EN.550.661: Nonlinear Optimization I

Instructor: Daniel P. Robinson

Homework Assignment #4

Starred exercises require the use of MATLAB.

Exercise 4.1: Using the language of mathematics, describe how you would find the Cauchy point for a second-order model

$$m_k(s) = f_k + s^T g_k + \tfrac{1}{2} s^T B_k s$$

of $f(x_k + s)$ within the trust-region $\|s\| \le \delta_k$.

Exercise 4.2: Solve the trust-region subproblem

$$\underset{s \in \mathbb{R}^n}{\text{minimize}} \quad s^T g + \tfrac{1}{2} s^T B s \quad \text{subject to} \quad \|s\|_2 \le \delta \tag{1}$$

in the following cases:

(a)

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad g = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad \delta = 2,$$

(b)

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad g = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad \delta = 5/12$$

Hint: $\lambda = 2$ is a root of the nonlinear equation

$$\frac{1}{(1+\lambda)^2} + \frac{1}{(2+\lambda)^2} = \frac{25}{144}.$$

(c)

$$B = \begin{pmatrix} -2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \quad g = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad \delta = 5/12,$$

(d)

$$B = \begin{pmatrix} -2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \quad g = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad \delta = 1/2, \quad \text{and}$$

(e)

$$B = \begin{pmatrix} -2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \quad g = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad \delta = \sqrt{2}.$$

Exercise 4.3: Sketch the solution of problem (1) with data

$$B = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix} \quad \text{and} \quad g = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

as a function of the trust-region radius δ, i.e., plot $s(\lambda)$ as discussed in class. In which direction does the solution point as δ shrinks to zero? How does the Lagrange multiplier for the trust-region constraint depend on the trust-region radius? For what value of the trust-region radius does the solution become unconstrained?

Exercise 4.4*: Write a MATLAB m-function called steihaug_CG.m that implements the truncated linear conjugate gradient method by Steihaug based on Algorithm 4 in the Lecture 06 slides. The function call should have the form

```
[ p, iters, flag ] = steihaug_CG( B, g, radius, tol )
```

where the input `B` is required to be a symmetric (possibly indefinite) matrix, `g` is a vector, `radius` $>$ 0 is the trust-region radius, and `tol` $\in (0,1)$ is the stopping tolerance. On exit, the resulting approximate Steihaug truncated-CG solution should be stored in the vector `p`, the number of iterations performed stored in `iters`, and the parameter `flag` should be set to one of the following values: -1 if the algorithm terminated because of negative curvature, 0 if the stopping tolerance was met, and 1 if the algorithm returned a boundary solution simply because the CG iterations grew larger than the trust-region radius.

Exercise 4.5*: Consider the problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x)$$

where f is a twice continuously differentiable function.

(a) Write a MATLAB m-function called `unc_TR.m` that implements a trust-region method with trial steps computed from the Steihaug truncated linear CG method as given in Problem 4.4 above. The function call should have the form

```
[x,F,G,H,iter,status] = unc_TR(fun,x0,maxit,printlevel,tol)
```

where `fun` is of type *string* and represents the name of a Matlab m-function that computes $f(x)$, $\nabla f(x)$, and $\nabla^2 f(x)$ for some desired function f; it should be of the form

```
[F,G,H] = fun(x)
```

where for a given value x it returns the values of the function, gradient, and Hessian, respectively. The parameter `x0` is an initial guess at a minimizer of f, `maxit` is the maximum number of iterations allowed, `printlevel` determines the amount of printout required, and `tol` is the final stopping tolerance. In the code, if the parameter `printlevel` has the value zero, then no printing should occur; otherwise, a single line of output is printed (in column format) per iteration. On output, the parameters `x`, `F`, `G`, and `H` should contain the final iterate, function value, gradient vector, and Hessian matrix computed by the algorithm. The parameter `iter` should contain the total number of iterations performed. Finally, `status` should have the value 0 if the final stopping tolerance was obtained and the value 1 otherwise.

(b) Write a separate MATLAB m-file with function declaration `[F,G,H] = fun(x)` that returns the value F, gradient G, and Hessian H at the point $x \in \mathbb{R}^2$ of the function

$$f(x) = 10(x_2 - x_1^2)^2 + (x_1 - 1)^2.$$

Use your m-function `unc_TR.m` from part (a) to minimize f with starting point $x_0 = (0, 0)$.