APPENDIX A. Computer Programs

In this appendix listings of all the computer programs (codes) referred to in the main text are provided. The programs are documented in the same order they were referenced in the text. The programs were written in both Fortran 90 and MATLAB programming languages. Programs or parts of programs (either subroutines or section of programs) from other sources are referenced accordingly. Some programs use only particular input formats and must be used accordingly.

A.1 A Fortran program to assign magnetic data (from cesium magnetometer file) and geographic positions (from the Garmin GPS file)

Author: M. Muundjua and A. Galdeano
March 2005

```
real Anomaly,MAG_time,MAG_value,MAG_lat,MAG_long,MAG_dist
parameter (max_gps_pts=50000,max_mag_pts=50000)
dimension GPS_time(max_gps_pts),GPS_long(max_gps_pts)
dimension GPS_lat(max_gps_pts),speed(max_gps_pts)

dimension MAG_time(max_mag_pts),MAG_lat(max_mag_pts)
dimension MAG_value(max_mag_pts),MAG_long(max_mag_pts)
dimension MAG_dist(max_mag_pts), Anomaly(max_mag_pts)

character nf_GPS*120,nf_MAG*120,nf_out*120
character line*150,line1*30,line2*30,line3*30,SN*1,EW*1,fmt*100
character GPS_dir*120,MAG_dir*120,out_dir*120
character date0*6,date1*8

GPS_dir='C:\Parys_data\GPS_data\GPS_raw\'
MAG_dir='C:\Parys_data\Magnetic_data\Mag_daily_stations\mag_raw\'
out_dir='C:\Parys_data\Magnetic_data\Mag_daily_stations\mag_raw\'

iun_mag = 11
iun_gps = 12
iun_out = 13

l_GPS_dir = length(GPS_dir)
l_MAG_dir = length(MAG_dir)
l_out_dir = length(out_dir)

write(*,'(a$)') ' day, month and year (ddmmyy) : '
read(*,'(a)') date0
if(date0.eq.' ') then
```

```
flag = 1
else
flag = 0
date1(1:2) = date0(1:2)
date1(3:3) = '_'
date1(4:5) = date0(3:4)
date1(6:6) = '_'
date1(7:8) = date0(5:6)
endif

100 continue
if(flag.eq.0) then
nf_GPS = 'gps_raw_00_00_00.txt'
nf_GPS(9:16) = date1
line = nf_GPS
l_nf_GPS = length(line)
nf_GPS = GPS_dir
nf_GPS(l_GPS_dir+1:l_GPS_dir+l_nf_GPS) = line
else
write(*,'(a$)') ' GPS file : '
read(*,'(a)') nf_GPS
if(nf_GPS .eq. ' ') nf_GPS = 'gps_raw_22_02_05.txt'
endif

open(iun_gps,file=nf_GPS,err=100)

110 continue
if(flag.eq.0) then
nf_MAG = 'mag_raw_00_00_00.stn'
nf_MAG(9:16) = date1
line = nf_MAG
l_nf_MAG = length(line)
nf_MAG = MAG_dir
nf_MAG(l_MAG_dir+1:l_MAG_dir+l_nf_MAG) = line
else
write(*,'(a$)') ' MAG file : '
read(*,'(a)') nf_MAG
if(nf_MAG .eq. ' ') nf_MAG = 'mag_raw_22_02_05.stn'
endif

open(iun_mag,file=nf_MAG,err=110)

120 continue
if(flag.eq.0) then
nf_out = 'mag_geog_00_00_00.dat'
nf_out(10:17) = date1
```

```
line = nf_out
l_nf_out = length(line)
nf_out = out_dir
nf_out(l_out_dir+1:l_out_dir+l_nf_out) = line
else
write(*,'(a$)') ' OUTPUT file : '
read(*,'(a)') nf_out
if(nf_out .eq. ' ') nf_out = 'test.dat'
endif

open(iun_out,file=nf_out,err=120)

! reading the MAG values
n_mag = 0
200 continue
read(iun_mag,'(a)',end=201) line
if (line(1:1) .eq. '0') then
read(line,'(4x,f9.3,12x,f2.0,1x,f2.0,1x,f5.2)')
_           xmag,th,tm,ts
time = th*3600+tm*60+ts
n_mag = n_mag+1
if(n_mag . gt. max_mag_pts) then
write(*,*) ' *** MAG : problem of dimension '
pause
stop
endif
MAG_value(n_mag) = xmag
MAG_time(n_mag) = time
endif
goto 200
201 continue
close(iun_mag)

! reading the GPS values
n_gps = 0
210 continue
read(iun_gps,'(a)',end= 211) line
if (line(1:10) .eq. 'Trackpoint') then
line1 = line(12:21)
line2 = line(23:32)
line3 = line(45:52)
!fmt ='(12x,a1,i2,f7.3,1x,a1,i2,f7.3,12x,f2.0,1x,f2.0,1x,f2.0)'
!read(line,fmt)SN,la,xla_mn,EW,lo,xlo_mn,th,tm,ts
read(line1,'(a1,i2,f7.3)') SN,la,xla_mn
read(line2,'(a1,i2,f7.3)') EW,lo,xlo_mn
read(line3,'(f2.0,1x,f2.0,1x,f2.0)') th,tm,ts
```

```
time = th*3600+tm*60+ts
n_gps = n_gps+1
if(n_gps . gt. max_gps_pts) then
write(*,*) ' *** GPS : problem of dimension '
pause
stop
endif
GPS_time(n_gps) = time
xxlo = lo+xlo_mn/60.
if(EW .eq. 'W') xxlo = -xxlo
GPS_long(n_gps) = xxlo
xxla = la+xla_mn/60.
if(SN .eq. 'S') xxla = -xxla
GPS_lat(n_gps) = xxla
endif
goto 210
211 continue
close(iun_gps)

! computation of the speed along the track (in degrees/s !!!)
do i=1,n_gps-1
dlong = GPS_long(i+1)-GPS_long(i)
dlat  = GPS_lat(i+1)-GPS_lat(i)
dist = sqrt(dlong*dlong+dlat*dlat)
dt = GPS_time(i+1)-GPS_time(i)
speed(i) = dist/dt
enddo

! estimation of the mean speed
speed_m = 0.
do i=1,n_gps-1
speed_m = speed_m + speed(i)
enddo
speed_m = speed_m / n_gps

i_ret = 0
do i=1,n_gps-1
if (speed(i).ne.0. .and. speed(i).lt. 4.*speed_m ) then
i_ret = i_ret + 1
GPS_long(i_ret) = GPS_long(i)
GPS_lat(i_ret)  = GPS_lat(i)
GPS_time(i_ret) = GPS_time(i)
endif
enddo

! NEW VALUE FOR N_GPS
```

```
n_gps = i_ret

! reverse the mag values/times in order to have an incresing order in time
do i=1,n_mag/2
aux = MAG_value(i)
MAG_value(i) = MAG_value(n_mag-i+1)
MAG_value(n_mag-i+1) = aux
aux = MAG_time(i)
MAG_time(i) = MAG_time(n_mag-i+1)
MAG_time(n_mag-i+1) = aux
enddo

write(*,*) ' MAG time', n_mag,mag_time(1),mag_time(n_mag)
write(*,*) ' GPS time', n_gps,gps_time(1),gps_time(n_gps)

! loop through the mag value and find the corresponding time into
! gps vector by interpolation
! then use interpolated time to assign position to mag value
do imag=1,n_mag
t_mag = MAG_time(imag)
if((t_mag-GPS_time(1))*(t_mag-GPS_time(n_gps)).le.0.) then
do igps=1,n_gps
if(t_mag .lt. GPS_time(igps)) go to 500
enddo  ! end of GPS loop
write(*,*) ' *** ERROR in picking up the GPS time '
500     continue

d_time1 = GPS_time(igps) - GPS_time(igps-1)
d_time2 = t_mag - GPS_time(igps-1)

! interpolation for latitude value
xlat1 = GPS_lat(igps-1)
xlat2 = GPS_lat(igps)
MAG_lat(imag) = xlat1 + (xlat2-xlat1)*d_time2/d_time1

! interpolation for lomgitude value
xlong1 = GPS_long(igps-1)
xlong2 = GPS_long(igps)
MAG_long(imag) = xlong1 + (xlong2-xlong1)*d_time2/d_time1
else
! HERE the mag point is out of the time interval defined by GPS tracks
! the value is forced to -1 in order to don't write it on the output file
MAG_value(imag) = -1.
endif
enddo  ! end of mag loop
```

```
dist = 0.
do i=1,n_mag-1
dlong = Mag_long(i+1) - Mag_long(i)
dlat = Mag_lat(i+1) - Mag_lat(i)
dd = sqrt(dlong*dlong + dlat*dlat)
dist = dist + dd
MAG_dist(i) = dist * 111.11111
enddo
MAG_dist(n_mag) = (dist + dd) * 111.11111

open(iun_out,file=nf_out)

sum = 0
do i=1,n_mag
sum = sum+Mag_value(i)
enddo
mean = sum/n_mag

do i=1,n_mag
Anomaly(i) = Mag_value(i) - mean
enddo

! out put of mag values associated to long-lat positions
do i=1,n_mag

time = MAG_time(i)
ih = int(time/3600.)
r = time - ih*3600
mn = int (r/60)
sec = r - mn*60
field = Mag_value(i)
!if(field.gt.18000. .and. field.lt.40000) then
if(field.gt.1) then
write (iun_out,'(2f12.5,f10.2,5x,1h",i2,1h:,i2,1h:,f5.2,2h"
_          ,f10.2,f12.5,f10.2)') Mag_long(i),Mag_lat(i),field
_          ,ih,mn,sec,time, Mag_dist(i),Anomaly(i)
endif

enddo

pause 'Press "Return" to continue'

stop
end
```

```
function length(c)
character*(*) c
do length=len(c),1,-1
if(c(length:length).ne.' ') return
enddo
length = 0
return
end
```

A.2 Codes for calculating AGC of map data

```
function [out1]=Grid_gain_Control(data,wsize);

% The following codes were taken from the website given and was applied as
% such
% http://utam.geophys.utah.edu/stanford/node14.html
% Gerard Schuster 1998-07-29
% Note that the AGC depends on the size of the window, wsize
% data must be a matrix

np = wsize;
[nx,nt]=size(data);
inn=[fliplr(data(:,1:np)) data fliplr(data(:,nt-np:nt))];
inn=inn.^2;
f=ones(np,1)/np;
out=conv2(inn,f');
start=round(np+np/2);endd=start+nt-1;
dataAGCtt=dataAGCout(:,start:endd);
dataAGCout1=data./(sqrt(dataAGCtt)+.000001);

figure(1)
subplot(2,1,1); imagesc(data); axis tight; title('Raw data');
subplot(2,1,2); imagesc(dataAGCout1); axis tight; title('AGC of map data');
```

A.3 Code for calculating local depth using the wavenumber method proposed by Thurston and Smith (1997).

Author: M Muundjua and Armand Galdeano
March 2006
Inputs magnetic anomaly data and the codes return the local depth

```
function [hal,wavnum,theta] = apparent(data);
```

```
x=data(:,1); data=data(:,2);
[n,m]=size(data);% give the size of data i.e, n=max(size(data));

dtr=pi/180;

% Portion of codes below are due to Gordon Cooper (personal communications)

GeoInc=75*dtr; GeoDec=0*dtr; azimuth=0*dtr;
%return
npts=2^nextpow2(n);
xint=abs(x(2)-x(1));
ndiff=(npts-n)/2;
%data Padding (to 2^N points)
ndiff=round(npts-n)/2;%Round towards nearest integer.
datapadd=zeros(npts,m);
datapadd(ndiff+1:ndiff+n)=data;% zero padding
xpadd=zeros(npts,m);
xpadd(ndiff+1:ndiff+n)=x; %1:length(data);
n=length(data); deltax=abs(x(2)-x(1));
fftdata = fft(datapadd,npts);

 dx=gradient(datapadd(ndiff+1:ndiff+n),deltax); dz=imag(hilbert(dx));M=dz;
 dxx = gradient(dx,deltax); dxz = imag(hilbert(dxx)); dzz = imag(hilbert(dxz));M1=dzz;
 A = sqrt(dx.^2+dz.^2);A1 =sqrt(dxx.^2+dzz.^2);
 gdiff = (dxz.*dx - dxx.*dz);

% Portion of codes above are due to Gordon Cooper (personal communications)

%A =  sqrt(dz.^2+dz.^2);
dmx = gradient(M,deltax); dmz = imag(hilbert(dmx));
dmx1 = gradient(M1,deltax); dmz1 = imag(hilbert(dmx1));

A0 = sqrt(dmx.^2+dmz.^2); A1 = sqrt(dmx1.^2+dmz1.^2);
figure(2)

 h = zeros(n,1); wavnum = zeros(n,1); theta = zeros(n,1);
 for i=1:n
    k = (gdiff(i)/(A(i)^2))/2; % wave number from Smith and Thurston (1997)
    wavnum(i) = k;
    h(i) = (1+sqrt(abs(1+4*k*x(i))))/(2*k);
    theta(i) = atan(dz(i)/dx(i)); % Calculating local phase
    %h(i) = 1/k;
 end
 [k,i] = max(wavnum); % wavenumber of deepest dyke at x = 0
h = 1/k  % depth to top body
```

A.4 Code for calculating magnetization of 2-dimensional bodies.

Author: M Muundjua and Armand Galdeano
June 2006
Inputs the magnetic anomaly, magnetization directions and the program returns the magnetization of prismatic bodies

```
function Tinv = ribbon(data);

n = length(data);
x = data(:,1);
Bz = data(:,2);

dtr = pi/180;
alpha = 50; % Profil azimuth from the North
% azimuth from the North of Profile 2 is 13 degrees
% azimuth from the North of Profile 3 is 50 degrees
GeoInc = -63.8; %IGRF % vertical component
GeoDec = -18.6;
mInc = -63.8;
mDec = -18.6;
mInc = 57; % Inc Dike&Pseudos magnetization
mDec = 20; % Dec Dike&Pseudos magnetization
% you need only one calcul for the cosinus directors to save speed because
% you called it at each time to product always the same values
ax=cos(GeoInc*dtr)*cos(GeoDec*dtr-alpha*dtr);
ay=cos(GeoInc*dtr)*sin(GeoDec*dtr-alpha*dtr);
az=sin(GeoInc*dtr);
xincl=mInc*dtr;
xdecl=mDec*dtr;
xazim=alpha*dtr;
mx=cos(mInc*cos(mDec*dtr-alpha*dtr)*dtr);
mz=sin(mInc*cos(mDec*dtr-alpha*dtr)*dtr);
m = 1;
%real data
% top = 0; bot = 5000; % Depth of each box (in meters)
% z0 = -40; % Observation altitude (in meters)
% width = 100; % width of each box
%synthetic
top = 0; bot = 5000; % Depth of each box (in meters)
z0 = -40; % Observation altitude (in meters)
width = 100; % width of each box
mini = min(x);
```

```
maxi = max(x); k = mini:width:maxi; GRsols = zeros(length(k),1);
grm=zeros(n,1);
ndata = 0;
side1 = zeros(4,1); side2 = zeros(4,1); side3 = zeros(4,1); side4 = zeros(4,1); midpoint =
zeros(2,1);
for k = mini:width:maxi%+2*width
   side1(1,:) = k; side1(2,:) = top; side1(3,:) = k+width; side1(4,:) = top;
   side2(1,:) = k+width; side2(2,:) = top; side2(3,:) = k+width; side2(4,:) = bot;
   side3(1,:) = k+width; side3(2,:) = bot; side3(3,:) = k; side3(4,:) = bot;
   side4(1,:) = k; side4(2,:) = bot; side4(3,:) = k; side4(4,:) = top;
   face = [side1 side2 side3 side4];
   Ab = zeros(n,1); npw=0;
   ndata = ndata + 1;

   for i=1:n
      npw = npw + 1;
      p = 0;totalsum = 0;
      for j=1:4
         x1 = face(1,j); z1=face(2,j); x2 = face(3,j); z2=face(4,j);
         [T,ier]=mag2d(x(i),z0,x1,z1,x2,z2,mx,mz,ax,ay,az);
         value = T;
         totalsum = totalsum + value;
      end

      Ab(npw) = totalsum;
   end
   grm(:,ndata) = Ab;
   midpoint(ndata) = (k+k+width)/2;
end
midpoint;
sols = grm\Bz;
%disp([midpoint,sols]);
disp([midpoint]);
disp([sols]);
m = length(sols);
Tinv=grm*sols; % Calculating magnetic response due to the source bodies
Bz-Tinv;
min(sols)
max(sols)
funct1=0.0; npw = 0;
for i=1:n
   npw = npw + 1;
   funct1=funct1+(Bz(npw)-Tinv(npw))^2;
end
fprintf('Misfit = %g\n',funct1);
subplot(2,2,1);plot(x,Bz);axis xy; axis tight;  % Original data
```

```
title('Data'); ylabel('Intensity (nT)');
subplot(2,2,2);plot(midpoint,sols);hold on; plot(20,0.8,'r*');plot(65,1.5,'r*'); axis xy; axis
tight; %axis([min(midpoint) max(midpoint) 1 23.5589])% Magnetizations of bodies
title('Magnetization'); ylabel('NRM (A/m)');
subplot(2,2,3);plot(x,Tinv);axis xy; axis tight;  % Recovered data
title('Recovered model'); ylabel('Intensity (nT)'); xlabel('Profile Distance (m)');
subplot(2,2,4);plot(x,Bz-Tinv);axis xy; axis tight; % Error
title('Error'); ylabel('Intensity (nT)'); xlabel('Profile Distance (m)');
return


%
************************************************************************
*
%
************************************************************************
*
function t=mbox(x0,y0,z0,x1,y1,z1,x2,y2,mi,md,fi,fd,m,theta)

*                                   *
Subroutine From Blakely (1996)

alpha = zeros(2,1);
beta = zeros(2,1);
cm = 1.e-7; t2nt = 1.e9;
[ma,mb,mc]=dircos(mi,md,0);
[fa,fb,fc]=dircos(fi,fd,0);
fm1=ma*fb+mb*fa;
fm2=ma*fc+mc*fa;
fm3=mb*fc+mc*fb;
fm4=ma*fa;
fm5=mb*fb;
fm6=mc*fc;
alpha(1)=x1-x0;
alpha(2)=x2-x0;
beta(1)=y1-y0;
beta(2)=y2-y0;
h=z1-z0;
t=0;
hsq=h^2;

for i=1:2
   alphasq=alpha(i)^2;
   for j=1:2
      sign=1;
      if(i~=j)sign=-1; end;
```

```
        r0sq=alphasq+beta(j)^2+hsq;
        r0=sqrt(r0sq);
        r0h=r0*h;
        alphabeta=alpha(i)*beta(j);
        arg1=(r0-alpha(i))/(r0+alpha(i));
        arg2=(r0-beta(j))/(r0+beta(j));
        arg3=alphasq+r0h+hsq;
        arg4=r0sq+r0h-alphasq;
        tlog=fm3*log(arg1)/2+fm2*log(arg2)/2-fm1*log(r0+h);
        tatan=-fm4*atan2(alphabeta,arg3)-
fm5*atan2(alphabeta,arg4)+fm6*atan2(alphabeta,r0h);
        t=t+sign*(tlog+tatan);
    end
end
t=t*m*cm*t2nt;


%
************************************************************************
*
%
************************************************************************
*
function [a,b,c]= dircos(incl,decl,azim)

Subroutine from Blakely (1996)

d2rad=0.017453293;
xincl=incl*d2rad;
xdecl=decl*d2rad;
xazim=azim*d2rad;
a=cos(xincl)*cos(xdecl-xazim);
b=cos(xincl)*sin(xdecl-xazim);
c=sin(xincl);
```

APPENDIX B. Paleomagnetic data

Table B1. NRM values for the 100 samples and the associated angle of magnetization vector. (Az, in situ azimuth of core; Dip, in-situ dip angle of the core; Dec, declination of magnetization vector; Inc, inclination of magnetization vector; α_{95} standard deviation in the computation of the NRM and the magnetization angles.)

The following is a list of the NRM values of the 100 samples over the 10 m x 10 m grid. The name of the samples (letters) correspond to the position on the ground from which the core was taken. The X and Y values are the same as the geomagnetic grid given in Chapter 4, Section 4.2. Sample C03 corresponds to the bottom left position (Fig. 4.24), Chapter 4, Section 4.2) and L12 correspond to the top right corner (Fig. 4.24).

Specimen	Az (˚)	Dip (˚)	NRM (A/m)	NRM Dec (˚)	NRM Inc (˚)	α_{95} (˚)	X (m)	Y (m)
C03	225.8	19.5	9.00	117.4	12.5	0.3	3	3
C04	220	15	8.50	115.5	-39.9	0.2	4	3
C05	263.1	17	8.60	313.8	-13.4	0.2	5	3
C06	6.4	20	12.00	275.7	-63.8	0.1	6	3
C07	75.1	17.5	1.70	349.9	23.7	0.2	7	3
C08	125	12	6.60	307.5	-60.6	0.2	8	3
C09	121.8	13	49.00	353.5	82.4	0.1	9	3
C10	81.7	22.5	19.00	113.9	33.4	0.1	10	3
C11	65.2	39.5	100.00	253.7	-38	0.2	11	3
C12	60.8	29	0.16	142.9	6.9	0.3	12	3
D03	47.6	14	9.30	138.9	-2	0.4	3	4
D04	89	13.5	38.00	160.5	57.2	0.2	4	4
D05	74	13	6.70	306.9	-40.3	0.4	5	4
D06	99.6	11.5	4.30	107.2	-71.5	0.1	6	4

Continuation of Table B1

D07	83.6	17.5	8.80	275	-35.3	0.2	7	4
D08	83.8	13	16.00	311.9	-61.1	0.2	8	4
D09	92.4	17	88.00	333.6	74.5	0.4	9	4
D10	101	20	20.00	329.6	-39.8	0.4	10	4
D11	103.3	29	110.00	287.6	39.2	0.4	11	4
D12	83	36.5	83.00	26.7	30.9	0.2	12	4
E03	68.4	16	17.00	162.6	-71.5	0.1	3	5
E04	61.7	21	3.30	145.2	24.4	0.4	4	5
E05	53.9	17.5	11.00	282.3	-12.8	0.4	5	5
E06	68.7	18.5	4.00	163.9	-56.7	0.2	6	5
E07	98.5	19	66.00	128.4	-71.2	0.3	7	5
E08	79.7	20.5	5.50	257.9	-17.1	0.2	8	5
E09	103	17	21.00	331	23.3	0.5	9	5
E10	99.6	26	31.00	117.2	-55	0.4	10	5
E11	102.4	27	8.20	120.3	65.4	0.1	11	5
E12	313.9	23	17.00	246.1	-10.6	0.3	12	5
F03	337.4	19	0.74	230.3	-31	0.4	3	6
F04	35.1	31	54.00	130.8	31.5	0.5	4	6
F05	9.2	28	14.00	325.3	33.9	0.4	5	6
F06	47	25.5	4.00	68.5	-40.5	0.3	6	6
F07	85.2	23	1.60	164.9	-72.4	0.3	7	6
F08	122.7	29	250.00	347.3	-50.3	0.4	8	6
F09	48.4	20	92.00	318.6	19.8	0.1	9	6
F10	100.6	21	2.80	119.2	46.3	0.2	10	6
F11	146.6	20.5	8.30	200.4	18.1	0.45	11	6
F12	143.9	25	2.00	218.9	15	0.35	12	6

Continuation of Table B1

G03	302.5	20	3.90	131.1	-10.8	0.5	3	7
G04	326.7	22.5	13.00	119.3	56.6	0.4	4	7
G05	347.4	23.5	17.00	168.2	-52.1	0.3	5	7
G06	36.3	21	1.70	126.2	63.3	0.4	6	7
G07	77.7	27	5.40	108	6.7	0.4	7	7
G08	28.4	35.5	27.00	133.7	-16.8	0.2	8	7
G09	93.3	29	26.00	324.2	-34.2	0.4	9	7
G10	85	24.5	20.00	295.5	19.2	0.5	10	7
G11	22.3	24	67.00	59.2	71.9	0.2	11	7
G12	27.2	24.5	33.00	166.5	37.5	0.3	12	7
H03	101.6	23	42.00	287.5	-58.8	0.1	3	8
H04	80.8	20.5	13.00	141.3	20.6	0.1	4	8
H05	68.2	21	3.80	179.6	1.3	0.2	5	8
H06	77	24.5	37.00	342.6	-61.2	0.2	6	8
H07	250.9	19.5	45.00	150.5	-64.1	0.3	7	8
H08	19.3	24	20.00	284.7	9.7	0.1	8	8
H09	89.5	16	9.90	317.7	-72.2	0.1	9	8
H10	5.8	17.5	6.90	333.3	-0.1	0.1	10	8
H11	339.2	21	3.10	153.2	9.6	0.3	11	8
H12	19	19.5	450.00	98.2	0.3	0.4	12	8
I03	142.7	22.5	14.00	285.2	-84.8	0.2	3	9
I04	132.6	21	95.00	147.3	46.1	0.5	4	9
I05	138.9	24	7.00	151.6	52.2	0.2	5	9
I06	122.2	18	26.00	301.1	0.2	0.2	6	9
I07	101.5	19	65.00	102.8	-26.6	0.2	7	9
I08	92.8	19	7.00	226	-75.7	0.3	8	9

Continuation of Table B1

I09	90.2	20	15.00	310.2	-40.1	0.4	9	9
I10	133.4	22	32.00	305.9	59.9	0.2	10	9
I11	118.7	21	17.00	84.6	30.9	0.5	11	9
I12	359.5	27	34.00	124.7	-70	0.3	12	9
J03	116.3	17.5	0.74	278.8	-6.3	0.4	3	10
J04	110.5	17	15.00	135.5	20.2	0.2	4	10
J05	185.9	23.5	4.80	97.3	15.5	0	5	10
J06	292.3	21.5	4.70	301.9	-64.5	0.3	6	10
J07	346.6	19	9.10	156.3	-40.7	0.3	7	10
J08	345.3	23	29.00	44.9	-66.2	0.4	8	10
J09	74	15	45.00	342.8	-55.2	0.3	9	10
J10	42.3	16	140.00	74.4	36.3	0.1	10	10
J11	314.5	23.5	20.00	106.9	59.7	0.2	11	10
J12	341.4	19.5	2.80	147.5	-19	0.2	12	10
K03	120.5	23	9.60	243.1	-57.2	0.3	3	11
K04	122.8	18	36.00	125.4	66.4	0.1	4	11
K05	111.6	24	14.00	247	-68.6	0.2	5	11
K06	111.5	20.5	7.00	246.8	10.4	0.1	6	11
K07	28.8	17	3.20	257.4	35.8	0.4	7	11
K08	327.8	25.5	24.00	85.2	29.2	0.1	8	11
K09	39.6	17.5	92.00	87.7	-73.1	0.2	9	11
K10	73.4	18.5	47.00	332.8	-52.9	0.4	10	11
K11	341.3	20.5	110.00	334.5	42.7	0.5	11	11
K12	357.5	21.5	26.00	156.2	18.2	0.3	12	11
L03	172.8	22.5	54.00	351.5	50.8	0.2	3	12
L04	284.6	17.5	120.00	130	39.5	0.5	4	12

Continuation of Table B1

L05	298.9	20	13.00	152.4	-7.5	0.4	5	12
L06	339.3	18	6.50	182.9	-88.2	0.2	6	12
L07	66.6	17	3.40	117.3	12.9	0.4	7	12
L08	332.2	16.5	1.30	187.3	-24.6	0.1	8	12
L09	45.9	21	21.00	320.2	-70.4	0.3	9	12
L10	7.7	18.5	75.00	140.2	-57.6	0.4	10	12
L11	7.6	21.5	48.00	109.4	-24.7	0.2	11	12
L12	346.9	19	120.00	121.7	65.2	0.15	12	12

Table B2. Bulk susceptibility and anisotropy of magnetic susceptibility (AMS) values for the 100 samples. k_1, k_2 and k_3 are the maximum, intermediate and minimum anisotropy of magnetic susceptibility of the 100 samples, s.d is the standard deviation in the computation of the k1, k2 and k3 values.

The following is a list of the Anisotropy of magnetic susceptibility values of the 100 samples over the 10 m x 10 m grid.

Specimen	Susceptibility (SI)	k1	K2	k3	s.d	X (m)	Y (m)
C03	0.0047	1.0457	1.0061	0.9481	0.0010	3	3
C04	0.0046	1.0535	0.9926	0.9539	0.0005	4	3
C05	0.0025	1.0329	1.0029	0.9643	0.0006	5	3
C06	0.0044	1.0458	1.0154	0.9388	0.0007	6	3
C07	0.0002	1.0090	1.0060	0.9851	0.0006	7	3
C08	0.0007	1.0687	1.0313	0.9000	0.0010	8	3
C09	0.0111	1.0860	1.0519	0.8621	0.0006	9	3
C10	0.0191	1.1711	1.0673	0.7617	0.0039	10	3
C11	0.0311	1.0631	1.0443	0.8927	0.0008	11	3
C12	0.0001	1.0130	1.0016	0.9854	0.0004	12	3
D03	0.0079	1.1199	1.0416	0.8384	0.0007	3	4
D04	0.0046	1.0651	1.0267	0.9082	0.0006	4	4
D05	0.0016	1.0506	1.0041	0.9453	0.0001	5	4
D06	0.0033	1.0846	0.9997	0.9157	0.0008	6	4
D07	0.0030	1.0549	1.0359	0.9092	0.0001	7	4
D08	0.0058	1.0631	1.0483	0.8886	0.0006	8	4
D09	0.0336	1.1011	1.0581	0.8408	0.0006	9	4
D10	0.0238	1.1150	1.0384	0.8466	0.0010	10	4
D11	0.0294	1.0986	1.0757	0.8257	0.0005	11	4

Continuation of Table B2

D12	0.0132	1.0509	1.0403	0.9088	0.0009	12	4
E03	0.0032	1.0543	1.0209	0.9248	0.0000	3	5
E04	0.0008	1.0431	1.0020	0.9550	0.0001	4	5
E05	0.0033	1.0579	1.0282	0.9139	0.0007	5	5
E06	0.0018	1.0403	1.0263	0.9333	0.0002	6	5
E07	0.0347	1.0969	1.0541	0.8490	0.0014	7	5
E08	0.0021	1.0543	1.0361	0.9096	0.0005	8	5
E09	0.0025	1.0573	1.0113	0.9314	0.0008	9	5
E10	0.0044	1.0499	1.0309	0.9192	0.0015	10	5
E11	0.0048	1.0520	1.0178	0.9302	0.0007	11	5
E12	0.0087	1.0684	1.0265	0.9050	0.0005	12	5
F03	0.0005	1.0474	1.0110	0.9416	0.0004	3	6
F04	0.0070	1.0506	1.0410	0.9083	0.0001	4	6
F05	0.0026	1.0553	1.0286	0.9160	0.0004	5	6
F06	0.0012	1.0329	1.0172	0.9499	0.0009	6	6
F07	0.0005	1.0234	1.0052	0.9714	0.0005	7	6
F08	0.1873	1.1819	0.9723	0.8458	0.0003	8	6
F09	0.0150	1.1036	0.9994	0.8971	0.0007	9	6
F10	0.0014	1.0357	1.0197	0.9446	0.0006	10	6
F11	0.0039	1.0415	1.0216	0.9369	0.0001	11	6
F12	0.0022	1.0362	1.0104	0.9533	0.0003	12	6
G03	0.0014	1.0429	1.0222	0.9348	0.0003	3	7
G04	0.0029	1.0604	1.0141	0.9254	0.0002	4	7
G05	0.0017	1.0352	1.0114	0.9533	0.0002	5	7
G06	0.0010	1.0219	1.0162	0.9619	0.0010	6	7
G07	0.0026	1.0440	0.9954	0.9606	0.0001	7	7

Continuation of Table B2

G08	0.0100	1.0646	1.0574	0.8780	0.0002	8	7
G09	0.0072	1.0689	1.0310	0.9002	0.0011	9	7
G10	0.0077	1.1081	1.0277	0.8643	0.0020	10	7
G11	0.0470	1.0596	1.0172	0.9232	0.0012	11	7
G12	0.0186	1.0769	1.0202	0.9029	0.0009	12	7
H03	0.0160	1.0819	1.0425	0.8757	0.0006	3	8
H04	0.0054	1.0475	1.0297	0.9228	0.0006	4	8
H05	0.0053	1.0609	1.0353	0.9038	0.0013	5	8
H06	0.0167	1.1254	1.0382	0.8365	0.0006	6	8
H07	0.0117	1.1049	1.0466	0.8486	0.0001	7	8
H08	0.0134	1.1142	1.0233	0.8625	0.0002	8	8
H09	0.0020	1.0231	1.0131	0.9638	0.0006	9	8
H10	0.0025	1.0298	0.9898	0.9805	0.0002	10	8
H11	0.0013	1.0373	1.0079	0.9549	0.0001	11	8
H12	0.1867	1.1781	0.9617	0.8602	0.0003	12	8
I03	0.0112	1.0911	1.0508	0.8581	0.0005	3	9
I04	0.0262	1.0928	1.0315	0.8757	0.0003	4	9
I05	0.0058	1.0702	1.0228	0.9070	0.0004	5	9
I06	0.0042	1.0431	1.0273	0.9296	0.0013	6	9
I07	0.0395	1.0750	1.0390	0.8860	0.0014	7	9
I08	0.0154	1.0874	1.0385	0.8741	0.0002	8	9
I09	0.0034	1.0698	1.0131	0.9171	0.0002	9	9
I10	0.0033	1.0681	1.0258	0.9061	0.0018	10	9
I11	0.0064	1.0687	1.0380	0.8933	0.0007	11	9
I12	0.0080	1.0522	1.0273	0.9205	0.0014	12	9
J03	0.0009	1.0684	0.9993	0.9323	0.0002	3	10

Continuation of Table B2

J04	0.0051	1.1027	1.0214	0.8759	0.0002	4	10
J05	0.0006	1.0164	1.0041	0.9794	0.0001	5	10
J06	0.0007	1.0280	1.0151	0.9569	0.0001	6	10
J07	0.0013	1.0365	1.0014	0.9621	0.0005	7	10
J08	0.0565	1.1012	1.0435	0.8553	0.0008	8	10
J09	0.0384	1.1198	1.0544	0.8258	0.0006	9	10
J10	0.0327	1.1091	1.0566	0.8344	0.0001	10	10
J11	0.0027	1.0524	1.0179	0.9298	0.0000	11	10
J12	0.0013	1.0392	1.0207	0.9401	0.0003	12	10
K03	0.0106	1.0991	1.0344	0.8665	0.0002	3	11
K04	0.0043	1.0537	1.0103	0.9360	0.0000	4	11
K05	0.0030	1.0654	1.0049	0.9297	0.0009	5	11
K06	0.0030	1.0478	1.0065	0.9457	0.0001	6	11
K07	0.0004	1.0245	0.9998	0.9757	0.0005	7	11
K08	0.0239	1.0253	1.0127	0.9620	0.0003	8	11
K09	0.0842	1.1096	1.0692	0.8212	0.0001	9	11
K10	0.0282	1.0937	1.0413	0.8650	0.0008	10	11
K11	0.0271	1.0771	1.0318	0.8911	0.0017	11	11
K12	0.0038	1.0373	1.0248	0.9380	0.0004	12	11
L03	0.0242	1.0692	1.0368	0.8940	0.0027	3	12
L04	0.0260	1.0726	1.0274	0.9001	0.0019	4	12
L05	0.0137	1.0720	1.0239	0.9041	0.0001	5	12
L06	0.0027	1.0522	1.0090	0.9387	0.0012	6	12
L07	0.0025	1.0767	1.0202	0.9030	0.0001	7	12
L08	0.0029	1.0427	1.0089	0.9483	0.0001	8	12
L09	0.0152	1.1041	1.0570	0.8389	0.0001	9	12

Continuation of Table B2

L10	0.0361	1.0821	1.0447	0.8732	0.0011	10	12
L11	0.0111	1.1005	1.0332	0.8662	0.0005	11	12
L12	0.0366	1.0758	1.0689	0.8553	0.0001	12	12