Geomatics Engineering

ENGO 361: Least Squares Estimation

Lab TWO

Parametric Least Squares Adjustment

Ahmed A. Youssef

ahmed.youssef1@ucalgary.ca

UNIVERSITY OF CALGARY

SCHULICH School of Engineering

Mobile **M**ulti-**S**ensors **S**ystems Research Group

January 2018

ENGO361 – Lab TWO

Due Date: Thursday, March 1st 2018, at 11:59 PM

➢LAB OVERVIEW

- ➢ **Part One**: Parametric Least Squares Adjustment for General Estimation Problems
  - Problem No. 1: Use Parametric Least Squares Adjustment for Curve Fitting, for two different types curves. Go through estimation process, and understand and discuss the computational process in relation to the quality of the estimation.

- ➢ **Part Two**: Parametric Least Squares Adjustment for measurements.
  - Problem No. 2: Use Parametric Least Squares for adjustment of a Levelling Network. You are required to write a C++ program to perform the estimation process. You need to understand and discuss the results from your program.

Part ONE: General Estimation - Curve Fitting
Problem ONE

- **Given:**
  - A set of 2D coordinates (k , y)
  - k values are error free values (i.e., **constants**).
- **Requirements:**
  I. Use parametric least squares to fit the given data points to:

k	y
-1.5608	-2.6285
-1.4137	-2.0562
⋮	⋮
1.5708	-2.4057

a. $y = ak^2 + bk + c$ Parabola Equation

b. $y = \mathrm{a}\cos(2k) + b$ Sinusoidal Equation

For both curves (a), and (b), determine the following:

i. Point of Expansion (POE)

ii. Design Matrix (A), Weight Matrix (P), Normal Matrix (N).

iii. Normal Vector (u) of the 1st iteration

iv. Vector of Estimated Parameters ($\hat{x}$)

v. Vector of Estimated Residuals ($\hat{v}$)

vi. $\hat{v}^{\mathrm{T}}\mathrm{P}\hat{v}$ and $\hat{\sigma}_{o}^{2}$

vii. Variance Covariance Matrix of the estimated parameters ($C_{\hat{x}}$).

Part ONE: General Estimation - Curve Fitting
Problem ONE

- **Given:**
  - A set of 2D coordinates (k , y)
  - k values are error free values (i.e., **constants**).

k	y
-1.5608	-2.6285
-1.4137	-2.0562
⋮	⋮
1.5708	-2.4057

- **Requirements - Continued:**

II. Using Matlab, plot on the same figure, for curve (a), and (b):

i. The given data points (using marker points).

ii. The fitted curves (using line format)

- Curve for Parabola Equation Fitting
- Curve for Sinusoidal Equation Fitting

III. From the results of (I), for curves (a) and (b), which of the fitting functions better fits the given data? Why?

Part ONE: General Estimation - Curve Fitting Problem ONE

➢ **Solution:** **For Requirement (I) - curve (a)**

a. $y = ak^2 + bk + c$ Parabola Equation

Step (1):

Identify unknown parameters, measurements/observations, and constants

- Let unknown parameters to be:

$$\hat{x}_{3\times1} = [a \quad b \quad c]^T\text{, where } [.]^T \text{ is transpose of } [.]$$

- Let measurements to be:

$$l_{21\times1} = y_{21\times1}$$

- Let constants to be:

$$k_{21\times1}$$

Part ONE: General Estimation - Curve Fitting
Problem ONE

Step (2): Form Observation Equations

Step (2.a): Is the model linear or non-linear?

- The model is linear because the relation is between the measurements (y) unknown parameters (a, b, c). While (k) are constants in our estimation problem.

Step (2.b): Write Parametric Model Equations

- **Remember from Lab ONE:**
- The parametric/indirect model is written as:

Measurements = f(Unknowns, Constants) OR $\hat{l} = f(\hat{x}, c)$

Then:

For 1st Observation: $-2.6285 = (-1.5608)^2 a + (-1.5608) b + c$

For 2nd Observation: $-2.0562 = (-1.4137)^2 a + (-1.4137) b + c$

$\vdots \qquad \vdots = \vdots + \vdots + \vdots$

For 21st Observation: $-2.4057 = (1.5708)^2 a + (1.5708) b + c$

Part ONE: General Estimation - Curve Fitting Problem ONE

Step (2.c): Write Parametric Model in Matrix Form - CONTINUED

$$l_{n\times1} = A_{n\times u} * \hat{x}_{u\times1} - \hat{v}_{n\times1}$$

Where:

- $(n = 21)$ is the number of observations, and $(u = 3)$ is the number of unknown parameters.
- $(l_{n\times1} = l_{21\times1})$ is the observations vector.
- $(A_{n\times u} = A_{21\times3})$ is the design matrix for this linear model.
- $(\hat{x}_{u\times1} = \hat{x}_{3\times1})$ is the unknown parameters vector.
- $(\hat{v}_{n\times1} = \hat{v}_{21\times1})$ is the residuals vector.

Then:

$$\begin{bmatrix} -2.6285 \\ -2.0562 \\ \vdots \\ -2.4057 \end{bmatrix}_{21\times1} = \begin{bmatrix} (-1.5608)^2 & -1.5608 & 1 \\ (-1.4137)^2 & -1.4137 & 1 \\ \vdots & \vdots & \vdots \\ (1.5708)^2 & 1.5708 & 1 \end{bmatrix}_{21\times3} * \begin{bmatrix} a \\ b \\ c \end{bmatrix}_{3\times1} - \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{21} \end{bmatrix}_{21\times1}$$

Step (3): Find the Point of Expansion (POE)

Alternative A:

- Take the least number of equations required to give a unique solution, (i.e., for this example, we need 3 equations to solve for 3 unknowns)
- Consider the equations of the first 3 data points (k , y):

For 1st Observation: $-2.6285 = (-1.5608)^2 a + (-1.5608) b + c$

For 2nd Observation: $-2.0562 = (-1.4137)^2 a + (-1.4137) b + c$

For 3rd Observation: $-1.7129 = (-1.2566)^2 a + (-1.2566) b + c$

- We can solve these 3 equations to get (a, b, c), but this is unnecessary in this problem, because the model is linear.

Alternative B:

- Since the model is linear, then we can assume the initial values of the unknowns to be [**ZEROs**].

$\therefore x_0 = [0]_{3\times1}$ **OR** $\therefore a = 0$ $\therefore b = 0$ $\therefore c = 0$

Part ONE: General Estimation - Curve Fitting
Problem ONE

Step (4): Build the Design Matrix ($A_{n\times u}$)

- By Definition, the design matrix (A) is given as:

$$A = \frac{\partial f}{\partial x}\bigg|_{x=x_o}$$

- From the parametric model (Parabola Equation)

$$y = ak^2 + bk + c$$

$$\therefore \frac{\partial f}{\partial a} = \frac{\partial y}{\partial a} = k^2 \qquad \therefore \frac{\partial f}{\partial b} = \frac{\partial y}{\partial b} = k \qquad \therefore \frac{\partial f}{\partial c} = \frac{\partial y}{\partial c} = 1$$

$\therefore$ For One Equation: $A = [k^2 \quad k \quad 1]$

$$\therefore A_{n\times u} = \begin{bmatrix} k^2 & k & 1 \\ k^2 & k & 1 \\ \vdots & \vdots & \vdots \\ k^2 & k & 1 \end{bmatrix}_{21\times 3} = \begin{bmatrix} (-1.5608)^2 & -1.5608 & 1 \\ (-1.4137)^2 & -1.4137 & 1 \\ \vdots & \vdots & \vdots \\ (1.5708)^2 & 1.5708 & 1 \end{bmatrix}_{21\times 3}$$

- Same matrix we derived when we were writing the model in matrix form – slide 7.
- However, this is the systematic way to derive the design matrix, and you should follow this way, especially for **non-linear model**.

Part ONE: General Estimation - Curve Fitting
Problem ONE

<u>Step (5):</u> Establish Variance Covariance Matrix for the Observations (C_l)

- From the given:
  - ➢ All measurements have the same variances ($\sigma_l^2 = 1$)
  - ➢ All measurements are uncorrelated ($\sigma_{l_1 l_2} = \sigma_{l_1 l_3} = \sigma_{l_i l_j} = 0$)
- Remember:

➢ The covariance matrix is a square matrix, with size ($n \times n$), where (n) is the number of measurements.

➢ All diagonal elements are the variances

➢ All off-diagonal elements are covariances.

$$\therefore C_{l_{n\times n}} = \begin{bmatrix} \sigma_{l_1}^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ \vdots & \vdots & \\ 0 & 0 & \sigma_{l_{21}}^2 \end{bmatrix}_{21\times 21} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 \end{bmatrix}_{21\times 21}$$

$$\therefore C_{l_{n\times n}} = diag([1 \quad 1 \quad 1 \quad \ldots \quad 1]_{1\times 21}) \qquad \therefore C_{l_{n\times n}} = I_{21\times 21}$$

Identity Matrix

Step (6): Establish Weight Matrix (P)

- Remember: (σ_o^2) is the proportionality constant, and is called a-priori variance factor.
- For this example, assume: $\sigma_o^2 = 1$

$$\therefore P = \sigma_o^2 C_l^{-1} = (1).C_l^{-1}$$

$$\because C_{l_{n\times n}} = I_{21\times 21} \rightarrow \therefore C_l^{-1}{}_{n\times n} = I_{21\times 21} \rightarrow \therefore P_{n\times n} = I_{21\times 21}$$

Step (7): Calculate the Misclosure Vector (w)

- The misclosure vector is defined as: $w = f(x_o) - l$

$f(x_o)$: is the model evaluated at initial values of unknowns

For this example, the initial values are: $a = 0 \quad b = 0 \quad c = 0$

Then, the model evaluated at initial values is as follows:

$$f(x_o) = A_{n\times u} x_{o_{u\times 1}} = A_{n\times u}[0]_{u\times 1} = [0]_{n\times 1}$$

$$w_{21\times 1} = -l_{21\times 1}$$

Part ONE: General Estimation - Curve Fitting
Problem ONE

Step (8): Solve for Correction to Initial Values of Unknowns (P)

$$\boxed{\delta_{u\times 1} = -N_{u\times u} . u_{u\times 1}}$$

$N_{u\times u} = N_{3\times 3}$: is the normal matrix. $u_{u\times u} = u_{3\times 3}$: is the normal vector.

Step (8.a): Calculate the Normal Matrix (N)

$$\therefore N_{u\times u} = N_{3\times 3} = A^T_{u\times n} . P_{n\times n} . A_{n\times u} = A^T_{3\times 21} . P_{21\times 21} . A_{21\times 3}$$

Step (8.b): Calculate the Normal Vector (u)

$$\therefore u_{u\times 1} = u_{3\times 1} = A^T_{u\times n} . P_{n\times n} . w_{n\times 1} = A^T_{3\times 21} . P_{21\times 21} . w_{21\times 1}$$

Step (8.c): Calculate the Corrections Vector (δ)

$$\boxed{\therefore \delta_{u\times 1} = \delta_{3\times 1} = -N_{u\times u} . u_{u\times 1} = -N_{3\times 3} . u_{3\times 1}}$$

Matlab Syntax:

```
A =[......]; % Fill in the design matrix.
P = eye(n , n); % n = 21
w = -y; % y is the observations vector
N = A'*P*A; % N is the normal matrix
u = A'*P*w; % u is the normal vector
Delta = -inv(N)*u; % Delta is the corrections vector
```

Part ONE: General Estimation - Curve Fitting Problem ONE

Step (9): Calculate the vector of estimated parameters ($\hat{x}$)

$$\hat{x}_{u\times1} = \hat{x}_{3\times1} = x_{o_{u\times1}} + \delta_{u\times1} = [0]_{3\times1} + \delta_{3\times1}$$

Since the model is linear, then there are no more iterations required.

Step (10): Calculate the vector of adjusted residuals ($\hat{v}$)

$$\hat{v}_{n\times1} = \hat{v}_{21\times1} = A_{n\times u}\hat{x}_{u\times1} - l_{n\times1} = A_{21\times3}.\hat{x}_{3\times1} - l_{21\times1}$$

As given

From Step (4)

From Step(9)

Step (11): Calculate the Optimality Criteria($\hat{v}^T P \hat{v}$) and a-posteriori variance factor ($\hat{\sigma}_o^2$)

$\therefore \hat{v}^T P \hat{v}$

Can be calculated using Matlab

$$\therefore \hat{\sigma}_o^2 = \frac{\hat{v}^T P \hat{v}}{D.o.F}$$

$D.o.F$: is the degrees of freedom

$D.o.F = n - u = 21 - 3 = 18$

Step (12): Calculate Parameters Variance Covariance Matrix ($C_{\hat{x}}$)

From Lecture Notes: $\therefore C_{\hat{x}_{3\times3}} = \hat{\sigma}_o^2 N_{u\times u}^{-1} = \hat{\sigma}_o^2 N_{3\times3}^{-1}$

- **<u>Solution:</u> For Requirement (I) - curve (a)**

b. $y = a\cos(2k) + b$ Sinusoidal Equation

DO IT BY YOURSELF

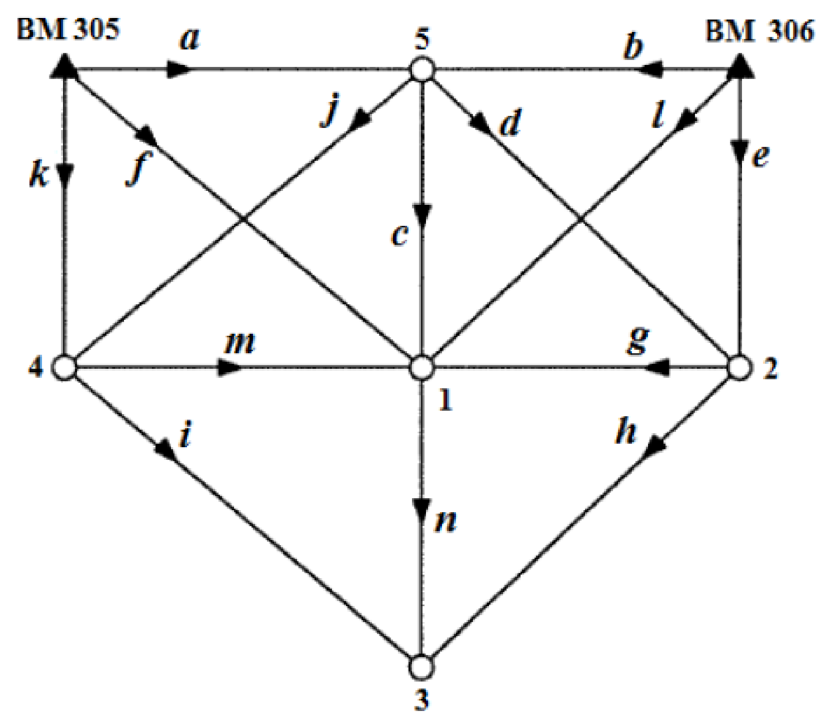
Repeat the same steps for curve (a)

Part TWO: Levelling Network Estimation Case Problem TWO

➢ Given:

- Leveling Network, as shown in figure, and measurements table.
- All observations are uncorrelated.
- Levels of BM_A, and BM_B are known constants. $H_A = 263.453\ m$, and $H_B = 263.453\ m$

Obs.	From station	To station	Observed height differences (m)	σ(m)
a	305	5	25.102	± 0.018
b	306	5	-6.287	± 0.019
c	5	1	10.987	± 0.016
d	5	2	24.660	± 0.021
e	306	2	17.993	± 0.017
f	305	1	36.075	± 0.021
g	2	1	-13.295	± 0.018
h	2	3	-20.732	± 0.022
i	4	3	18.445	± 0.022
j	5	4	-14.906	± 0.021
k	305	4	10.218	± 0.017
l	306	1	4.693	± 0.020
m	4	1	25.893	± 0.018
n	1	3	-7.456	± 0.020



Part TWO: Levelling Network Estimation Case
Problem TWO

- **Requirements:**

I. Write a computer program to perform parametric least squares adjustment that computes:

   i. Estimates of Adjusted Residuals ($\hat{v}$).
   ii. Estimates of Unknown Parameters ($\hat{x}$).
   iii. Estimates of Adjusted Observations ($\hat{l}$).
   iv. A-posteriori Variance Factor ($\hat{\sigma}_o^2$).
   v. Variance Covariance Matrix and Correlation Matrix for estimated parameters ($C_{\hat{x}}, \rho_{\hat{x}}$).
   vi. Variance Covariance Matrix and Correlation Matrix for adjusted observations ($C_{\hat{l}}, \rho_{\hat{l}}$).

II. Discussions:

   i. Interpret the meaning of diagonal and off-diagonal elements in each of the two covariance matrices. Interpret any significant off-diagonal elements in the correlation matrices.
   ii. Assume (σ_o^2) and repeat the adjustment. What do you notice about the results? Explain the results.

Part TWO: Levelling Network Estimation Case
Problem TWO

➢ **Solution:**

Phase One: Solution Scheme

Here we are going to follow the same steps as explained in problem **ONE**.

1. Identify unknown parameters ($\hat{x}$), observations ($\hat{l}$), and constants

Parameters vector ($\hat{x}$): $x_{5\times1} = [H_1 \quad H_2 \quad H_3 \quad H_4 \quad H_5]^T$ $\boxed{u = 5}$

Observations vector ($\hat{l}$): $l_{14\times1} = [a \quad b \quad \dots \quad m \quad n]^T$ $\boxed{n = 14}$

Constants vector (c): $c_{2\times1} = [H_A \quad H_B]^T$ $\boxed{c = 2}$

Let (BM_{305}) be (BM_A), and (BM_{306}) be (BM_B)

2. Form Observation Equations

Let No. of Equations be (m), and since we use indirect model, then: $m = n = 14$

$$
\begin{array}{lllll}
a = H_5 - H_A & d = H_2 - H_5 & g = H_1 - H_2 & j = H_4 - H_5 & m = H_1 - H_4 \\
b = H_5 - H_B & e = H_2 - H_B & h = H_3 - H_2 & k = H_4 - H_A & \\
c = H_1 - H_5 & f = H_1 - H_A & i = H_3 - H_4 & l = H_1 - H_B & n = H_3 - H_1
\end{array}
$$

<u>Phase One:</u> Solution Scheme - CONTINUED

2. Form Observation Equations - CONTINUED

$$l_{n\times1} = A_{n\times u} * \hat{x}_{u\times1} - \hat{v}_{n\times1} + Constants_{n\times1}$$

$$\begin{bmatrix} a \\ b \\ c \\ \vdots \\ m \\ n \end{bmatrix}_{14\times1} = \begin{bmatrix} 25.102 \\ -6.287 \\ 10.987 \\ \vdots \\ 25.893 \\ -7.456 \end{bmatrix}_{14\times1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & -1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 & 0 \end{bmatrix}_{14\times5} \begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \\ H_5 \end{bmatrix}_{5\times1} - \begin{bmatrix} v_a \\ v_b \\ v_c \\ \vdots \\ v_m \\ v_n \end{bmatrix}_{14\times1} + \begin{bmatrix} -H_A \\ -H_B \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{14\times1}$$

From table (2)

Remember: $A = \left.\frac{\partial f}{\partial x}\right|_{x=x_o}$; is matrix for Partial derivative of equations w.r.t. unknowns

3. Determine the Point of Expansion (POE)

∵ The model is linear. ➡ We can assume the initial values of the unknowns to be [**ZEROs**].

$$\therefore x_0 = [0]_{5\times1}$$

Part TWO: Levelling Network Estimation Case Problem TWO

Phase One: Solution Scheme - CONTINUED

4. Build Design Matrix (A)

From step **2** , because model is linear; however, if the model is non-linear, then you should evaluate the A-matrix using the POE.

Remember: $A = \frac{\partial f}{\partial x}\Big|_{x=x_o}$ is matrix for Partial derivative of equations w.r.t. unknowns evaluated at the initial values.

5. Establish Variance Covariance for Observations

$\because$ All measurements are uncorrelated ⟶ Covariances (σ_{ij}) = 0

$$\therefore C_{l_{n\times n}} = \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \cdots & \sigma_{an} \\ \sigma_{ba} & \sigma_b^2 & \cdots & \sigma_{bn} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{na} & \sigma_{nb} & \cdots & \sigma_n^2 \end{bmatrix}_{14\times 14} = \begin{bmatrix} (0.018)^2 & 0 & \cdots & 0 \\ 0 & (0.019)^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (0.020)^2 \end{bmatrix}_{14\times 14}$$

Alternatively: $\therefore C_{l_{n\times n}} = diag([(0.018)^2 \quad (0.019)^2 \quad \ldots \quad \ldots \quad (0.020)^2]_{1\times 14})$

Standard Deviations as given

Phase One: Solution Scheme - CONTINUED

6. Establish Weight Matrix (P)

Choose a priori variance factor (σ_o^2) → $\sigma_o^2 = 1$, for simplicity

Compute Weight Matrix (P) → $\boxed{\therefore P = \sigma_o^2 C_l^{-1} = (1).C_l^{-1}}$

7. Calculate Misclosure Vector ($w_{n\times1}$)

$$w_{n\times1} = f(x_o) - l \quad \longrightarrow \quad w_{14\times1} = \begin{bmatrix} H_5 - H_A - a \\ H_5 - H_B - b \\ H_1 - H_5 - c \\ \vdots \\ H_3 - H_1 - n \end{bmatrix}_{14\times1}$$

8. Calculate Normal Matrix (N) and Normal Vector (u)

$$\therefore N_{u\times u} = N_{5\times5} = A^T_{u\times n}.P_{n\times n}.A_{n\times u} = A^T_{5\times14}.P_{14\times14}.A_{14\times5}$$

Step 4 Step 6

$$\therefore u_{u\times1} = u_{5\times1} = A^T_{u\times n}.P_{n\times n}.w_{n\times1} = A^T_{5\times14}.P_{14\times14}.w_{14\times1}$$

Step 4 Step 6 Step 7

Phase One: Solution Scheme - CONTINUED

9. Calculate Corrections Vector (δ)

$$\therefore \delta_{u\times1} = \delta_{5\times1} = -N_{u\times u}^{-1} . u_{u\times1} = -N_{5\times5}^{-1} . u_{5\times1}$$

10. Calculate Vector of Estimated Parameters ($\hat{x}_{5\times1}$)

$$\hat{x}_{u\times1} = \hat{x}_{5\times1} = x_{o_{u\times1}} + \delta_{u\times1} = [0]_{5\times1} + \delta_{5\times1}$$

11. Calculate Vector of Adjusted Residuals ($\hat{v}$)

$$\hat{v}_{n\times1} = \hat{v}_{14\times1} = A_{n\times u}\hat{x}_{u\times1} - l_{n\times1} = A_{14\times5} . \hat{x}_{5\times1} - l_{14\times1}$$

Given

Step 4

Step 10

12. Calculate Vector of Adjusted Observations ($\hat{l}$)

$$\hat{l}_{n\times1} = \hat{l}_{14\times1} = l_{n\times1} + \hat{v}_{n\times1} = l_{14\times1} + \hat{v}_{14\times1}$$

Given

Step 11

Part TWO: Levelling Network Estimation Case Problem TWO

Phase One: Solution Scheme - CONTINUED

13. Calculate Posteriori Variance Factor ($\hat{\sigma}_o^2$)

$$\therefore \hat{\sigma}_o^2 = \frac{\hat{v}^T P \hat{v}}{D.o.F}$$

$D.o.F$: is the degrees of freedom

$$D.o.F = n - u = 14 - 5 = 9$$

14. Calculate Parameters Variance Covariance Matrix ($C_{\hat{x}}$)

$$C_{\hat{x}_{u\times u}} = C_{\hat{x}_{5\times 5}} = \hat{\sigma}_o^2 N_{u\times u}^{-1} = \hat{\sigma}_o^2 N_{5\times 5}^{-1}$$

15. Calculate Parameters Correlation Coefficient Matrix ($\rho_{\hat{x}}$)

$\rho_{\hat{x}_{u\times u}} = \rho_{\hat{x}_{5\times 5}}$ → Same as Lab ONE

16. Calculate Observations Variance Covariance Matrix ($C_{\hat{l}}$)

$C_{\hat{l}_{n\times n}} = C_{\hat{l}_{14\times 14}} = A_{n\times u} C_{\hat{x}_{u\times u}} A_{n\times u}^T = A_{14\times 5} C_{\hat{x}_{5\times 5}} A_{14\times 5}^T$ → From Variance Propagation Law

17. Calculate Observations Correlation Coefficient Matrix ($\rho_{\hat{l}}$)

$\rho_{\hat{l}_{n\times n}} = \rho_{\hat{l}_{14\times 14}}$ → Same as Lab ONE

Part TWO: Levelling Network Estimation Case Problem TWO

<u>Phase Two:</u> Programming Flowchart

Start

↓

Declare Input Variables
- Observations Vector (l)
- Parameters Vector (x)
- Constants Vector (c)
- A priori Standard Dev. (sigmaO)
- Standard Deviations (StdVs)

↓

Declare In-process Variables
- Design Matrix (A)
- V-C Observations (Cl)
- Weight Matrix (P)
- Normal Matrix (N)
- Normal Vector (u)
- Corrections Vector (delta)

↓

Declare Output Variables
- Estimated Parameters (xHat)
- Adjusted Residuals (vHat)
- Adjusted Observations (lHat)
- Posteriori Standard Dev. (sigmaOHat)
- V-C Parameters (Cx)
- V-C Adjusted Obs. (Cl)

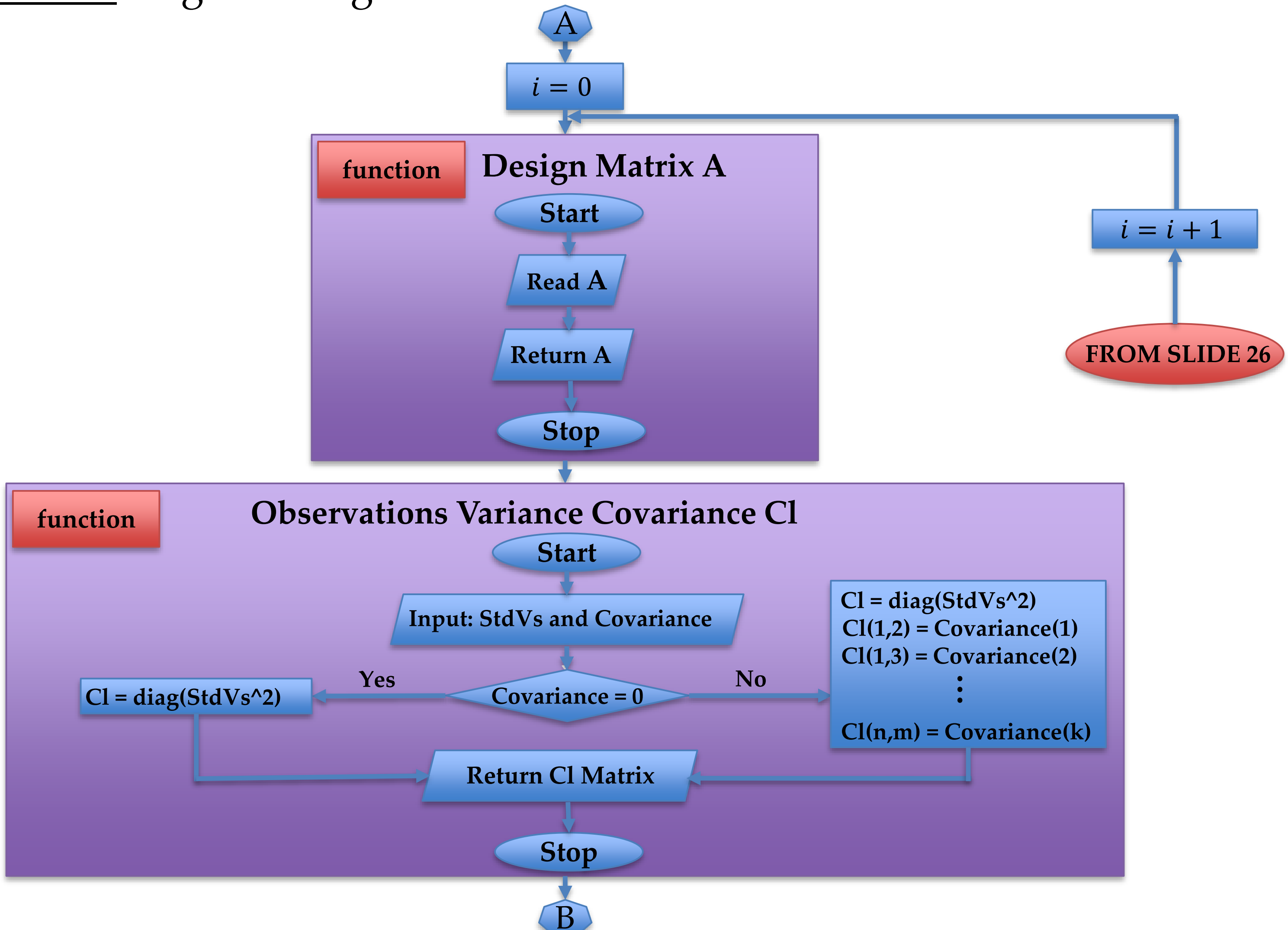
↓

Read Input Variables
- Observations
- Constants
- Parameters
- Standard Deviations

↓

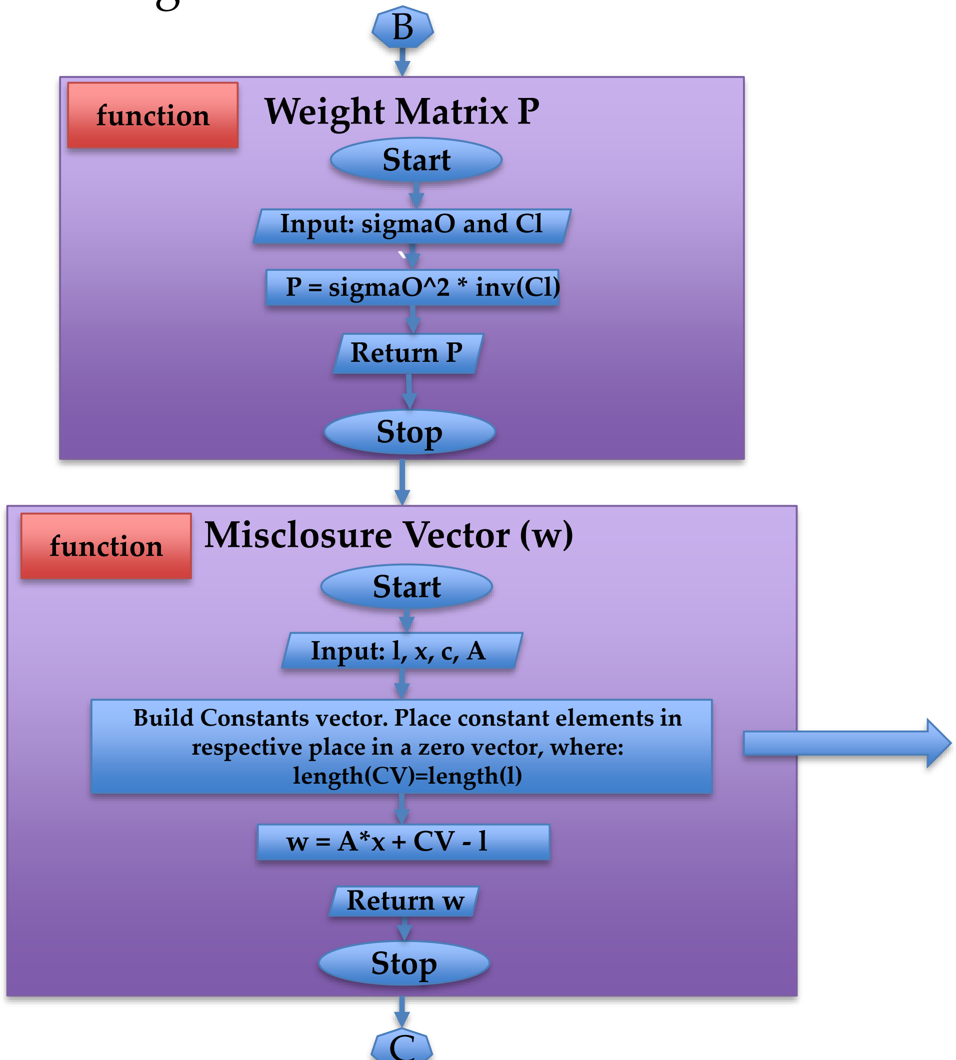
A

Phase Two: Programming Flowchart - Continued



Part TWO: Levelling Network Estimation Case Problem TWO

Phase Two: Programming Flowchart - Continued

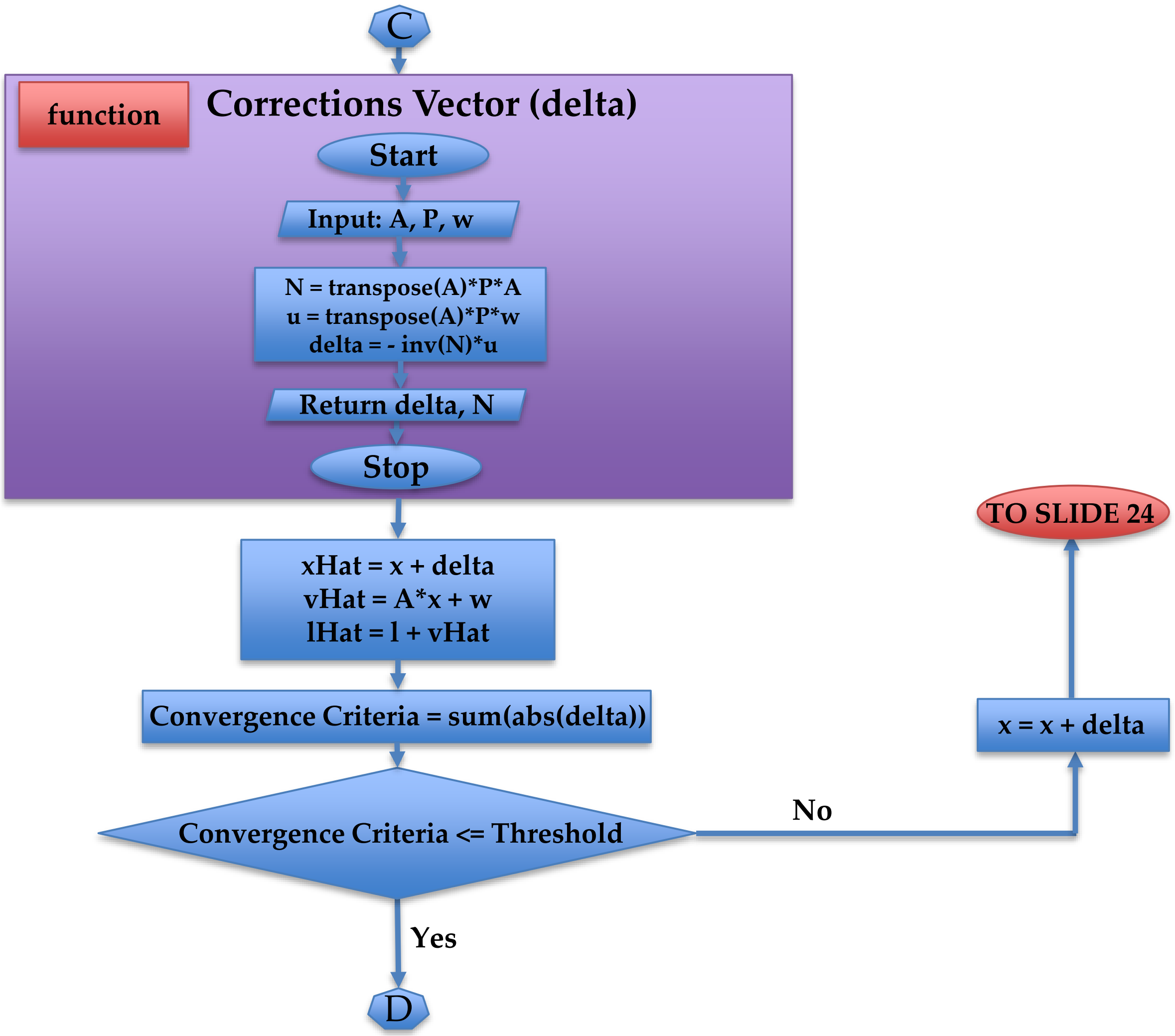


E.g.:

$$CV = \begin{bmatrix} -H_A \\ -H_B \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{14\times 1}$$

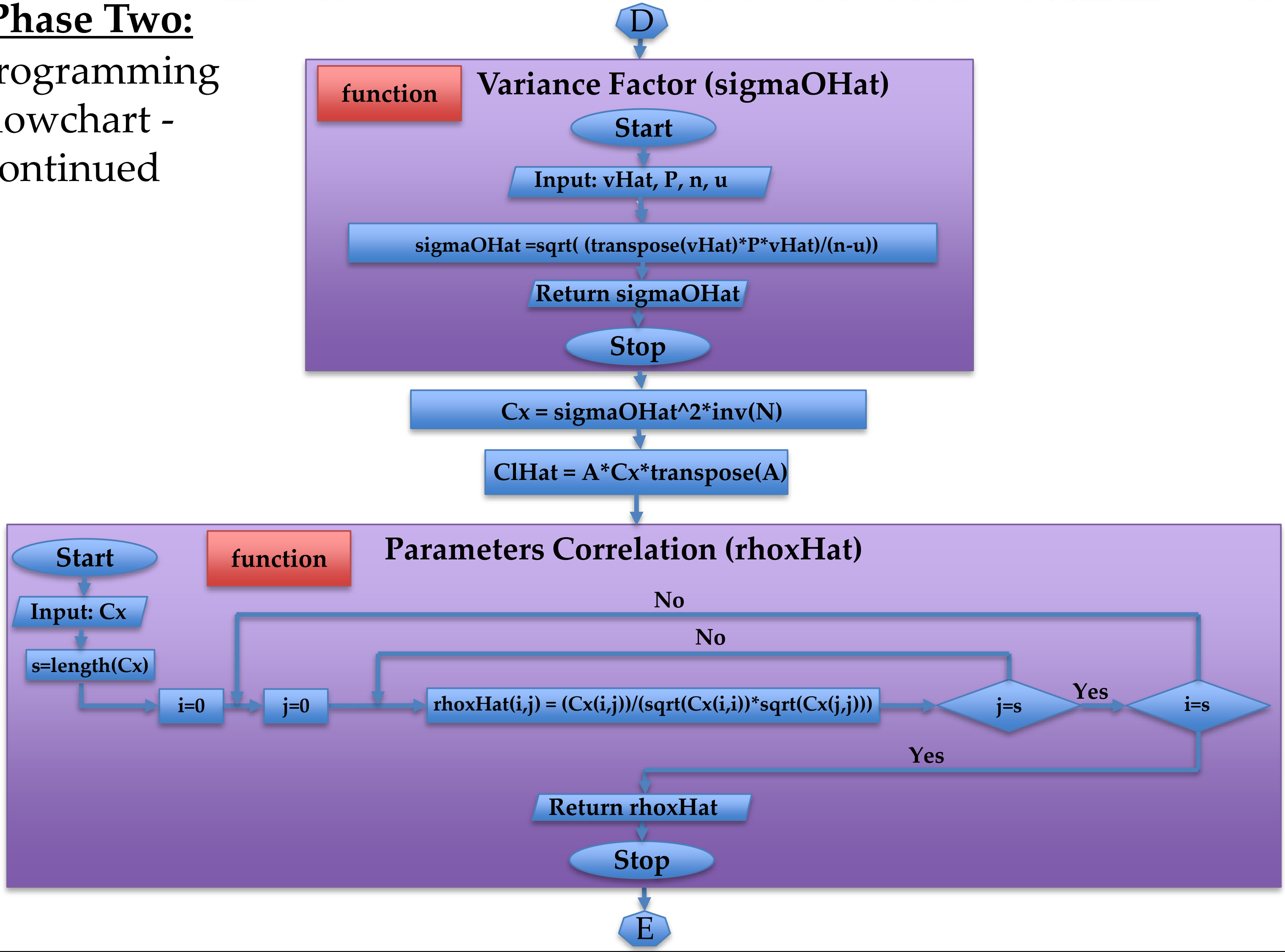
Part TWO: Levelling Network Estimation Case Problem TWO

Phase Two: Programming Flowchart - Continued



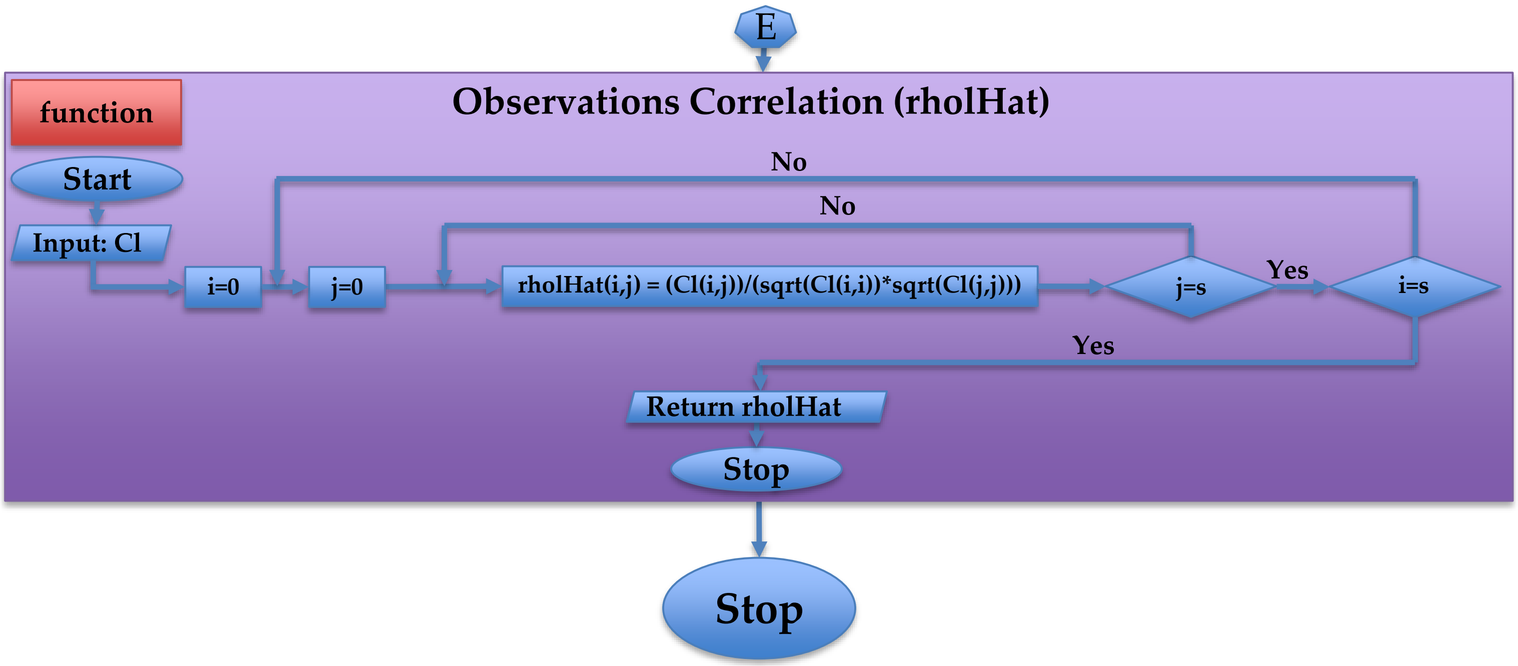
Phase Two:

Programming Flowchart - Continued



Part TWO: Levelling Network Estimation Case Problem TWO

Phase Two: Programming Flowchart - Continued



Part TWO: Levelling Network Estimation Case
Problem TWO

Phase Three: Program Requirements and Program Specifications

- **The program must be written in C or C++ language (preferably C++)**

1. Program must be written in a general form, such that you can re-dimension and redefine matrices and vectors.
2. Program should be modular, such that it consists of predefined functions to perform any repetitive tasks and enhance the readability of the code.
3. Documentation must be provided with the source code. The purpose of the program, functions, and equations must be described in detail within the code (in the form of comments).
4. All input data for the program must come from external files. No hard coding of data is permitted.
5. All output data are to be written to text files, and they must be submitted with the lab report.
6. Besides being working, the code will be evaluated on the use of comments, layout, readability, and style.
7. Program must be run for problem ONE, and problem TWO.

Part TWO: Levelling Network Estimation Case Problem TWO

Phase Four: Program Build Pre-requisites

1. Integrated Development Environment (IDE)
   - Use any IDE for C++
   - Examples provided hereafter are based on Code Blocks IDE.
     - Download the IDE from: https://sourceforge.net/projects/codeblocks/

2. Compilers [depend mostly on Eigen Library (will be discussed later) and its compatible compliers]
   - List of compatible compilers is given in: http://eigen.tuxfamily.org/index.php?title=Main_Page
   - Examples provided hereafter are based on GNC GCC Compiler

3. If you are a beginner in C++, you can make use of this video tutorials on You Tube. The URL is:
   - https://www.youtube.com/watch?v=e1YafrxYOWw&list=PLS1QulWo1RIYSyC6w2-rDssprPrEsgtVK

4. C++ libraries required for the program build:

   - `iostream`
   - `fstream`
   - `stdlib.h`
   - `math.h`
   - `Eigen`

Phase Four: Program Build Pre-requisites

5. Eigen Library has to be used for all vectors and matrices computations within your C++ code.
6. Eigen Library download and setup:
   - You can download the eigen library from: http://bitbucket.org/eigen/eigen/get/3.3.4.tar.bz2
   - Extract the downloaded file in a well-known directory on your PC.
   - To link the Eigen library to your C++ Code Blocks IDE, follow the instructions provided on this webpage: http://www.learncpp.com/cpp-tutorial/a3-using-libraries-with-codeblocks/
   - Then, to include the Eigen library to your code, type in:

```
#include<Eigen/Eigen>
```

   - You need to disable "assertions" to avoid, what is referred to as "assertion failure". To do so you need to type in, **BEFORE** you include the Eigen library:

```
#define EIGEN_NO_DEBUG
```

   - To use the `typedef` and the class templates in the Eigen library, you need to type in, **BEFORE** the main function:

```
using namespace Eigen;
```

   - To use dynamic double matrices and vectors, you need to type in, **BEFORE** the main function:

```
using Eigen::MatrixXd
using Eigen::VectorXd
```

Part TWO: Levelling Network Estimation Case Problem TWO

Phase Four: Program Build Pre-requisites

7. Useful functions and operations in Eigen Library:
   - To use dynamic double matrices and vectors, you need to type in, **BEFORE** the main function:

```
using Eigen::MatrixXd
using Eigen::VectorXd
```

   - To declare a matrix (A) of dynamic dimensions and double data type, you need to type in:

```
MatrixXd A(rows , columns);
```

   - To declare a vector (x) of dynamic dimensions and double data type: you need to type in:

```
VectorXd x(rows);
```

   - Basic Operations:

Operation	C++ Syntax
Summation and Subtraction	`Matrix3 = Matrix1 + Matrix2;` `// Matrix 1,2, and 3 are of same size`
Multiplication	`Matrix3 = Matrix2 * Matrix1;` `// No. of cols in 2 = No. of rows in 1`
Transpose Matrix/Vector	`Matrix2 = Matrix1.transpose();`
Inverse Matrix	`Matrix2 = Matrix1.inverse();`

Part TWO: Levelling Network Estimation Case Problem TWO

<u>Phase Four:</u> Program Build Pre-requisites

7. Useful functions and operations in Eigen Library - CONTINUED:
   - Basic Operations - CONTINUED:

Operation	C++ Syntax
Represent vector as diagonal matrix	`Matrix1 = Vector1.asDiagonal();`
Matrix Indexing	`//To get element in row (i), column (j), use:` `Element = Matrix1(i,j);`
Matrix Size	`Matrix.size(); OR Matrix.rows(); OR Matrix.cols();`

For those who are interested in learning the full functionality of the Eigen library, you can refer to, this documentation webpage:
http://eigen.tuxfamily.org/dox/

8. Use of functions in C++

```
Datatype FunctionName(datatype1 arguement1, datatype2 arguement2,…)
{
                        FUNCTION BODY
return variable;}
```

Phase Four: Program Build Pre-requisites

8. Example of functions in C++: Function to build Weight Matrix

```
// FUNCTION SIX: Build Weight Matrix
    MatrixXd WeightMatrix(double sigmaO, MatrixXd Cl){
    MatrixXd P(Cl.size(),Cl.size());
    P = sigmaO * Cl.inverse();
    return P;}
```

9. Read/Write with Text Files in C++
   - Use the following links:
     - Write to a *.txt file in C++: https://www.youtube.com/watch?v=13TrhiKLZg8&list=PLS1QulWo1RIYSyC6w2-rDssprPrEsgtVK&index=45
     - Read from a *.txt file in C++: https://www.youtube.com/watch?v=lzxWNtjii8U&index=46&list=PLS1QulWo1RIYSyC6w2-rDssprPrEsgtVK

Part TWO: Levelling Network Estimation Case Problem TWO

Phase Five: Guidelines for Program Build

1. Prepare a separate text file for:
   - Observations
   - Point of Expansions of Parameters
   - Constants
   - Standard Deviations of Observations
   - Design Matrix.
2. Open a new C++ project in your IDE of choice.
3. In your main source file:
   3.1. Include all required libraries.
   3.2. Define used namespaces.
   3.3. Build the required functions, to:
   - Read data from text to vector
   - Read data from text to matrix
   - Build/Read Design Matrix
   - Build Observations Variance Covariance Matrix
   - Build weight Matrix
   - Build Misclosure Vector
   - Compute Corrections Vector

Part TWO: Levelling Network Estimation Case Problem TWO

Phase Five: Guidelines for Program Build

3.3. Build the required functions - CONTINUED:

- Compute Variance Factor
- Compute Correlation Matrix

3.4. In your main function:

- Declare all inputs and outputs
- Parametric LS solution loop [as in flowchart]
- Computed the required outputs
- Type the outputs in a text file.

➢ **INCLUDE YOUR SOURCE CODE**, with all of its descriptions and comments, with your final report

➢ **INCLUDE FOUR OUTPUT TEXT FILES**, from your program, with your final report, such that you have one output files for each of the following:

- Problem One – Part A: Parabola Curve Fitting
- Problem One – Part B: Sinusoidal Curve Fitting
- Problem Two – Main Requirement
- Problem Two – Part II, No. (b) using $\sigma_o^2 = 2$

Part TWO: Levelling Network Estimation Case Problem TWO

➢ **Requirements:**

II. Discussions:

i. Interpret the meaning of diagonal and off-diagonal elements in each of the two covariance matrices. Interpret any significant off-diagonal elements in the correlation matrices.

Same as you did in Lab ONE, problem TWO, for interpreting the correlation coefficient matrix

ii. Assume $(\sigma_o^2 = 2)$ and repeat the adjustment. What do you notice about the results? Explain the results.

- Your discussion should include:

a. Describe the differences between the two output files?
b. Which solution is better in terms of quality?
c. Why did you conclude that?