Numerical analysis
Midterm Project, 2025.11.4
Due to 2025.11.11. Please submit the pdf file of solutions to the email,
jmwu@gms.ndhu.edu.tw

1. (160 points) Let

$$\left\{\begin{array}{rcc} f_1(x_1,x_2,x_3) &=& 3x_1 - cos(x_2x_3) - \frac{1}{2} \\ f_2(x_1,x_2,x_3) &=& x_1^2 - 81(x_2+0.1)^2 + sin(x_3) + 1.06 \\ f_3(x_1,x_2,x_3) &=& e^{-x_1x_2} + 20x_3 + \frac{1}{3}(10\pi - 3) \end{array}\right\},$$

A. Write a Matlab function, F = myfun(x), to evaluate the following nonlinear functions,

$$F(x_1,x_2,x_3) = \begin{bmatrix} f_1(x_1,x_2,x_3) \\ f_2(x_1,x_2,x_3) \\ f_3(x_1,x_2,x_3) \end{bmatrix}$$

where x denotes an input vector containing elements x_1, x_2 and x_3, and F is an vector containing output values of three coordinate functions f_1, f_2 and f_3.

B. Use Matlab function fsolve to solve the following nonlinear system by the Levenberg–Marquart algorithm,

$$F(x_1,x_2,x_3) = \mathbf{0}$$

C. Let $J(x_1,x_2,x_3)$ denote the Jacobian matrix, where the joint element of row i and column j is $\frac{df_i}{dx_j}$. Use Matlab function jacobian to build an inline function for evaluation of $J(x_1,x_2,x_3)$.

D. Let $E(x_1,x_2,x_3) = \sum_{i=1}^{3} f_i^2(x_1,x_2,x_3)$. Derive $\frac{dE}{dx_j}$.

E. Let

$$g(x) = \frac{dE}{dx} = \begin{bmatrix} \frac{dE}{dx_1} \\ \frac{dE}{dx_2} \\ \frac{dE}{dx_3} \end{bmatrix}.$$

Express $g(x)$ in terms of $F(x_1,x_2,x_3)$ and $J(x_1,x_2,x_3)$.

F. Show the expression in problem 1E.

G. Express the updating rule of minimizing $E(x)$ by the gradient descent method.

H. Let $l(x_1,x_2) = x_1x_2$. Illustrate the difference between $\frac{\partial^2 l}{\partial x_1 \partial x_2}$ and $\frac{dl}{dx_1}\frac{dl}{dx_2}$

I. Let x denote collection of x_1, x_2 and x_3. Let

$$L(x+\Delta x) = E(x) + g^T\Delta x + \frac{1}{2}\Delta x^T H \Delta x$$

approximate $E(x+\Delta x)$, where Δx denotes a small change, g denotes the gradient in problem 1E and H denotes the Newton–Gauss Hessian. Express H in terms of $g(x)$.

J. Express the updating rule of minimizing $E(x)$ by the Newton–Gauss method.

K. Express the updating rule of minimizing $E(x)$ by the Levenberg–Marquardt method.

L. (30 points) Illustrate the flow chart in figure 1 for explaining minimization of $E(x)$ by the Levenberg–Marquardt method. State details of the halting condition.

M. Describe the relation between the Levenberg–Marquardt method and the gradient descent method.

N. Describe the relation between the Levenberg–Marquardt method and the Newton–Gauss method.

2. (80 points) Write Matlab codes to implement the flow chart in problem 1L and give the numerical results.

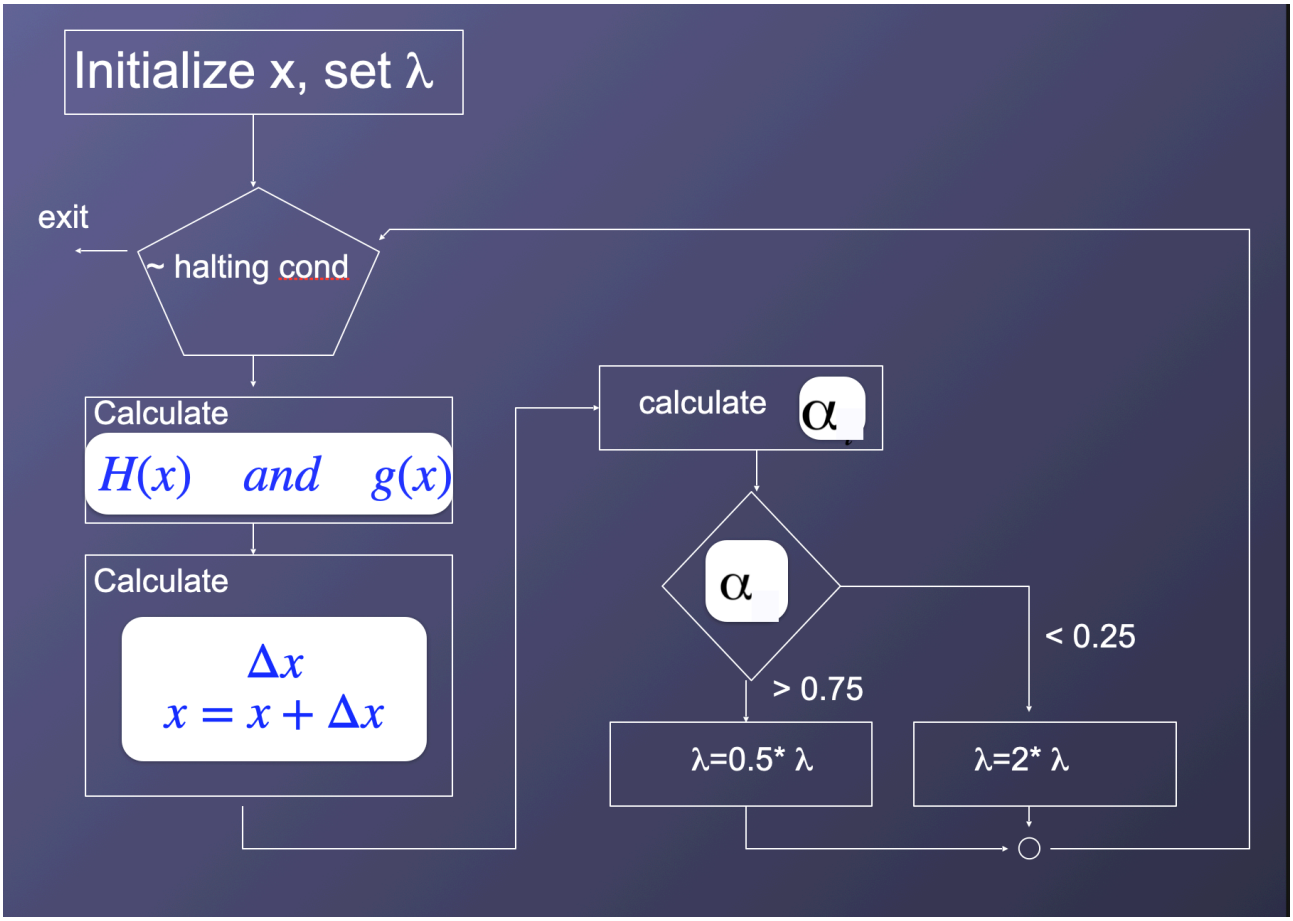

Figure 1

3. (80 points) Figure 2 shows the conjugate gradient algorithm. Let p_k denote the searching direction, α_k denote the weight and x_k denote the state at iteration k. Following the CG algorithm in figure 2, $x_k = \sum_{i=1}^{k} \alpha_i p_i$. Let A be symmetric and positive definite. Further assume $\langle p_k, p_j \rangle_A \equiv p_k^T A p_j = 0$ for $j \neq k$.
   A. State the problem that is exactly solved by the conjugate gradient algorithm in figure 2.
   B. Explain why iteratively executing equations (1–5) can solve the problem stated in problem 3A.
   C. State derivation of $x_k = x_{k-1} + \alpha_k p_k$ (eq 2), where $\alpha_k = \dfrac{\langle p_k, b \rangle}{\langle p_k, p_k \rangle_A}$.
   D. State derivation of $r_{k+1} = r_k - \alpha_k A p_k$ (eq 3), where

$$\beta_k = -\frac{\mathbf{r}_{k+1}^{\mathsf{T}} \mathbf{A} \mathbf{p}_k}{\mathbf{p}_k^{\mathsf{T}} \mathbf{A} \mathbf{p}_k}$$

   E. State derivation of $p_{k+1} = r_{k+1} + \beta_k p_k$ (eq 4) and
   F. (15 points) State derivation of

$$\beta_k := \frac{\mathbf{r}_{k+1}^{\mathsf{T}} \mathbf{r}_{k+1}}{\mathbf{r}_k^{\mathsf{T}} \mathbf{r}_k} \qquad \text{(eq 5)}$$

   G. (15 points) State derivation of $\alpha_k := \dfrac{\mathbf{r}_k^{\mathsf{T}} \mathbf{r}_k}{\mathbf{p}_k^{\mathsf{T}} \mathbf{A} \mathbf{p}_k}$ (eq 6)

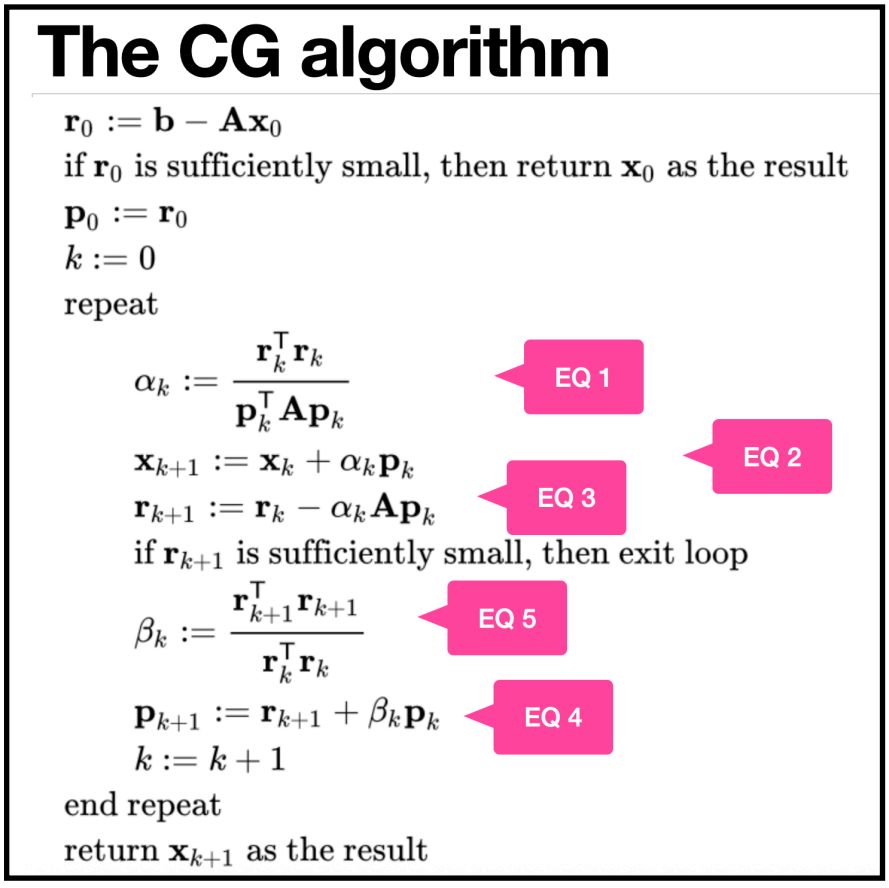

Figure 2