Appendices

Appendix(A)

%This program calculate corrections of coordinates, variance covariance,
%standard error, and mean standard error. getvals from txt file samples.
Program written in Matlab language.

A: is the coefficient matrix of parameters.

b: is the vector matrix (o – c).

w: is the weight matrix.

N: is the variance covariance matrix.

D: is the check (unit matrix).

S: is the standard error.

M.S.E: mean standard error.

```
%This  getvals () function works for 5 elements
CLOSE ALL;
%function [E, N, Z] = getvals (j , Samples);
%for i=1:5;
%Initialize variables
%x(:,i)=[E(i); N(i); Z(i)];
syms E1 E2 E3 E4 E5 N1 N2 N3 N4 N5 Z1 Z2 Z3 Z4 Z5
f1=(atan((E2-E1)/(N2-N1)))-(atan((E3-E1)/(N3-N1)));
%syms E N
f2=(atan((E3-E2)/(N3-N2)))-(atan((E1-E2)/(N1-N2)));
%syms E N
f3=(atan((E1-E3)/(N1-N3)))-(atan((E2-E3)/(N2-N3)));
%syms E N Z
f4=asin((Z3-Z1)/((E3-E1)^2+(N3-N1)^2+(Z3-Z1)^2)^0.5);
%syms E N Z
```

```
f5=asin((Z3-Z2)/((E3-E2)^2+(N3-N2)^2+(Z3-Z2)^2)^0.5);
%syms E N
f6=atan((E3-E1)/(N3-N1));
%syms E N
f7=atan((E3-E2)/(N3-N2));
%syms E N
f8=((atan((E4-E2)/(N4-N2)))-(atan((E3-E2)/(N3-N2))));
%syms E N
f9=(atan((E2-E3)/(N2-N3)))-(atan((E4-E3)/(N4-N3)));
%syms E N
f10=(atan((E3-E4)/(N3-N4)))-(atan((E2-E4)/(N2-N4)));
%syms E N Z
f11=asin((Z4-Z2)/((E4-E2)^2+(N4-N2)^2+(Z4-Z2)^2)^0.5);
%syms E N Z
f12=asin((Z4-Z3)/((E4-E3)^2+(N4-N3)^2+(Z4-Z3)^2)^0.5);
%syms E N
f13=((atan((E4-E3)/(N4-N3)))-(atan((E5-E3)/(N5-N3))));
%syms E N
f14=(atan((E3-E5)/(N3-N5)))-(atan((E4-E5)/(N4-N5)));
%syms E N
f15=(atan((E5-E4)/(N5-N4)))-(atan((E3-E4)/(N3-N4)));
%syms E N Z
f16=asin((Z5-Z3)/((E5-E3)^2+(N5-N3)^2+(Z5-Z3)^2)^0.5);
%syms E N Z
f17=asin((Z4-Z5)/((E4-E5)^2+(N4-N5)^2+(Z4-Z5)^2)^0.5);
```

```
A=[diff(f1,E3) diff(f1,N3) diff(f1,Z3) diff(f1,E4) diff(f1,N4) diff(f1,Z4) diff(f1,E5) diff(f1,N5) diff(f1,Z5);
   diff(f2,E3) diff(f2,N3) diff(f2,Z3) diff(f2,E4) diff(f2,N4)  diff(f2,Z4)  diff(f2,E5)  diff(f2,N5)  diff(f2,Z5);
   diff(f3,E3) diff(f3,N3) diff(f3,Z3) diff(f3,E4) diff(f3,N4)  diff(f3,Z4)  diff(f3,E5)  diff(f3,N5)  diff(f3,Z5);
   diff(f4,E3) diff(f4,N3) diff(f4,Z3) diff(f4,E4) diff(f4,N4)  diff(f4,Z4)  diff(f4,E5)  diff(f4,N5)  diff(f4,Z5);
   diff(f5,E3) diff(f5,N3) diff(f5,Z3) diff(f5,E4) diff(f5,N4)  diff(f5,Z4)  diff(f5,E5)  diff(f5,N5)  diff(f5,Z5);
   diff(f6,E3) diff(f6,N3) diff(f6,Z3) diff(f6,E4) diff(f6,N4)  diff(f6,Z4)  diff(f6,E5)  diff(f6,N5)  diff(f6,Z5);
   diff(f7,E3) diff(f7,N3) diff(f7,Z3) diff(f7,E4) diff(f7,N4)  diff(f7,Z4)  diff(f7,E5)  diff(f7,N5)  diff(f7,Z5);
   diff(f8,E3) diff(f8,N3) diff(f8,Z3) diff(f8,E4) diff(f8,N4)  diff(f8,Z4)  diff(f8,E5)  diff(f8,N5)  diff(f8,Z5);
   diff(f9,E3) diff(f9,N3) diff(f9,Z3) diff(f9,E4) diff(f9,N4)  diff(f9,Z4)  diff(f9,E5)  diff(f9,N5)  diff(f9,Z5);
   diff(f10,E3)diff(f10,N3)diff(f10,Z3)diff(f10,E4)diff(f10,N4)diff(f10,Z4)diff(f10,E5)diff(f10,N5) diff(f10,Z5);
   diff(f11,E3)diff(f11,N3)diff(f11,Z3)diff(f11,E4)diff(f11,N4)diff(f11,Z4)diff(f11,E5)diff(f11,N5) diff(f11,Z5);
   diff(f12,E3)diff(f12,N3)diff(f12,Z3)diff(f12,E4)diff(f12,N4)diff(f12,Z4)diff(f12,E5)diff(f12,N5) diff(f12,Z5);
diff(f13,E3)diff(f13,N3)diff(f13,Z3)diff(f13,E4)diff(f13,N4)diff(f13,Z4)diff(f13,E5)diff(f13,N5) diff(f13,Z5);
   diff(f14,E3)diff(f14,N3)diff(f14,Z3)diff(f14,E4)diff(f14,N4)diff(f14,Z4)diff(f14,E5)diff(f14,N5) diff(f14,Z5);
diff(f15,E3)diff(f15,N3)diff(f15,Z3)diff(f15,E4)diff(f15,N4)diff(f15,Z4)diff(f15,E5)diff(f15,N5) diff(f15,Z5);
   diff(f16,E3)diff(f16,N3)diff(f16,Z3)diff(f16,E4)diff(f16,N4)diff(f16,Z4)diff(f16,E5)diff(f16,N5) diff(f16,Z5);
   diff(f17,E3)diff(f17,N3)diff(f17,Z3)diff(f17,E4)diff(f17,N4)diff(f17,Z4)diff(f17,E5)diff(f17,N5) diff(f17,Z5)]
```

```
Load  Samples.txt;
   E= [Samples (:, 1)];
   E1=E(1);
   E2=E(2);
   E3=E(3);
   E4=E(4);
   E5=E(5);
   N = [Samples (:, 2)];
   N1=N(1);
   N2=N(2);
   N3=N(3);
   N4=N(4);
   N5=N(5);
   Z = [Samples (:, 3)'];
```

```
Z1=Z(1);
Z2=Z(2);
Z3=Z(3);
Z4=Z(4);
Z5=Z(5);
a =  [Samples (:, 4)];
b =  [Samples (:, 5)];
c =  [Samples (:, 6)];
d =  [Samples (:, 7)];
e =  [Samples (:, 8)];
f =   [Samples (:, 9)];
g =  [Samples (:, 10)];
h =  [Samples (:, 11)];
i =   [Samples (:, 12)];
j =   [Samples (:, 13)];
k =  [Samples (:, 14)];
l =   [Samples (:, 15)];
m = [Samples (:, 16)];
n =  [Samples (:, 17)];
u =  [Samples (:, 18)];
p =  [Samples (:, 19)];
w = [Samples (:, 20)];

A=eval(A);
```

```
W=[0.01   0   0   0    0    0    0   0   0   0   0    0   0   0   0    0    0;
   0  0.01 0   0    0    0    0   0   0   0   0    0   0   0   0    0    0;
   0   0  0.01 0    0    0    0   0   0   0   0    0   0   0   0    0    0;
   0   0   0 0.0025 0    0    0   0   0   0   0    0   0   0   0    0    0;
   0   0   0   0 0.0025 0    0   0   0   0   0    0   0   0   0    0    0;
   0   0   0   0    0  0.01    0   0   0   0   0    0   0   0   0    0    0;
   0   0   0   0    0   0  0.01  0   0   0   0    0   0   0   0    0    0;
   0   0   0   0    0   0   0 0.01  0   0   0    0   0   0   0    0    0;
   0   0   0   0    0   0   0   0 0.01  0   0    0   0   0   0    0    0;
   0   0   0   0    0   0   0   0   0 0.01  0    0   0   0   0    0    0;
   0   0   0   0    0   0   0   0   0   0 0.0025 0   0   0   0    0    0;
   0   0   0   0    0   0   0   0   0   0   0 0.0025 0   0   0    0    0;
   0   0   0   0    0   0   0   0   0   0   0    0 0.01  0   0    0    0;
   0   0   0   0    0   0   0   0   0   0   0    0   0 0.01  0    0    0;
   0   0   0   0    0   0   0   0   0   0   0    0   0   0 0.01   0    0;
   0   0   0   0    0   0   0   0   0   0   0    0   0   0   0 0.0025 0;
   0   0   0 0 0   0   0   0   0   0   0   0    0   0   0   0    0 0.0025]*((206265)^2);
W=W;

b=[(a(1)-a(2));
  (b(1)-b(2));
  (c(1)-c(2));
  (d(1)-d(2));
  (e(1)-e(2));
  (f(1)-f(2));
  (g(1)-g(2));
  (h(1)-h(2));
  (i(1)-i(2));
  (j(1)-j(2));
```

```
   (k(1)-k(2));
   (l(1)-l(2));
   (m(1)-m(2));
   (n(1)-n(2));
   (u(1)-u(2));
   (p(1)-p(2));
   (w(1)-w(2))]*(0.017453292);
A=A;
D=(A'*W*A);
N= (A'*W*A) ^-1;
T= (N*D);
U= (A'*W*b);
Xre= (N*U);
V= (A*Xre)-b;
Check = (A'*W*V);
S1= (N (1, 1) ^0.5);
S2= (N (2, 2) ^0.5);
S3= (N (3, 3) ^0.5);
S4= (N (4, 4) ^0.5);
S5= (N (5, 5) ^0.5);
S6= (N (6, 6) ^0.5);
S7= (N (7, 7) ^0.5);
S8= (N (8, 8) ^0.5);
S9= (N (9, 9) ^0.5);
M.S.E= ((S1+S2+S3+S4+S5+S6+S7+S8+S9)/9);
```