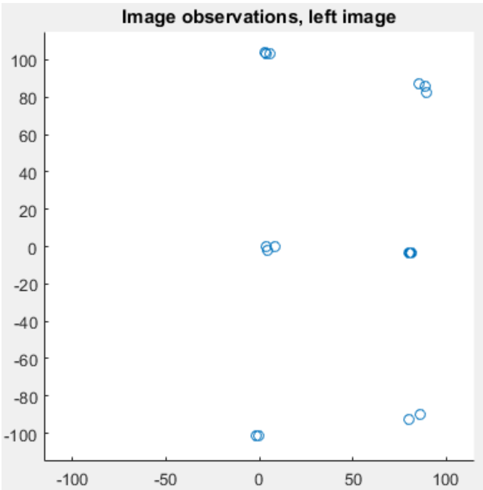
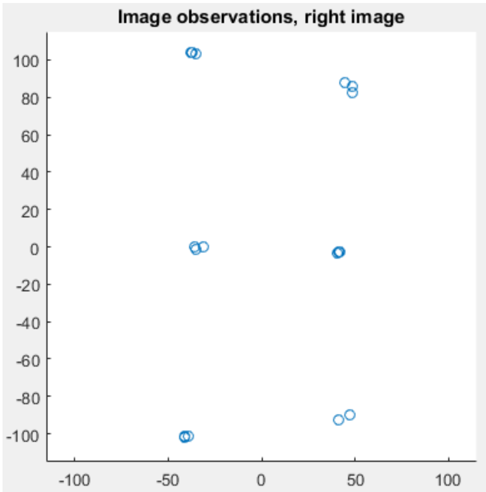
GIS-E3010 Least-Squares Methods in Geosciences

Fotogrammetrian harjoitus

Harjoituksessa harjoitellaan käytännössä pienimmän neliösumman menetelmän käyttöä useissa fotogrammetrisissa tehtävissä. Käytössä on kaksi simuloitua ilmakuvaa, niiden kuvahavainnot ja maaston tukipisteitä. Näiden aineistojen avulla tehtävänäsi on laskea (ohjelmoida Matlabissa) yhden kuvan ulkoinen orientointi, kahden kuvan keskinäinen orientointi ja laskea eteenpäinleikkauksella 3D pisteitä. Simuloituilla ilmakuvilla (koko 23 x 23 cm) on noin 80 % pituuspeitto ja jokaiselta kuvalta on mitattu 18 pistettä (jakauma vastaa lähes Gruberin pisteitä, kustakin paikasta on mitattu 3 pistettä) Kaikki mittaukset ovat siinä mielessä ideaalisia, että kuvakorjauksia ei tarvitse tehdä. Kaikki tarvittavat lähtölikiarvot on annettu. Todellisuudessa nämä likiarvoiset tunemattomien parametrien arvot pitäisi laskea jonkin suoran ratkaisumenetelmän avulla.





Lisäksi sädekimmputasoituksen periaatetta harjoitellaan kynällä ja paperilla. Tämä osuus ei sisällä ohjelmointia (välttämättä).

Arviointi

Harjoitus arvioidaan. Fotogrammetrian harjoituksen paino kokonaisarvosanaan on 1/3. Maksimipisteet tästä tehtävästä on 90 pistettä, joka skaalataan vastaamaan muita kurssin harjoituksia. Liitä raporttiin myös palautetta ja arvio, paljonko tarvitsit aikaa harjoituksen tekemiseen.

Deadline on 15.2.2019.

Laitteistovaatimukset

Harjoituksen ohjelmointiosuuksiin tarvitaan Matlabia ja osaan kohtia sen Symbolic library –kirjastoa. Nämä löytyvät Aalto-koneilta.

Help

Kurssilla on kaksi harjoituskertaa, jossa ohjelmointiosuuksiin on mahdollista saada apua (Ke 30.1. 14-16 ja Ke 6.2. 14-16).

Kysymyksiä voi myös lähettää sähköpostilla: petri.ronnholm@aalto.fi

Ulkoisen orientoinnin ratkaisu (27 p)

Tässä harjoituksen osassa oletetaan, että kohdepisteet tunnetaan äärettömän tarkasti eli ne ovat vakioita.

Tavoitteena on ratkaista kollineaarisuusyhtälöistä tuntemattomat parametrit ($\omega, \varphi, \kappa, X_0, Y_0, Z_0$). Kollineaarisuusyhtälön ylempää yhtälöä merkitään f_x ja alempaa f_y.

$$\begin{cases} x = -c\dfrac{r_{11}(X - X_0) + r_{12}(Y - Y_0) + r_{13}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)} = \dfrac{U}{W} = f_x \\ y = -c\dfrac{r_{21}(X - X_0) + r_{22}(Y - Y_0) + r_{23}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)} = \dfrac{V}{W} = f_y \end{cases}$$

Yhtälöissä 3D matriisi on

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = R_\kappa R_\varphi R_\omega = \begin{bmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\varphi & 0 & -\sin\varphi \\ 0 & 1 & 0 \\ \sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{bmatrix}$$

$$= \begin{bmatrix} \cos\varphi\cos\kappa & \cos\omega\sin\kappa + \sin\omega\sin\varphi\cos\kappa & \sin\omega\sin\kappa - \cos\omega\sin\varphi\cos\kappa \\ -\cos\varphi\sin\kappa & \cos\omega\cos\kappa - \sin\omega\sin\varphi\sin\kappa & \sin\omega\cos\kappa + \cos\omega\sin\varphi\sin\kappa \\ \sin\varphi & -\sin\omega\cos\varphi & \cos\omega\cos\varphi \end{bmatrix}$$

Yhtälöiden rakenne ja trigonometriset yhtälöt tekevät siitä epälineaarisen. Siksi yhtälöt on linearisoitava. Tällöin tarkoituksena on ratkaista parannukset (=korjaukset) lähtölikiarvoille (tai parametrien edellisen iteraatiokierroksen arvoille) käyttäen pienimmän neliösumman menetelmää. Ratkaistut parannukset parametreille lisätään parametrien senhetkisiin arvoihin seuraavaa iteraatiokierrosta varten. Iteraatio voidaan lopettaa, kun kaikki korjaukset ovat riittävän pieniä (muutos ei ole enää merkittävä).

<u>Yksityiskohtaiset ohjeet</u>

Vastaa myös annettuihin kysymyksiin

K: Mitä havainnot ovat tässä tapauksessa vastinpistehavaintoja? (1 p)

K: Montako yhtälöä yksi vastinpistemittaus muodostaa? (1 p)

K: Kuinka monta yhtälöä syntyy, kun on mitattu 6 vastinpistehavaintoa? (1 p)

K: Montako tuntematonta parametria systeemissä on? (1 p)

K: Montako yhtälöä (vähintään) tarvitset, jotta saat ratkaistua tuntemattomat ja montako vastinpistehavaintoa tällöin tarvitset? (1 p)

Huomaa: on edullista tehdä enemmän vastinpistehavaintoja kuin minimi edellyttäisi.

K: Mikä on systeemin ylimääritys, jos käytössä on 4 vastinpistehavaintoa? (1 p)

Seuraava vaihe on luoda virheyhtälöt

$$v = Ax - y$$

joka näyttää yhdelle vastinpistehavainolle seuraavalta:

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f_x}{\partial \omega} & \frac{\partial f_x}{\partial \varphi} & \frac{\partial f_x}{\partial \kappa} & \frac{\partial f_x}{\partial X_0} & \frac{\partial f_x}{\partial Y_0} & \frac{\partial f_x}{\partial Z_0} \\ \frac{\partial f_y}{\partial \omega} & \frac{\partial f_y}{\partial \varphi} & \frac{\partial f_y}{\partial \kappa} & \frac{\partial f_y}{\partial X_0} & \frac{\partial f_y}{\partial Y_0} & \frac{\partial f_y}{\partial Z_0} \end{bmatrix} \begin{bmatrix} d\omega \\ d\varphi \\ d\kappa \\ dX_0 \\ dY_0 \\ dZ_0 \end{bmatrix} - \begin{bmatrix} x - f_x^0 \\ y - f_y^0 \end{bmatrix}$$

Jotta saisimme ratkaistua ratkaisuvektorin parametrien parannuksille (x), kaikki muut osat pitää muodostaa.

K: Kirjoita tämä yhtälö 4:lle vastinpistehavainnolle käyttäen yleisiä symboleita (lisää x:lle ja y:lle ali-indeksi viittaamaan monesko vasrtinpiste on kyseessä). Esimerkiksi ensimmäiselle vastinpisteelle: (1 p)

$$\begin{bmatrix} v_{x1} \\ v_{y1} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_{x1}}{\partial \omega} & \frac{\partial f_{x1}}{\partial \varphi} & \frac{\partial f_{x1}}{\partial \kappa} & \frac{\partial f_{x1}}{\partial X_0} & \frac{\partial f_{x1}}{\partial Y_0} & \frac{\partial f_{x1}}{\partial Z_0} \\ \frac{\partial f_{y1}}{\partial \omega} & \frac{\partial f_{y1}}{\partial \varphi} & \frac{\partial f_{y1}}{\partial \kappa} & \frac{\partial f_{y1}}{\partial X_0} & \frac{\partial f_{y1}}{\partial Y_0} & \frac{\partial f_{y1}}{\partial Z_0} \end{bmatrix} \begin{bmatrix} d\omega \\ d\varphi \\ d\kappa \\ dX_0 \\ dY_0 \\ dZ_0 \end{bmatrix} - \begin{bmatrix} x_1 - f_{x1}^0 \\ y_1 - f_{y1}^0 \end{bmatrix}$$

Käytännössä vain vektorit *v* ja *y* sekä matriisi *A* kasvavat (niihin tulee lisää rivejä).

Rakennematriisi *A* on Jakobin matriisi, jonka alkiot ovat matemaattisen mallin osittaisderivaattoja kunkin tuntemattoman parametrin suhteen. Kukin sarake on varattu yhdelle tuntemattomalle parametrille.

Matlab-tehtävä: Luo matriisiin *A* tarvittavat osittaisderivaatat. Lisää koodisi raporttiin. (10 p)

Valmiiksi johdetut osittaisderivaatat löytyvät kirjallisuudesta (ja ne voi tehdä myös itse kynällä ja paperilla), mutta tässä harjoituksessa suositaan Matlabin symbolista kirjastoa, jonka avulla luot omat versiosi osittaisderivaatoista. Ohjeita tähän löytyy Liitteestä 1.

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f_x}{\partial \omega} & \frac{\partial f_x}{\partial \varphi} & \frac{\partial f_x}{\partial \kappa} & \frac{\partial f_x}{\partial X_0} & \frac{\partial f_x}{\partial Y_0} & \frac{\partial f_x}{\partial Z_0} \\ \frac{\partial f_y}{\partial \omega} & \frac{\partial f_y}{\partial \varphi} & \frac{\partial f_y}{\partial \kappa} & \frac{\partial f_y}{\partial X_0} & \frac{\partial f_y}{\partial Y_0} & \frac{\partial f_y}{\partial Z_0} \end{bmatrix} \begin{bmatrix} d\omega \\ d\varphi \\ d\kappa \\ dX_0 \\ dY_0 \\ dZ_0 \end{bmatrix} - \begin{bmatrix} x - f_x^0 \\ y - f_y^0 \end{bmatrix} \qquad v = Ax - y$$

Seuraavaksi pitäisi ymmärtää vektoria *y*, joka sisältää havaittujen kuvakoordinaattien (*x*,*y*) ja laskennallisesti estimoitujen (vastaavien) kuvakoordinattien (f_x^0, f_y^0) erotuksen. Estimoidut kuvakoordinaatit lasketaan tässä käyttämällä likiarvoja kaikille tuntemattomille parametreille eli siis lasket annettujen maaston 3D pisteiden projektiot (sijainnit) kuvalle kollineaarisuusyhtälöillä käyttämällä likiarvoista orientointitietoa. Voit tarkistaa osittaisderivaattasi kopioimalla tulokset tiedostoon check_partial_derivatives.m ja ajamalla sen.

Matlab-tehtävä: Täydennä annettu Matlab-funktio rakentamalla A-matriisin ja y-vektorin sekä laskemalla ulkoisen orientoinnin parametrit. Lisää koodisi ja tulokset raporttiin. (10 p)

Keskinäisen orientoinnin ratkaisu (26 p)

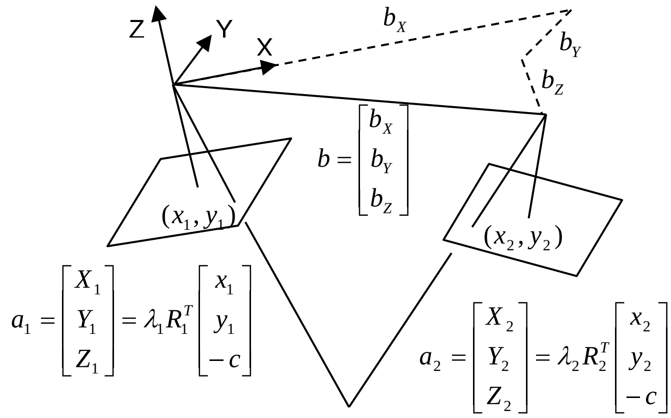
Tällä kertaa matemaattisena mallina on koplanariteettiyhtälö. Tavoitteena on ratkaista keskinäinen orientointi kuvaliitosmenetelmällä (eli yksi kuva pysyy paikallaan ja toista siirretään ja kierretään).

$$G = \begin{vmatrix} b_X & b_Y & b_Z \\ X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \end{vmatrix} = b_X(Y_1Z_2 - Y_2Z_1) - b_Y(X_1Z_2 - X_2Z_1) + b_Z(X_1Y_2 - X_2Y_1) = 0$$

jossa

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = R_1^T \begin{bmatrix} x_1 \\ y_1 \\ -c \end{bmatrix}$$

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = R_2^T \begin{bmatrix} x_2 \\ y_2 \\ -c \end{bmatrix}$$



Kiertomatriisi on identtinen kuin edellä (ulkoisen orientoinnin laskennassa). Trigonometriset funktiot yhtälössä tekevät siitä epälineaarisen, joten se on linearisoitava. Kuten aina epälineaarisessa tapauksessa, kaikille tuntemattomille parametreille on oltava kohtuullisen hyvät lähtölikiarvot (tässä harjoituksessa likiarvot on annettu). Tällä kertaa käytetään yleistä (sekamalli) pienimmän neliösumman tasoitusta. Rakenne on tällöin

$$\begin{bmatrix} \frac{\partial G^0}{\partial x_1} & \frac{\partial G^0}{\partial y_1} & \frac{\partial G^0}{\partial x_2} & \frac{\partial G^0}{\partial y_2} \end{bmatrix} \begin{bmatrix} dx_1 \\ dy_1 \\ dx_2 \\ dy_2 \end{bmatrix} + \begin{bmatrix} \frac{\partial G^0}{\partial b_Y} & \frac{\partial G^0}{\partial b_Z} & \frac{\partial G^0}{\partial \omega_2} & \frac{\partial G^0}{\partial \varphi_2} & \frac{\partial G^0}{\partial \kappa_2} \end{bmatrix} \begin{bmatrix} db_Y \\ db_Z \\ d\omega_2 \\ d\varphi_2 \\ d\kappa_2 \end{bmatrix} + G^0 = 0$$

eli $C^T dl + Ddp + G^0 = 0$

K: Mitä ovat tässä tapauksessa vastinpistehavainnot? (1 p)

K: Kuinka monta yhtälöä yksi vastinpistehavainto tuottaa? (1 p)

K: Kuinka monelle parametrille ratkaisemme parannuksen? (2 p)

K: Mikä on systeemin ylimääritys, jos käytössä on 9 vastinpistehavaintoa? (1 p)

K: Kuinka voit valita tulevan 3D mallin mittakaavan? (1 p)

Matlab-tehtävä: Luo osittaisderivaatat matriisien *C* ja *D* alkioille (käytä annettua Matlab-pohjaa make_all_partial_derivates_C_and_D_matrices_RO.m). (Huomaa, että tämän ratkaisu on annettu valmiina eli saat ilmaiset 10 pistettä. Näin saadaan kevennettyä harjoituksen työkuormaa. Käy kuintekin koodi läpi ja koeta ymmärtää, mitä siinä tehtiin. Jos on ylimääräistä aikaa, vaoit toki koettaa tehdä tämän itsekin) (10 p)

Kopioi linearisoinnin tulokset (tulokset tallentuvat tiedostoon partial_derivatives_RO.m) toiseen Matlab-pohjaan: solve_RO_ls.m.

Matlab-tehtävä: Täydennä annettu Matlab-pohja: solve_RO_ls.m. Lisää koodisi ja tulokset raporttiin. (10 p)

Eteenpäinleikkauksen ratkaisu (12 p)

Tällä kertaa tiedämme kahden kuvan ulkoiset (ja sisäiset) orientoinnit. Tavoitteena on laskea kuvahavainnoista (virittävät havaintovektorit kohti kohdetta) 3D kohdepisteiden koordinaatteja etsimällä havaintovektoreiden leikkauspiste. Tähän tehtävään käytetään kollineaarisuusyhtälöitä.

$$\begin{cases} x = -c\dfrac{r_{11}(X - X_0) + r_{12}(Y - Y_0) + r_{13}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)} = f_x \\ y = -c\dfrac{r_{21}(X - X_0) + r_{22}(Y - Y_0) + r_{23}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)} = f_y \end{cases}$$

Yhtälöt kehitetään seuraavaan muotoon:

$$\begin{cases} (xr_{31} + cr_{11})X + (xr_{32} + cr_{12})Y + (xr_{33} + cr_{13})Z = (xr_{31} + cr_{11})X_0 + (xr_{32} + cr_{12})Y_0 + (xr_{33} + cr_{13})Z_0 \\ (yr_{31} + cr_{21})X + (yr_{32} + cr_{22})Y + (yr_{33} + cr_{23})Z = (yr_{31} + cr_{21})X_0 + (yr_{32} + cr_{22})Y_0 + (yr_{33} + cr_{23})Z_0 \end{cases}$$

Kaikki tuntemattomia parametreja sisältävät termit on ryhmitelty yhtäsuuruusmerkin vasemmalle puolelle ja “vakiotermit” oikealle puolelle. Koska kaikki kierrot ovat tunnettuja, yhtälö on lineaarinen ja saamme suoran tuloksen (ilman lähtölikiarvoja ja iterointia).

K: Mitkä ovat tuntemattomat parametrit, jotka haluamme ratkaista? (1 p)

Rakennematriisi (A) on Jakobin matriisi (kuten edellisissäkin tapauksissa) ja y sisältää havainnot (kaikki yhtälön yhtäsuuruusmerkin oikealla puolella olevat termit). $v = Ax - y$

K: Kirjoita matriisin A sisältö symbolein (kuten ulkoisen orientoinnin tapauksessa tehtiin) yhden vastinpistehavainnon tapauksessa.

(käytä indeksejä f_{x1}, f_{y1}, f_{x2}, f_{y2} erottamaan vasemman kuvan (1) ja oikean kuvan (2) kuvahavaintoja)

Muistutuksena, miltä tilanne näytti ulkoisen orientoinnin tapauksessa::

$$A = \begin{bmatrix} \dfrac{\partial f_x}{\partial \omega} & \dfrac{\partial f_x}{\partial \varphi} & \dfrac{\partial f_x}{\partial \kappa} & \dfrac{\partial f_x}{\partial X_0} & \dfrac{\partial f_x}{\partial Y_0} & \dfrac{\partial f_x}{\partial Z_0} \\ \dfrac{\partial f_y}{\partial \omega} & \dfrac{\partial f_y}{\partial \varphi} & \dfrac{\partial f_y}{\partial \kappa} & \dfrac{\partial f_y}{\partial X_0} & \dfrac{\partial f_y}{\partial Y_0} & \dfrac{\partial f_y}{\partial Z_0} \end{bmatrix}$$

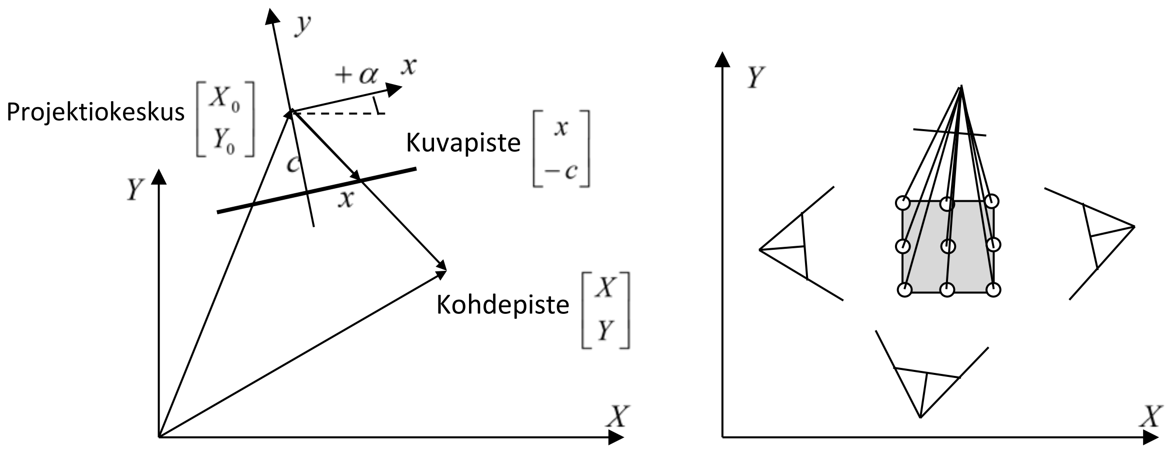
K: Mikä on systeemin ylimääritys? (1 p)

Matlab-tehtävä: Täydennä annettu Matlab-pohja: space_intersection.m. Liitä koodisi ja tulokset raporttiin. (10 p)

Sädekimpputasoitus (25 p)

Tässä tehtävässä tutkimme sädekimpputasoitusta erikoistilanteessa, jossa

- Kamera on yksiulotteinen (rivikamera) ja
- Kohdetila on 2D (XY taso)



Kuva 1 Kuva 2

Tässä tapauksessa kollineaarisuusyhtälöstä tulee:

$$\begin{bmatrix} x \\ -c \end{bmatrix} = k\begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}\begin{bmatrix} X - X_0 \\ Y - Y_0 \end{bmatrix} = kR\begin{bmatrix} X - X_0 \\ Y - Y_0 \end{bmatrix} \text{ eli}$$

$$x = -c\frac{(X - X_0)\cos\alpha + (Y - Y_0)\sin\alpha}{-(X - X_0)\sin\alpha + (Y - Y_0)\cos\alpha} = -c\frac{U}{W}$$

K: Mille kohdekoordinaatiston muunnokselle kuvahavainnot ovat tässä tapauksessa invariantteja? (1 p)

Tehtävä. Linearisoi (eli laske osittaisderivaatat kunkin tuntemattoman parametrin suhteen) tämä 2D kollineaarisuusyhtälö käyttäen 1. asteluvun Taylorin sarjakehitelmää (voit tehdä kynällä ja paperilla tai sitten Matlabin symbolisella laskennalla) (5 p)

$$x = x^0 + \frac{\partial x^0}{\partial\alpha}d\alpha + \frac{\partial x^0}{\partial X_0}dX_0 + \frac{\partial x^0}{\partial Y_0}dY_0 + \frac{\partial x^0}{\partial X}dX + \frac{\partial x^0}{\partial Y}dY$$

(Osittaisderivaatat voi laskea erikseen esim. $\partial x / \partial_{X_0}\alpha$ jne). Jos teet derivoinnin kynällä ja paperilla, voit käyttää seuraavaa derivoimissääntöä: $\dfrac{\partial x}{\partial p_k} = -c\left(\dfrac{\partial U}{\partial p_k}W - \dfrac{\partial W}{\partial p_k}U\right)/W^2 = \dfrac{-c}{W}\left(\dfrac{\partial U}{\partial p_k} - \dfrac{U}{W}\dfrac{\partial W}{\partial p_k}\right)$

K: Kuinka monta parametria tarvitaan kuvan *ulkoisen* orientoinninmäärittämiseksi? (1 p)

K: Kuinka monta kohdepistettä pitää vähintään tuntea kuvan *ulkoisen orientoinnin* parametrien ratkaisemiseksi (tässä erikoistapauksessa)? (1 p)

K. Mikä on (a) *havaintojen lukumäärä*, (b) *tuntemattomien parametrien lukumäärä* ja (c) *ylimääritys* kuvan 2 blokin (neljä kuvaa ja yhdeksän kohdepistettä) *sädekimpputasoituksessa*, kun oletetaan, että kaikki kohdepisteet ovat tuntemattomia ja että kaikki kohdepisteet näkyvät kaikilla kuvilla? (3 p)

K: Mikä on *datumvaje* edellisen kysymyksen blokissa? Entä rakennematriisin *ranki*? (2 p)

K. Yksinkertaisin keino poistaa datumvaje on ”kiinnittää” osa kohdepisteistä. Kuinka monta pistettä/koordinaattia tässä tapauksessa pitää vähintään kiinnittää? Kirjoita tarvittavat lisävirheyhtälöt kohdekoordinaateille (ulkorajoiteyhtälöt) $E^T x=0$. (4 p)

K. Datumvaje voidaan poistaa myös *sisärajoitteiden* (*inner constraints*) avulla. Kirjoita *sisärajoiteyhtälöt* tässä tapauksessa $E^T x=0$. (Yksinkertaisuuden vuoksi datumin määrittelyyn otetaan mukaan vain kohdepisteet, ei ulkoisia orientointeja) Käytä seuraavia yhtälöitä tuloksen johtamiseen: (3 p)

$$\begin{bmatrix} dX \\ dY \end{bmatrix} = \begin{bmatrix} dX_0 \\ dY_0 \end{bmatrix} + \begin{bmatrix} 0 & d\alpha \\ -d\alpha & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} + d\lambda \begin{bmatrix} X \\ Y \end{bmatrix}$$

Tehtävä. Kirjoita vapaaverkkotasoituksen rakenne (matriisit A ja vektorit Δ sekä f) käyttäen symboleja (kuten aiemmin ulkoisen orientoinnin tapauksessa), kun blokissasi on kaksi kuvaa ja neljä kohdepistettä. Vapaaverkkotasoitus vaatii joko sisä- tai ulkorajoiteyhtälöt, jotka jo ratkaisit edellä. Emme ratkaise käytännössä systeemiä tämän harjoituksen puitteissa, mutta se ratkeaisi yhtälöillä: (5 p)

$$\begin{bmatrix} N & E \\ E^T & 0 \end{bmatrix} \begin{bmatrix} \Delta \\ k \end{bmatrix} = \begin{bmatrix} u \\ 0 \end{bmatrix} \quad \text{jossa} \quad N = A^T P A \qquad u = A^T P f$$

Liite 1. Symbolinen osittaisderivointi tuntemattomien parametrien suhteen Matlabissa

Käytä annettua Matlab-pohjaa symbolisten derivointien luomiseksi:
`make_all_partial_derivates_A_matrix.m`

Tiedostossa on useampia funktioita. Ensimäinen funktio make_all_partial_derivates_A_matrix kutsuu kaikkia muita funktioita ja kirjoittaa tulokset uuteen tiedostoon: partial_derivatives.m. Tehtävänäsi on täydentää kaikki alifunktiot, esim.

```
function [ fx_d_o ] = derive_fx_d_o()
function [ fy_d_o ] = derive_fy_d_o()
```

jne.

Vaikka alifunktioita on 12 täydennettäväksi, pari ensimmäistä ovat tärkeimmät. Käytännössä, jos saat nämä kaksi funktiota kuntoon, loput saat nopeasti valmiiksi “copy-paste”-tyylillä. Näihin pitää vaihtaa vain muutama symboli.

Matlabissa symboliset muuttujat alustetaan komennolla “syms”. Tämä tulee tehdä erikseen jokaisessa funktiossa, jossa niitä käytetään. Esimerkiksi, jos funktiossa tarvittaisiin symbolisia muuttujia o=omega, p=fii ja k=kappa, nämä voitaisiin alustaa seuraavasti:

```
syms o p k;
```

Initialisoi symbolit omega, phi, kappa, X0, Y0, Z0, c, X, Y, Z, r11, r21, r31, r21, r22, r23, r31, r32, r33, joita tarvitaan kollineaarisuusyhtälöissä. Aloita rakentamalla symbolisesti ensin 3D kiertomatriisin elementit (malli on annettu pohjassa), koska sitten on helpompi rakentaa kollineaarisuusyhtälöt. Esimerkiksi ensimmäinen kiertomatriisin elementti on

```
r11=cos(p)*cos(k);
```

Seuraavaksi kirjoita kollineaarisuusyhtälö. Funktiossa derive_fx_d_o (ja kaikissa, missä esiintyy nimessä fx) sinun tarvii käyttää vain kollineaarisuusyhtälön ylempää yhtälöä.

fx=-c*(r11*…

Derivoi tämä funktiossa ajatellun tuntemattoman parametrin suhteen. Esimerkiksi derive_fx_d_o tuntematon parametri on o=omega. Käytä Matlabin funktiota diff() itse derivoinnin muodostamiseksi ja tallenna (symbolinen) tulos funktiosta ulos lähtevään parametriin, esim. fuktiossa

```
function [ fx_d_o ] = derive_fx_d_o()
 %fx_d_o on parametri, jonka funktio palauttaa ja siksi sijoita tulos:
```

```
fx_d_o=diff(…
```

Kun olet onnistunut viimeistelemään kaikki funktiot, aja funktio make_all_partial_derivates_A_matrix 

. Avaamalla tiedoston “partial_derivatives.m” näet useita funktioita, joihin derivointien tulokset on kirjotettu. Nämä ovat sopivassa muodossa ajateltuna varsinaista laskentafunktiota, joihin niitä on tarkoitus käyttää. Kopioi kaikki tulosfunktiot ja liitä ne tiedostoon “check_partial_derivatives.m”. Voit tällä fuktiolla tarkistaa (vain ulkoisen orientoinnin tapaus), onnistuitko saamaan oikean lopputuloksen vai onko jossain alifunktiossa ongelmia.