the exact minimizers of, at least, functions $\mathcal{J}_1$ and $\mathcal{J}_2$.

Problem 1

Write m-files which can be used to compute the values of given functions also in the case when coordinate matrices are given as inputs (similarly to built-in functions of MATLAB). Using the corresponding m-file, plot function $f(x)$.

Problem 2

Minimize the given functions using MATLAB commands `fmin` and `fmins`. Use commands `tic` and `toc` to measure the amount of CPU time that was needed for solving the problems. Compare the solution from `fmin` to the plot of $f(x)$.

Problem 3

Illustrate function $\mathcal{J}_1$ using a surface plot and point out the location of the minimizer in this image. Compute the numerical gradient for function $\mathcal{J}_1$ using the `gradient` command. Find the gradient vector of minimal length and print out the x and y coordinates of the corresponding point. What do you notice?

Problem 4

Seek from the MATLAB *Optimization Toolbox* a suitable routine, which you can use to minimize functions $\mathcal{J}_1$ and $\mathcal{J}_2$. Use it and compare the obtained results to the previous ones by means of the solution and the CPU time. Make a surface plot of function $\mathcal{J}_1$ and add to this image the route that the chosen routine has followed when seeking for the minimum. Finally, draw another figure which contains the corresponding history of function values during the minimization.

Problem 5 (If you still have some time left)

Make a grid on the interval $[0.1, 4]$ with the distance 0.01 between the grid points. Plot function $f(x)$ and its derivative (analytic or numerical computed using `diff` command). Program your own MATLAB function which seeks the strict local minima and maxima of $f(x)$ using the observation that for vector $\mathbf{f}$ containing the grid values of function $f(x)$ it holds in the local minimum: $\mathbf{f}_i < \min(\mathbf{f}_{i-1}, \mathbf{f}_{i+1})$. Point out the obtained local critical values and also global minimum and maximum from the function plot in a "suitable way" (up to you, you can use the format of the corresponding pictures in the lecture notes). Seek then these sama locally critical values using the derivative and illustrate them as you wish. Here you may find commands `find` and `abs` useful.