CE697 Adv. Data Adj.: Homework 1

assigned Tue. 24 May, due Thur. 2 June

1. derive symbolic expression for $Q_{\Delta\Delta}$ for

   (a) direct treatment of constraints with Full rank N

   (9.15a), p. 214 : $Q_{\Delta\Delta} = N^{-1}(I - C^T M^{-1} C N^{-1})$

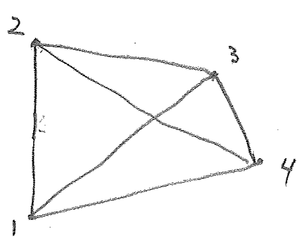
   $$M = C N^{-1} C^T$$

   (b) solution by elimination of constraints

   (9.29), p. 219 : $Q_{\Delta\Delta} = \begin{bmatrix} C_1^{-1} C_2 \bar{N}^{-1} C_2^T (C_1^{-1})^T & -C_1^{-1} C_2 \bar{N}^{-1} \\ -\bar{N}^{-1} C_2^T (C_1^{-1})^T & \bar{N}^{-1} \end{bmatrix}$

   hint: use cross cofactor technique for off-diagonal blocks

2. Solve the 2D range network using minimal constraints (direct treatment, N not full rank)



(obs.)

$d_{12} = 364.75$

$d_{23} = 327.16$

$d_{34} = 225.91$

$d_{41} = 522.26$

$d_{13} = 512.29$

$d_{24} = 492.70$

$\sigma_d = 0.5$

constraints: $X_1 = 1140.00$

$Y_1 = 310.00$

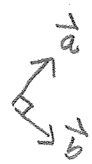
$Y_2 - Y_3 = 40.00$

extra credit: solve also by elimination.

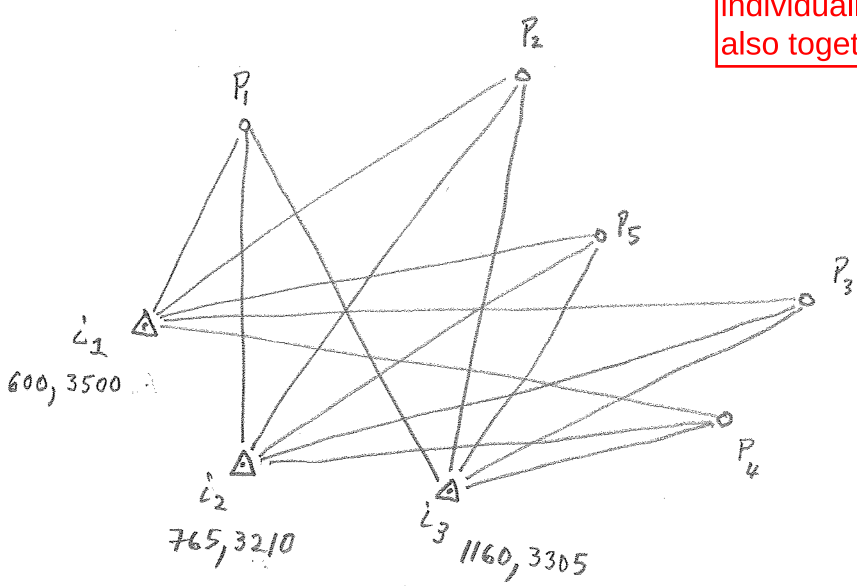
3. Solve the 2D range network with 3 fixed instrument stations and 5 unknown target points. Constrain angle P_2-P_3-P_4 to be exactly $90°$, and constrain distance P_1-P_4 to be exactly 865.97

First solve the unconstrained problem and report the p-value of the F statistic corresponding to each of the 2 constraints. When solving the ~~un~~constrained problem, do with both the direct method (N full rank), and by elimination (i.e. parameter reduction)

with each of the constraints individually and also together

Recommend 

$\vec{a} \cdot \vec{b} = 0$ to force vectors to be $90°$.

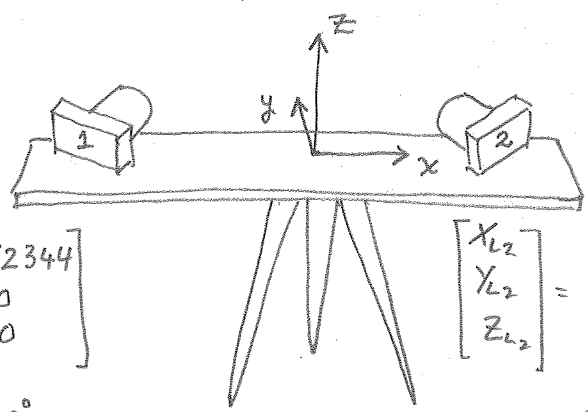


dist. observations

	P_1	P_2	P_3	P_4	P_5
i_1	364.21	846.02	1158.07	1000.62	837.12
i_2	595.55	948.59	1086.99	875.39	820.84
i_3	615.71	714.64	695.06	471.74	483.19

$\sigma_d = 0.5$

4. A 2 camera "cluster" is configured and calibrated with respect to a local reference system:



$$\begin{bmatrix} X_{L_1} \\ Y_{L_1} \\ Z_{L_1} \end{bmatrix} = \begin{bmatrix} -.982344 \\ 0 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} X_{L_2} \\ Y_{L_2} \\ Z_{L_2} \end{bmatrix} = \begin{bmatrix} +.982344 \\ 0 \\ 0 \end{bmatrix}$$

$\omega_1 = 90^\circ$ $\qquad$ $\omega_2 = 90^\circ$

$\phi_1 = -17.59^\circ$ $\qquad$ $\phi_2 = +17.59^\circ$

$K_1 = 0$ $\qquad$ $K_2 = 0$

The approximate transformation between a "world" coordinate system and the local system is:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{local} \approx M_{w2l} \left(\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{world} + \begin{bmatrix} -101.25 \\ -500.95 \\ -1.00 \end{bmatrix} \right)$$

$$M_{w2l} \approx M_Z(14.74^\circ)$$

World coordinates for 3 control points and 4 tie points (approximations) are:

△1	99.30	503.70	1.75
△2	101.20	504.20	1.75
△3	100.25	503.92	0.20
4	99.30	503.70	0.20
5	101.20	504.20	0.20
6	100.20	503.90	1.00
7	100.20	503.90	1.80

$f = 20.00$ mm

image observations:

	x (image 1)	y (image 1)	x (image 2)	y (image 2)
1	−7.793	5.107	−6.061	4.094
2	4.118	4.227	4.955	4.887
3	−1.304	−5.001	−1.168	−4.781
4	−7.789	−5.111	−6.081	−4.092
5	4.112	−4.236	4.966	−4.883
6	−1.302	0.002	−1.148	−0.006
7	−1.302	4.991	−1.167	4.788

image 1 $\qquad$ image 2

$\sigma_x = \sigma_y = 0.010$ mm

Enforce the constraints (camera cluster geometry) implicitly by fixing the positions and orientations of both cameras in local system, and transform world points into the local system on the fly, as you estimate the 6 transformation parameters.

$$\begin{array}{l} \text{observed} \\ \text{fixed} \end{array} \left\{ \begin{bmatrix} x \\ y \\ -f \end{bmatrix} \right. = \lambda \underset{\substack{\uparrow \\ \text{fixed} \\ \text{local}}}{M} \left[\underset{\substack{\uparrow \\ \text{unknown} \\ \text{angles}}}{M_{w2l}} \left[\underset{\substack{\uparrow \\ \text{fixed if CP} \\ \text{unknown if TP}}}{\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{world}} + \underset{\substack{\uparrow \\ \text{unknown} \\ \text{shifts}}}{\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}} \right] - \underset{\substack{\uparrow \\ \text{fixed local}}}{\begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix}} \right]$$

recommend constructing 3 matlab functions, different from those suggested in lecture #2 :

$$\begin{pmatrix} x \\ y \end{pmatrix} = fcoll2(f, \underbrace{\omega, \varphi, \kappa, X_L, Y_L, Z_L}_{\text{fixed local}}, \underbrace{X, Y, Z}_{\text{world}}, \underbrace{\omega_t, \varphi_t, \kappa_t, t_x, t_y, t_z}_{\substack{\text{w2l transformation} \\ \text{(unknowns)}}})$$

$$\begin{bmatrix} d_x \\ d_y \end{bmatrix} = \begin{bmatrix} x_0 - x_c \\ y_0 - y_c \end{bmatrix} = fcoll2\_0(\underbrace{x_0, y_0}_{\text{observed}}, f, \omega, \varphi, \kappa, X_L, Y_L, Z_L, X, Y, Z, \omega_t, \varphi_t, \kappa_t, t_x, t_y, t_z)$$

calls prior function

$$\begin{bmatrix} \cdots\cdots\cdots \\ \cdots\cdots\cdots \end{bmatrix} = fcoll2\_0\_part(x_0, y_0, f, \omega, \varphi, \kappa, X_L, Y_L, Z_L, X, Y, Z, \omega_t, \varphi_t, \kappa_t, t_x, t_y, t_z)$$

calls prior function

$$\begin{bmatrix} \frac{\partial F_x}{\partial X} & \frac{\partial F_x}{\partial Y} & \frac{\partial F_x}{\partial Z} & \frac{\partial F_x}{\partial \omega_t} & \frac{\partial F_x}{\partial \varphi_t} & \frac{\partial F_x}{\partial \kappa_t} & \frac{\partial F_x}{\partial t_x} & \frac{\partial F_x}{\partial t_y} & \frac{\partial F_x}{\partial t_z} \\ \frac{\partial F_y}{\partial X} & \frac{\partial F_y}{\partial Y} & \frac{\partial F_y}{\partial Z} & \frac{\partial F_y}{\partial \omega_t} & \frac{\partial F_y}{\partial \varphi_t} & \frac{\partial F_y}{\partial \kappa_t} & \frac{\partial F_y}{\partial t_x} & \frac{\partial F_y}{\partial t_y} & \frac{\partial F_y}{\partial t_z} \end{bmatrix}$$

suggest numerical approx. for partials :

$$\frac{\partial F}{\partial X} \approx \frac{F(x+dx, y, z) - F(x, y, z)}{dx}$$

etc.