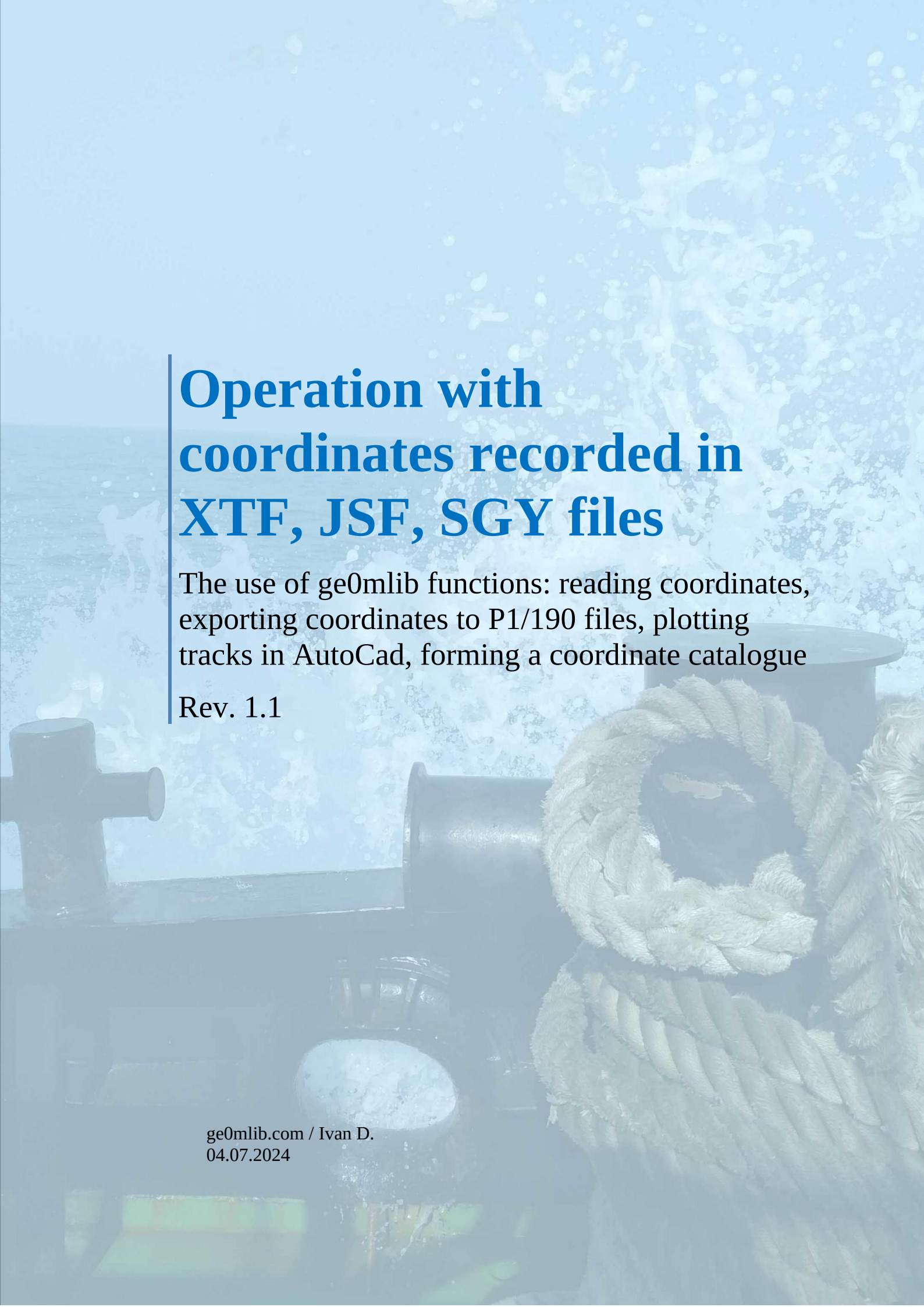


Operation with coordinates recorded in XTF, JSF, SGY files

The use of ge0mlib functions: reading coordinates, exporting coordinates to P1/190 files, plotting tracks in AutoCad, forming a coordinate catalogue

Rev. 1.1

ge0mlib.com / Ivan D.
04.07.2024

Contents

Revision History

Version	Date	Description of changes
1.0	2020-12-01	Document created // Ivan D.
1.1	2024-04-07	Document revised // Ivan D.

Figures list

Introduction

The paper discusses the use of ge0mlib library functions for operation with coordinates contained in Xtf, Jsf, Sgy files binary code. An example of code that uses the functions is given in the gNavP190_Script_Catalogue script. The code sequentially performs the following steps:

-- reading Xtf, Jsf, Sgy files from a given folder to obtain geophysical equipment coordinates written in the files code;
-- creating and saving P1/90 files containing equipment coordinates to HDD;
-- creating and saving a text file with coordinates to HDD (coordinate catalogue);
-- plotting tracks based on equipment coordinates in the MatLab graphics window;
-- creating files with tracks for drawing them in AutoCAD (AutoCAD scripts).

Xtf files used when running the gNavP190_Script_Catalogue script can be found at the following link: http://ge0mlib.com/g/example/ET4200_xtf.zip. To execute code that operates with Xtf files, one needs to run the script using the command

```
>> {'Xtf','c:\ET4200_xtf\'};gNavP190_Script_Catalogue.m;
```

specifying the path to the folder with Xtf files in the command code.

Jsf files used when running the gNavP190_Script_Catalogue script can be found at the following link: http://ge0mlib.com/g/example/ET4200_jsf.zip. To execute code that operates with Jsf files, one needs to run the script using the command

```
>> {'Jsf','c:\ET4200_jsf\'};gNavP190_Script_Catalogue.m;
```

specifying the path to the folder with Jsf files in the command code.

Sgy files used when running the gNavP190_Script_Catalogue script can be found at the following link: http://ge0mlib.com/g/example/ET3200SX512i_sgy.zip. To execute code that operates with Sgy files, one needs to run the script using the command

```
>> {'Sgy','c:\ET3200SX512i_sgy\'};gNavP190_Script_Catalogue.m;
```

specifying the path to the folder with the Sgy files in the command code.

The script requires the following ge0mlib library components:

-- gXtf for reading Xtf files;
-- gJsf for reading Jsf files;
-- gSgy for reading Sgy files;
-- gNav for converting coordinates between projections/coordinate systems;
-- gP190 for saving coordinates in a P1/90 file;
-- gAcad for writing coordinates to a script file for AutoCAD;
-- gMap for plotting tracks in the MatLab graphics window.

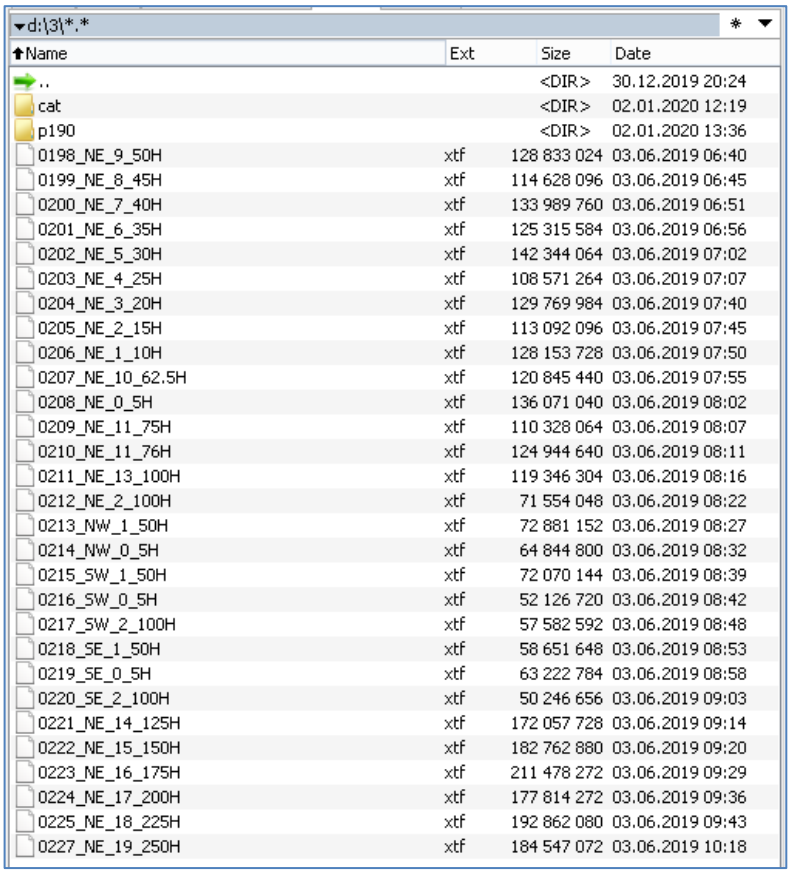
The gNavP190_Script_Catalogue script functions and code were tested in MatLab R2015b.

1. Operation with coordinates recorded in Xtf files

To run the code operating with Xtf files, one needs to place the Xtf files in a separate folder (an example of the folder contents is presented in ***Figure 1.1***). Then, one needs to run the script using a command such as

```
>> {'Xtf','d:\3\'};gNavP190_Script_Catalogue.m;
```

specifying the path to the folder with Xtf files in the command parameters.



d:\3*.*

Name	Ext	Size	Date
..		<DIR>	30.12.2019 20:24
cat		<DIR>	02.01.2020 12:19
p190		<DIR>	02.01.2020 13:36
0198_NE_9_50H	xtf	128 833 024	03.06.2019 06:40
0199_NE_8_45H	xtf	114 628 096	03.06.2019 06:45
0200_NE_7_40H	xtf	133 989 760	03.06.2019 06:51
0201_NE_6_35H	xtf	125 315 584	03.06.2019 06:56
0202_NE_5_30H	xtf	142 344 064	03.06.2019 07:02
0203_NE_4_25H	xtf	108 571 264	03.06.2019 07:07
0204_NE_3_20H	xtf	129 769 984	03.06.2019 07:40
0205_NE_2_15H	xtf	113 092 096	03.06.2019 07:45
0206_NE_1_10H	xtf	128 153 728	03.06.2019 07:50
0207_NE_10_62.5H	xtf	120 845 440	03.06.2019 07:55
0208_NE_0_5H	xtf	136 071 040	03.06.2019 08:02
0209_NE_11_75H	xtf	110 328 064	03.06.2019 08:07
0210_NE_11_76H	xtf	124 944 640	03.06.2019 08:11
0211_NE_13_100H	xtf	119 346 304	03.06.2019 08:16
0212_NE_2_100H	xtf	71 554 048	03.06.2019 08:22
0213_NW_1_50H	xtf	72 881 152	03.06.2019 08:27
0214_NW_0_5H	xtf	64 844 800	03.06.2019 08:32
0215_SW_1_50H	xtf	72 070 144	03.06.2019 08:39
0216_SW_0_5H	xtf	52 126 720	03.06.2019 08:42
0217_SW_2_100H	xtf	57 582 592	03.06.2019 08:48
0218_SE_1_50H	xtf	58 651 648	03.06.2019 08:53
0219_SE_0_5H	xtf	63 222 784	03.06.2019 08:58
0220_SE_2_100H	xtf	50 246 656	03.06.2019 09:03
0221_NE_14_125H	xtf	172 057 728	03.06.2019 09:14
0222_NE_15_150H	xtf	182 762 880	03.06.2019 09:20
0223_NE_16_175H	xtf	211 478 272	03.06.2019 09:29
0224_NE_17_200H	xtf	177 814 272	03.06.2019 09:36
0225_NE_18_225H	xtf	192 862 080	03.06.2019 09:43
0227_NE_19_250H	xtf	184 547 072	03.06.2019 10:18

Figure 1.1 Contents of the folder with Xtf files

Below are the main steps that the script performs. The given commands are taken from the script body.

1) A navigation datum is set to convert coordinates from the projection in which the geophysical equipment coordinates are recorded (in the Xtf file) into the DTEN fields projection (additional fields created by MatLab after reading the Xtf file).

```
>> NavS=struct('TargCode',2);
>> NavP=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode' ,6);
```

NavS (description of the "sensor" datum in which the geophysical equipment coordinates are recorded) includes the code "2" (***Figure 1.2***), indicating that Xtf files contain geographic coordinates. NavP (description of the "output" files datum) includes the code "6" (***Figure 1.2***), indicating that the

"output" files should contain coordinates in a rectangular projection. Additionally, NavP includes parameters to convert geographic coordinates "2" into a rectangular projection "6".

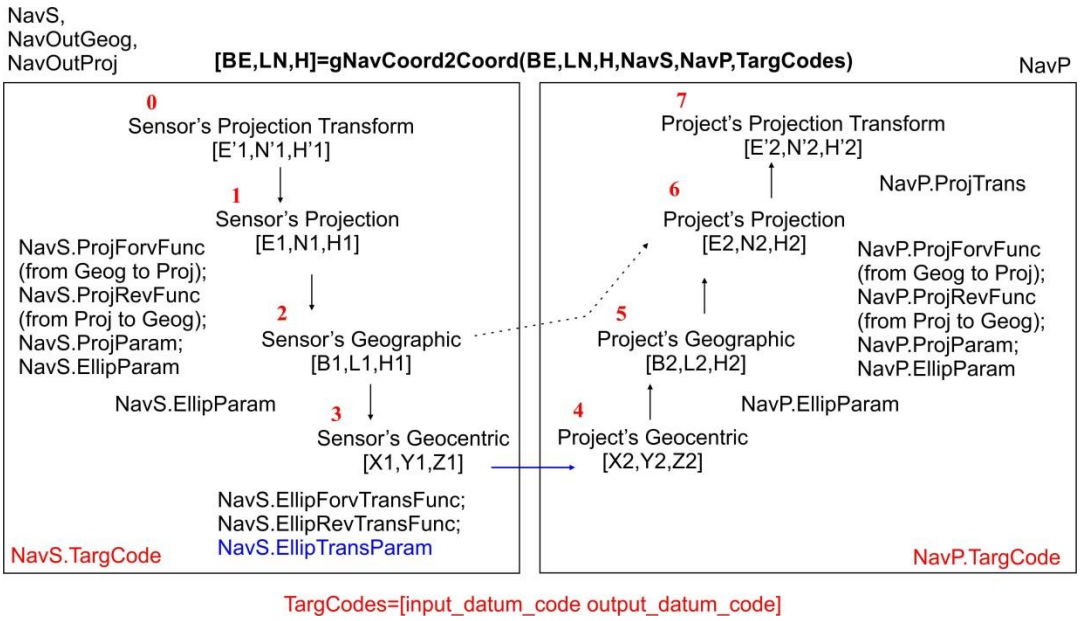


Figure 1.2 Diagram of coordinate conversion from NavS to NavP (red numbers are the values of the Nav.TargCode field)

2) The command is executed to read the contents of all Xtf files from the specified folder (***Figure 1.3***).

```
>> [XtfHead,Head,Data]=gXtf000DatasetImport('d:\3\','',[],0,[],[],'HPingNumber','HShipYcoordinate',
'HShipXcoordinate', NavS,NavP, []);
```

Information from the Xtf files' headers (from the d:\3\ folder) is placed in two variables of the "structure" type – XtfHead and Head (***Figure 1.3***). In this case, the headers' contents of one Xtf file are written to each element of the structure vector. When reading, DTEN fields containing coordinates converted from NavS to NavP (from "2" to "6") are created.

```
>> [XtfHead,Head,Data]=gXtf000DatasetImport('d:\3\','',[],0,[],[],'HPingNumber','HShipYcoordinate',
d:\3\0198_NE_9_50H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 5500 [ SubCh: 0, Num: 5500; ChFollow: 2, Num: 5500 ]
d:\3\0199_NE_8_45H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 6092 [ SubCh: 0, Num: 6092; ChFollow: 2, Num: 6092 ]
d:\3\0200_NE_7_40H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 7121 [ SubCh: 0, Num: 7121; ChFollow: 2, Num: 7121 ]
d:\3\0201_NE_6_35H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 6660 [ SubCh: 0, Num: 6660; ChFollow: 2, Num: 6660 ]
d:\3\0202_NE_5_30H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 7565 [ SubCh: 0, Num: 7565; ChFollow: 2, Num: 7565 ]
d:\3\0203_NE_4_25H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 4635 [ SubCh: 0, Num: 4635; ChFollow: 2, Num: 4635 ]
d:\3\0204_NE_3_20H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 5540 [ SubCh: 0, Num: 5540; ChFollow: 2, Num: 5540 ]
d:\3\0205_NE_2_15H.xtf
Rec: 0=XTF_HEADER_SONAR (Sidescan data), Num: 4828 [ SubCh: 0, Num: 4828; ChFollow: 2, Num: 4828 ]
```

Figure 1.3 Screenshot of the MatLab window, which shows the messages displayed when reading Xtf files (the messages show the number of SSS pings, channel and subchannel numbers)

In ***Table 1.1*** there are fields' names of the Head structure vector first element. The structure contains information from the 0198_NE_9_50H.xtf file header. DTEN fields are marked with a cyan color.

Table 1.1 The Head(1) structure fields containing information from the XTF file header

HMessageType: 0	HShipGyro: [1x5500 double]	HFishPositionErrorCode:
HSubChannelNumber: 0	HShipYcoordinate: [1x5500 double]	HOptionalOffsey: [1x5500 double]
HMessageNum: [1x5500 double]	HShipXcoordinate: [1x5500 double]	HCableOutHundredths: [1x5500
HYear: [1x5500 double]	HShipAltitude: [1x5500 double]	HReservedSpace2: [6x5500 double]
HMonth: [1x5500 double]	HShipDepth: [1x5500 double]	CChannelNumber: [2x5500 double]
HDay: [1x5500 double]	HFixTimeHour: [1x5500 double]	CDownsampleMethod: [2x5500
HHour: [1x5500 double]	HFixTimeMinute: [1x5500 double]	CSlantRange: [2x5500 double]
HMinute: [1x5500 double]	HFixTimeSecond: [1x5500 double]	CGroundRange: [2x5500 double]
HSecond: [1x5500 double]	HFixTimeHsecond: [1x5500	CTimeDelay: [2x5500 double]
HHSeconds: [1x5500 double]	HSensorSpeed: [1x5500 double]	CTimeDuration: [2x5500 double]
HJulianDay: [1x5500 double]	HKP: [1x5500 double]	CSecondsPerPing: [2x5500 double]
HEventNumber: [1x5500 double]	HSensorYcoordinate: [1x5500	CProcessingFlags: [2x5500 double]
HPingNumber: [1x5500 double]	HSensorXcoordinate: [1x5500	CFrequency: [2x5500 double]
HSoundVelocity: [1x5500 double]	HSonarStatus: [1x5500 double]	CInitialGainCode: [2x5500 double]
HOceanTide: [1x5500 double]	HRangeToFish: [1x5500 double]	CGainCode: [2x5500 double]
HReserved2: [1x5500 double]	HBearingToFish: [1x5500 double]	CBandWidth: [2x5500 double]
HConductivityFreq: [1x5500 double	HCableOut: [1x5500 double]	CContactNumber: [2x5500 double]
HTemperatureFreq: [1x5500 double]	HLayback: [1x5500 double]	CContactClassification: [2x5500
HPressureFreq: [1x5500 double]	HCableTension: [1x5500 double]	CContactSubNumber: [2x5500
HPressureTemp: [1x5500 double]	HSensorDepth: [1x5500 double]	CContactType: [2x5500 double]
HConductivity: [1x5500 double]	HSensorPrimaryAltitude: [1x5500	CNumSamples: [2x5500 double]
HWaterTemperature: [1x5500 doubl	HSensorAuxAltitude: [1x5500	CMillivoltScale: [2x5500 double]
HPressure: [1x5500 double]	HSensorPitch: [1x5500 double]	CContactTimeOffTrack: [2x5500
HComputedSoundVelocity: [1x5500	HSensorRoll: [1x5500 double]	CContactCloseNumber: [2x5500
HMagX: [1x5500 double]	HSensorHeading: [1x5500 double]	CReserved2: [2x5500 double]
HMagY: [1x5500 double]	HHeave: [1x5500 double]	CFixedVSOP: [2x5500 double]
HMagZ: [1x5500 double]	HYaw: [1x5500 double]	CWeight: [2x5500 double]
HAuxVal1: [1x5500 double]	HAttitudeTimeTag: [1x5500 double]	CReservedSpace: [2x5500x4
HAuxVal2: [1x5500 double]	HDOT: [1x5500 double]	GPSKP: [1x5500 double]
HAuxVal3: [1x5500 double]	HNavFixMilliseconds: [1x5500	GpsDay: [1x5500 double]
HAuxVal4: [1x5500 double]	HComputerClockHour: [1x5500	GpsTime: [1x5500 double]
HAuxVal5: [1x5500 double]	HComputerClockMinute: [1x5500	CompTime: [1x5500 double]
HAuxVal6: [1x5500 double]	HComputerClockSecond: [1x5500	GpsE: [1x5500 double]
HSpeedLog: [1x5500 double]	HComputerClockHsec: [1x5500	GPSN: [1x5500 double]
HTurbidity: [1x5500 double]	HFishPositionDeltaX: [1x5500	GpsH: [1x5500 double]
HShipSpeed: [1x5500 double]	HFishPositionDeltaY: [1x5500	

3) A navigation datum is set to convert coordinates from the projection in which they are recorded in DTEN fields into the projection required for P1/90 files.

```
>> NavOutGeog=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','Targ Code',2);
>> NavOutProj=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargC ode',1);
```

4) The Head structure variable DTEN fields (the GpsKP, GpsDay, GpsTime, GpsE, GpsN, GpsH fields containing coordinates and time) are converted into the fields of the P190Head and PHead structure variables (these structures include data for export to P1/90 files).

```
>> [P190Head,PHead]=gP190DTEN2P190(XtfHead,Head,NavP,NavOutGeog,NavOutProj,'
d:\3\p190\P190_Header.txt','S',(' ')','1','1' ,' ',' ');
```

The P1/90 files header is taken from the d:\3\p190\P190_Header.txt file (***Table 1.2***); as an example, we use a dummy header from UHR survey data.

The MatLab window screenshot with the PHead variable contents is shown in ***Figure1.4***.

Table 1.2 P1/90 files' header

```
H0100 Survey area ###, ############
H0102 Vessel details ############
H0103 Source details Sleeve Gun 4x40cu inch
H0104 Streamer details GeoEel (solid, 192 channels)
H0200 Survey date June 2019
H0201 Tape date ## June 2019
H0202 Tape version UKOOA P1/90
H0300 Client ENL
H0400 Geophysical contractor ######
H0500 Positioning contractor ######
H0700 Positioning system Veripos LD3S
H0800 Shotpoint position Center of Cluster
H1000 Clock time GMT 0.0 hours
H1400 Geodetic Datum: Survey WGS84 WGS 1984 6378137.000 298.2572235630
H1700 Vertical datum Current sea surface by Echo Sounder
H1800 Projection UTM Zone 54N
H2000 Grid unit 1 International Meter 1.00
H2001 Height unit 1 International Meter 1.00
H2002 Angular unit 1 Degrees
H2200 Central Meridian 141 00 00.000 W
H2301 Grid Origin 0 00 00.000 N
H2302 Grid coordinates at Origin500000.00 E 0.00 N
H2401 Scale factor 0.9996
H2600 Water depth Current sea surface by Echo Sounder
```

Fields	RecordId	Spare1	VesselId	SourceId	OtherId	Spare2	PointNum	WaterDepth	GpsDay	GpsTime	LineName	GpsLat	GpsLon	GpsE	GpsN
1	'S'		'1'	'1'	''	''	1x5500 double	1x5500 double	1x5500 dou...	1x5500 dou...	0198_NE_9_5...	1x5500 dou...	1x5500 dou...	1x5500 dou...	1x5500 dou...
2	'S'		'1'	'1'	''	''	1x6092 double	1x6092 double	1x6092 dou...	1x6092 dou...	0199_NE_8_4...	1x6092 dou...	1x6092 dou...	1x6092 dou...	1x6092 dou...
3	'S'		'1'	'1'	''	''	1x7121 double	1x7121 double	1x7121 dou...	1x7121 dou...	0200_NE_7_4...	1x7121 dou...	1x7121 dou...	1x7121 dou...	1x7121 dou...
4	'S'		'1'	'1'	''	''	1x6660 double	1x6660 double	1x6660 dou...	1x6660 dou...	0201_NE_6_3...	1x6660 dou...	1x6660 dou...	1x6660 dou...	1x6660 dou...
5	'S'		'1'	'1'	''	''	1x7565 double	1x7565 double	1x7565 dou...	1x7565 dou...	0202_NE_5_3...	1x7565 dou...	1x7565 dou...	1x7565 dou...	1x7565 dou...
6	'S'		'1'	'1'	''	''	1x4635 double	1x4635 double	1x4635 dou...	1x4635 dou...	0203_NE_4_2...	1x4635 dou...	1x4635 dou...	1x4635 dou...	1x4635 dou...
7	'S'		'1'	'1'	''	''	1x5540 double	1x5540 double	1x5540 dou...	1x5540 dou...	0204_NE_3_2...	1x5540 dou...	1x5540 dou...	1x5540 dou...	1x5540 dou...
8	'S'		'1'	'1'	''	''	1x4828 double	1x4828 double	1x4828 dou...	1x4828 dou...	0205_NE_2_1...	1x4828 dou...	1x4828 dou...	1x4828 dou...	1x4828 dou...
9	'S'		'1'	'1'	''	''	1x5471 double	1x5471 double	1x5471 dou...	1x5471 dou...	0206_NE_1_1...	1x5471 dou...	1x5471 dou...	1x5471 dou...	1x5471 dou...
10	'S'		'1'	'1'	''	''	1x5159 double	1x5159 double	1x5159 dou...	1x5159 dou...	0207_NE_10_...	1x5159 dou...	1x5159 dou...	1x5159 dou...	1x5159 dou...
11	'S'		'1'	'1'	''	''	1x5809 double	1x5809 double	1x5809 dou...	1x5809 dou...	0208_NE_0_5H	1x5809 dou...	1x5809 dou...	1x5809 dou...	1x5809 dou...
12	'S'		'1'	'1'	''	''	1x4710 double	1x4710 double	1x4710 dou...	1x4710 dou...	0209_NE_11_...	1x4710 dou...	1x4710 dou...	1x4710 dou...	1x4710 dou...

Figure1.4 The MatLab window screenshot with the PHead variable contents

5) The contents of the P190Head and PHead variables are written to P1/90 files.

```
>> gP190Write('d:\3\p190\',P190Head,PHead,1,'NNMMM');
```

The contents of the P1/90 file folder are shown in ***Figure 1.5***. The contents of P1/90 file are shown in ***Figure1.6*** (water depth is set to zero since there is no WaterDepth field).

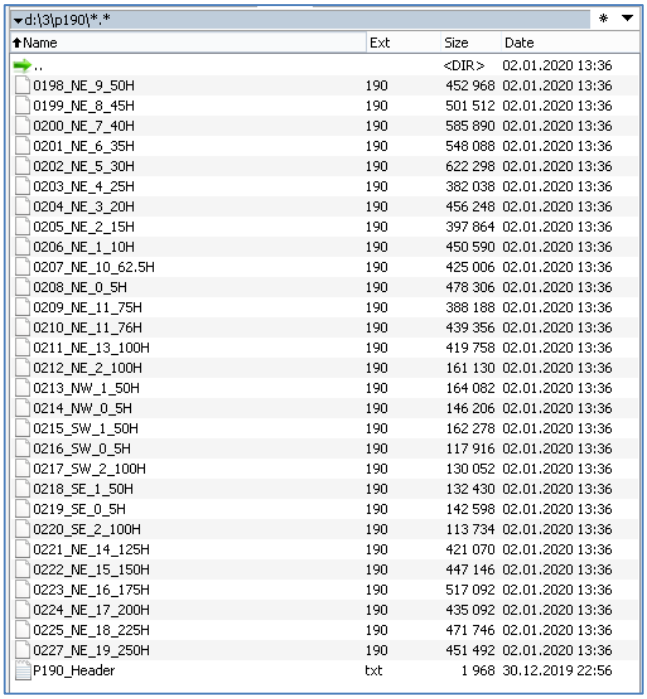


Figure 1.5 Contents of the P1/90 file folder

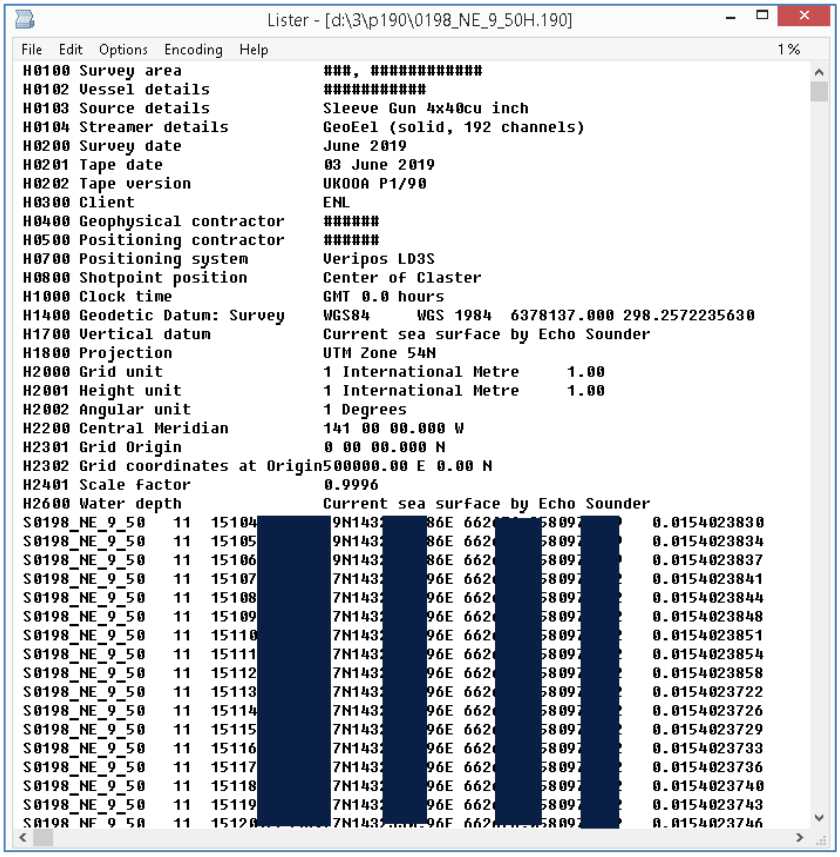


Figure1.6 The P1/90 file contents

6) The contents of the P190Head and PHead variables (data for P1/90) are written to a text file in the “coordinate catalogue” format (***Figure 1.7***):

```
>> gP1902Cat(P190Head,PHead,'d:\3\cat\zzz.txt','Ex2019y',10);
```

The names and contents of the text file columns as well as an optional header ('Ex2019y' in ***Figure 1.7***) are implemented in the gP1902Cat function body and “selected for execution” using the 'Ex2019y' function parameter. The “10” function parameter specifies that every tenth coordinate value is output to the catalogue file.

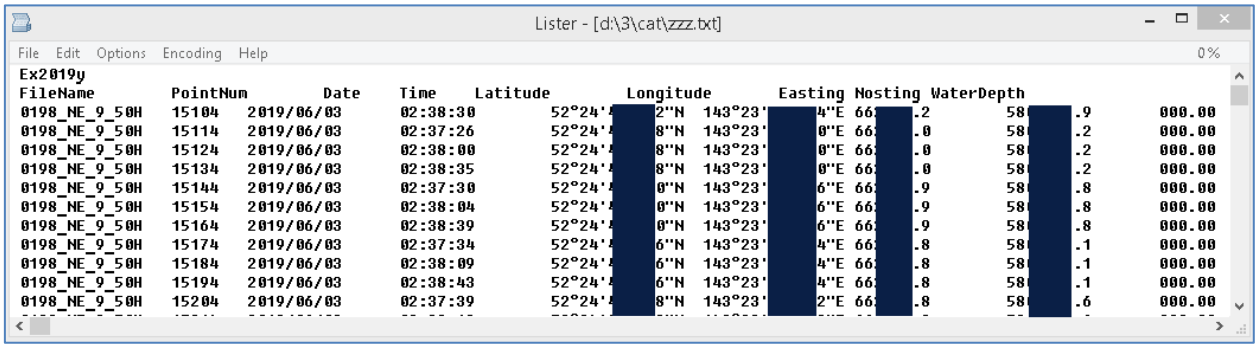


Figure 1.7 The “coordinate catalogue” file contents

7) The PHead structure variable is converted into the PL structure containing data for plotting “geophysical equipment” tracks in the MatLab graphics window. Then tracks are plotted (***Figure 1.8***).

```
>>PL=gP1902PL(PHead,'.-b');
>> gMapPLDraw(100,PL);axis equal; gMapTickLabel(100,'%1.1e',8);
```

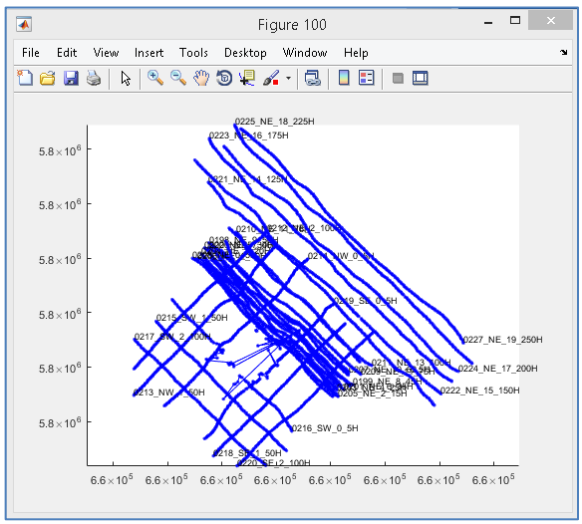


Figure 1.8 Geophysical equipment tracks plotted in the MatLab graphics window

8) The PL variable contents are written to a file containing the AutoCad script.

```
>> gMapPL2AcadExport('d:\3\cat\SSS_.scr',PL,[7 0 0 0 3],[1 1000],[6 0 5000 0],2,1,1);
```

The gMapPL2AcadExport function parameters specify the text display format, the point markers type, etc. A new layer is created for each survey line in AutoCad (***Figure 1.9***).

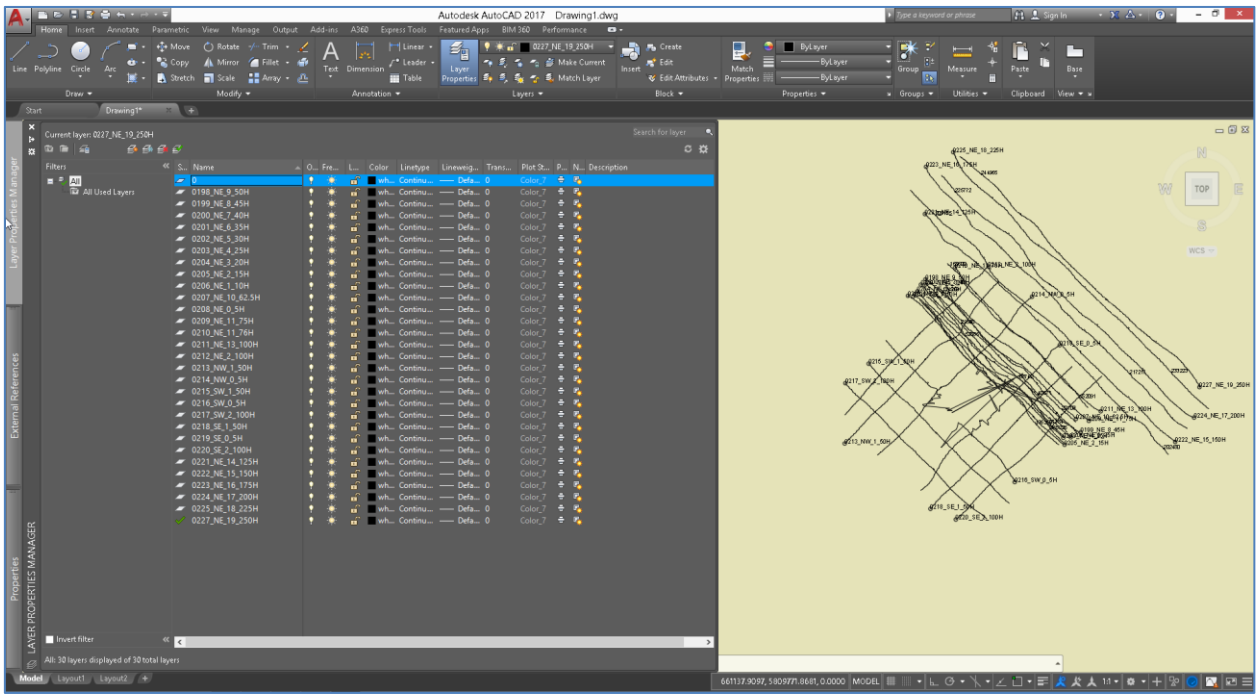


Figure 1.9 The AutoCAD window screenshot after executing AutoCAD script file commands

2. Operation with coordinates recorded in JSF files

To run the code operating with Jsf files, one needs to place the Jsf files in a separate folder (an example of the folder contents is presented in ***Figure 2.1***). Then, one needs to run the script using a command such as

```
>> {'Jsf','d:\8\ET4200\'};gNavP190_Script_Catalogue.m;
```

specifying the path to the folder with Jsf files in the command parameters.

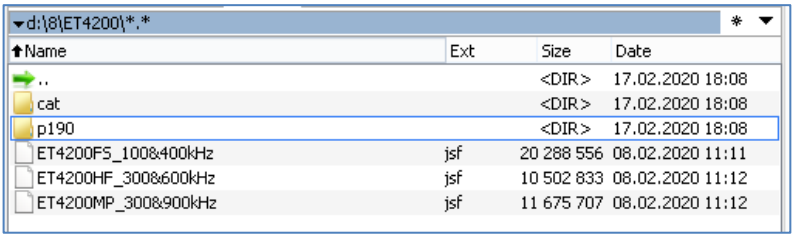


Figure 2.1 Contents of the folder with JSF files

Below are the main steps that the script performs. The given commands are taken from the script body.

1) A navigation datum is set to convert coordinates from the projection in which the geophysical equipment coordinates are recorded (in the Jsf file) into the DTEN fields projection (additional fields created by MatLab after reading the Jsf file).

```
>> NavS=struct('TargCode',2);
>> NavP=struct('EllipParam',[6378137 0.0818191908426215],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','T argCode',6);
```

2) The command is executed to read the contents of all JSF files from the specified folder. When reading, DTEN fields containing coordinates converted from NavS to NavP (from “2” to “6”) are created.

```
>> [JsfHead,Head,Data]=gJsf0080DatasetImport('d:\8\ET4200\','',[],0,20,[],[],'PingNumber',NavS,
NavP,[]);
```

3) A navigation datum is set to convert coordinates from the projection in which they are recorded in DTEN fields into the projection required for P1/90 files.

```
>> NavOutGeog=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','Targ Code',2);
>> NavOutProj=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargC ode',1);
```

4) The Head structure variable DTEN fields (the GpsKP, GpsDay, GpsTime, GpsE, GpsN, GpsH fields containing coordinates and time) are converted into the fields of the P190Head and PHead structure

variables (these structures include data for export to P1/90 files). The MatLab window screenshot with the PHead variable contents is shown in ***Figure 2.2***.

```
>> [P190Head,PHead]=gP190DTEN2P190(JsfHead,Head,NavP,NavOutGeog,NavOutProj,
'd:\8\ET4200\p190\P190_Header.txt','S',(' ')','1',' 1',' ',' ');
```

Fields	RecordId	Spare1	VesselId	SourceId	OtherId	Spare2	PointNum	WaterDepth	GpsDay	GpsTime	LineName	GpsLat	GpsLon	GpsE	GpsN
1	'S'		'1'	'1'	' '	' '	1x342 double	1x342 double	1x342 double	1x342 double	ET4200FS_100...	1x342 double	1x342 double	1x342 double	1x342 double
2	'S'		'1'	'1'	' '	' '	1x192 double	1x192 double	1x192 double	1x192 double	ET4200HF_30...	1x192 double	1x192 double	1x192 double	1x192 double
3	'S'		'1'	'1'	' '	' '	1x164 double	1x164 double	1x164 double	1x164 double	ET4200MP_30...	1x164 double	1x164 double	1x164 double	1x164 double

Figure 2.2 The MatLab window screenshot with the PHead variable contents

5) The contents of the P190Head and PHead variables are written to P1/90 files.

```
>> gP190Write('d:\8\ET4200\p190\',P190Head,PHead,1,'NNMMM');
```

6) The contents of the P190Head and PHead variables (data for P1/90) are written to a text file in the “coordinate catalogue” format.

```
>> gP1902Cat(P190Head,PHead,'d:\8\ET4200\cat\zzz.txt','Ex2019y',10);
```

7) The PHead structure variable is converted into the PL structure containing data for plotting “geophysical equipment” tracks in the MatLab graphics window.

```
>>PL=gP1902PL(PHead,'.-b');
>> gMapPLDraw(100,PL);axis equal; gMapTickLabel(100,'%1.1e',8);
```

8) The contents of the PL variable are written to a file containing the AutoCad script.

```
>> gMapPL2AcadExport('d:\8\ET4200\cat\SBP_.scr',PL,[7 0 0 0 3],[1 1000],[6 0 5000 0],2,1,1);
```

3. Operation with coordinates recorded in Sgy files

To run the code operating with Sgy files, one needs to place the Sgy files in a separate folder (an example of the folder contents is presented in ***Figure 3.1***). Then, one needs to run the script using a command such as

```
>> {'Sgy','d:\5\'};gNavP190_Script_Catalogue.m;
```

specifying the path to the folder with Sgy files in the command parameters.

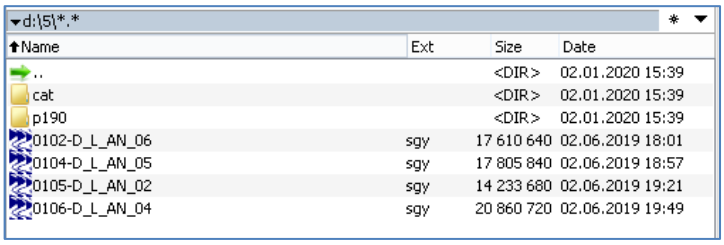


Figure 3.1 Contents of the folder with Sgy files

Below are the main steps that the script performs. The given commands are taken from the script body.

1) Setting a navigation datum is not required to convert coordinates from the projection in which the geophysical equipment coordinates are recorded (in the Sgy file) into the projection of DTEN fields (additional fields created by MatLab after reading the Jsf file). In the sgy files given as an example, the coordinates are already written in a rectangular projection, and they are simply copied into the DTEN fields.

2) The command is executed to read the contents of all Sgy files from the specified folder. When reading, DTEN fields containing coordinates in the NavP datum are created.

```
>>    [SgyHead,Head,Data]=gSgyDatasetImport('d:\5\','',[],[],[],'TraceSequenceFile','GroupX','GroupY',[],
[], []);
```

3) A navigation datum is set to convert coordinates from the projection in which they are recorded in DTEN fields into the projection required for P1/90 files.

```
>> NavP=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode' ,6);
>> NavOutGeog=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','Targ Code',2);
>> NavOutProj=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000 0],
'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargC ode',1);
```

4) The Head structure variable DTEN fields (the GpsKP, GpsDay, GpsTime, GpsE, GpsN, GpsH fields containing coordinates and time) are converted into the fields of the P190Head and PHead structure

variables (these structures include data for export to P1/90 files). The MatLab window screenshot with the PHead variable contents is shown in ***Figure 3.2***.

```
>> [P190Head,PHead]=gP190DTEN2P190(SgyHead,Head,NavP,NavOutGeog,NavOutProj,'
d:\5\p190\P190_Header.txt','S',(' ')','1','1' ,' ',' ');
```

Fields	RecordId	Spare1	VesselId	SourceId	OtherId	Spare2	PointNum	WaterDepth	GpsDay	GpsTime	LineName	GpsLat	GpsLon	GpsE	GpsN
1	'S'		'1'	'1'	''	''	1x1804 double	1x1804 double	1x1804 dou...	1x1804 dou...	0102-D_L_AN...	1x1804 dou...	1x1804 dou...	1x1804 dou...	1x1804 dou...
2	'S'		'1'	'1'	''	''	1x1824 double	1x1824 double	1x1824 dou...	1x1824 dou...	0104-D_L_AN...	1x1824 dou...	1x1824 dou...	1x1824 dou...	1x1824 dou...
3	'S'		'1'	'1'	''	''	1x1458 double	1x1458 double	1x1458 dou...	1x1458 dou...	0105-D_L_AN...	1x1458 dou...	1x1458 dou...	1x1458 dou...	1x1458 dou...
4	'S'		'1'	'1'	''	''	1x2137 double	1x2137 double	1x2137 dou...	1x2137 dou...	0106-D_L_AN...	1x2137 dou...	1x2137 dou...	1x2137 dou...	1x2137 dou...

Figure 3.2 The MatLab window screenshot with the PHead variable contents

5) The contents of the P190Head and PHead variables are written to P1/90 files.

```
>> gP190Write('d:\5\p190\',P190Head,PHead,1,'NNMMM');
```

6) The contents of the P190Head and PHead (data for P1/90) are written to a text file in the "coordinate catalogue" format (***Figure 3.3***).

```
>> gP1902Cat(P190Head,PHead,'d:\3\cat\zzz.txt','Ex2019y',10);
```

Lister - [d:\5\cat\zzz.txt]

File Edit Options Encoding Help 6%

Ex2019y

FileName	PointNum	Date	Time	Latitude	Longitude	Easting	Nosting	WaterDepth
0102-D_L_AN_06	1	2019/06/02	14:56:52	52°27'[redacted]7"N	143°40'2[redacted]"E	68[redacted].0	581[redacted].4	000.00
0102-D_L_AN_06	11	2019/06/02	14:56:54	52°27'[redacted]6"N	143°40'2[redacted]"E	68[redacted].2	581[redacted].7	000.00
0102-D_L_AN_06	21	2019/06/02	14:56:55	52°27'[redacted]6"N	143°40'2[redacted]"E	68[redacted].2	581[redacted].7	000.00
0102-D_L_AN_06	31	2019/06/02	14:56:57	52°27'[redacted]0"N	143°40'2[redacted]"E	68[redacted].9	581[redacted].7	000.00
0102-D_L_AN_06	41	2019/06/02	14:56:58	52°27'[redacted]3"N	143°40'2[redacted]"E	68[redacted].9	581[redacted].6	000.00
0102-D_L_AN_06	51	2019/06/02	14:57:00	52°27'[redacted]9"N	143°40'2[redacted]"E	68[redacted].4	581[redacted].4	000.00
0102-D_L_AN_06	61	2019/06/02	14:57:01	52°27'[redacted]5"N	143°40'2[redacted]"E	68[redacted].2	581[redacted].1	000.00
0102-D_L_AN_06	71	2019/06/02	14:57:03	52°27'[redacted]4"N	143°40'2[redacted]"E	68[redacted].7	581[redacted].1	000.00
0102-D_L_AN_06	81	2019/06/02	14:57:05	52°27'[redacted]9"N	143°40'2[redacted]"E	68[redacted].4	581[redacted].6	000.00

Figure 3.3 The "coordinate catalogue" file contents

7) The PHead structure variable is converted into the PL structure containing data for plotting "geophysical equipment" tracks in the MatLab graphics window. Then, the tracks are plotted (***Figure 3.4***).

```
>>PL=gP1902PL(PHead,'.-b');
>> gMapPLDraw(100,PL);axis equal; gMapTickLabel(100,'%1.1e',8);
```

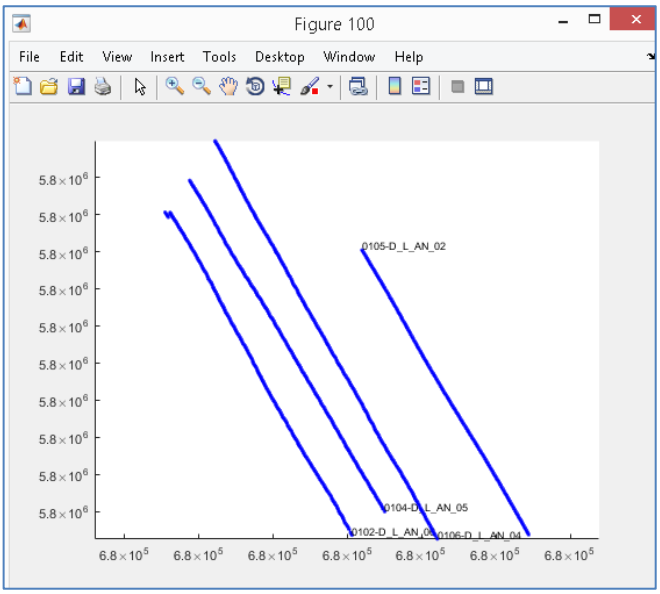


Figure 3.4 Geophysical equipment tracks plotted in the MatLab graphics window

8) The contents of the PL variable are written to a file containing the AutoCad script.

```
>> gMapPL2AcadExport('d:\5\cat\SBP_.scr',PL,[7 0 0 0 3],[1 1000],[6 0 5000 0],2,1,1);
```

The gMapPL2AcadExport function parameters specify the text display format, the point markers type, etc. A new layer is created for each survey line in AutoCad (*Figure 3.5*).

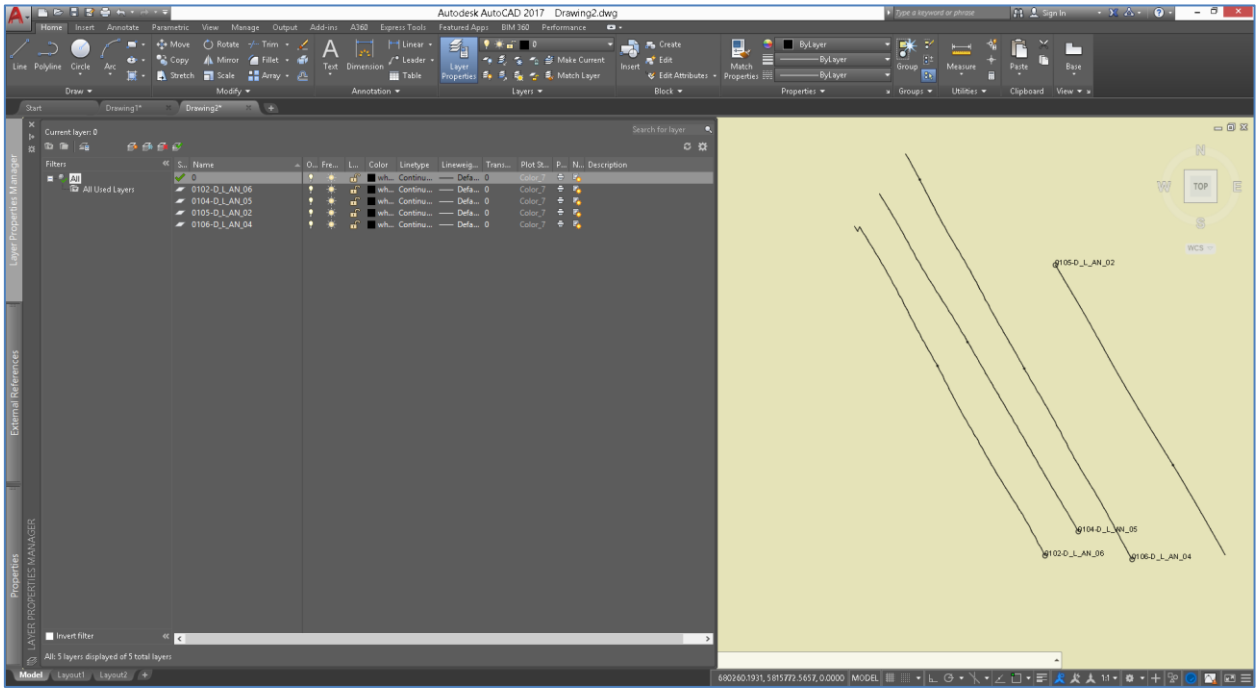


Figure 3.5 The AutoCAD window screenshot after executing AutoCAD script file commands

Appendix 1. gNavP190_Script_Catalogue script code

```
%script gNavP190_Script_Catalogue;
%Start script with command >>> {'Xtf','c:\ET4200_xtf\'}; gNavP190_Script_Catalogue; <<< or the same.
%Create P1/90, coordinate catalog and script with trackplots for AutoCAD using folder contained Xtf-files.
%The link to Xtf-files example: http://ge0mlib.com/g/example/ET4200_xtf.zip
%The link to Training detailed description: http://ge0mlib.com/g/gTraining01_Catalogue.pdf

gKey=ans;
ifstrcmp(gKey{1},'xtf'),%start processing for Xtf-files
 %Read root folder
 tryRootDir=gKey{2};catch,RootDir='c:\temp\ET4200_xtf\';end;
 %Read Xtf-files to Dataset
NavS=struct('TargCode',2);
NavP=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',6);
[XtfHead,Head,Data]=gXtf000DatasetImport(RootDir,'',[],0,[],[],'HPingNumber','HShipYcoordinate','HShipXcoordinate',NavS,NavP,[]);
 %Create P1/90 files
NavOutGeog=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',2);
NavOutProj=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',1);
[P190Head,PHead]=gNavP190DTEN2P190(XtfHead,Head,NavP,NavOutGeog,NavOutProj,[RootDir'ReadMe\P190_Header.txt'],'S',(' ')','1','1',' ','
');
gNavP190Write([RootDir'ReadMe\'],P190Head,PHead,1,'NNMMMM');
 %Create coordinates catalog with step 10 pings
gNavP1902Cat(P190Head,PHead,[RootDir'ReadMe\gTraining01_Survey_Catalogue.txt'],'gTraining01_Survey',10);
 %Create and draw polyline-structure
PL=gNavP1902PL(PHead,'.-b';gMapPLDraw(100,PL);axisequal; gMapTickLabel(100,'%8.1f',8);
 %Create AutoCAD script with trackplots
gMapPL2AcadExport([RootDir'ReadMe\gTraining01_Survey_Trackplot.scr'],PL,[7 0 0 0 3],[1 1000],[6 0 5000 0],2,1,1);
end;
ifstrcmp(gKey{1},'Jsf'),%start processing for Jsf-files
 %Read root folder
 tryRootDir=gKey{2};catch,RootDir='c:\temp\ET4200_jsf\';end;
 %Read Jsf-files to Dataset
NavS=struct('TargCode',2);
NavP=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',6);
[JsfHead,Head,Data]=gJsf0080DatasetImport(RootDir,'',[],0.20,[],[],'PingNumber',NavS,NavP,[]);
 %Create P1/90 files
NavOutGeog=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',2);
NavOutProj=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',1);
```

```
[P190Head,PHead]=gNavP190DTEN2P190(JsfHead,Head,NavP,NavOutGeog,NavOutProj,[RootDir'ReadMe\P190_Header.txt'],'S',(' ')','1','1',' ','
');
gNavP190Write([RootDir'ReadMe\'],P190Head,PHead,1,'NNMMMM');
 %Create coordinates catalog with step 10 pings
gNavP1902Cat(P190Head,PHead,[RootDir'ReadMe\gTraining01_Survey_Catalogue.txt'],'gTraining01_Survey',10);
 %Create and draw polyline-structure
PL=gNavP1902PL(PHead,'.-b';gMapPLDraw(101,PL);axisequal; gMapTickLabel(101,'%8.1f',8);
 %Create AutoCAD script with trackplots
gMapPL2AcadExport([RootDir'ReadMe\gTraining01_Survey_Trackplot.scr'],PL,[7 0 0 0 3],[1 1000],[6 0 5000 0],2,1,1);
end;
ifstrcmp(gKey{1},'Sgy'),%start processing for Sgy-files
 %Read root folder
 tryRootDir=gKey{2};catch,RootDir='c:\temp\ET3200SX512i_sgy\';end;
 %Read Sgy-files to Dataset
[SgyHead,Head,Data]=gSgyDatasetImport(RootDir,'',[],[],[],'TraceSequenceFile','GroupX','GroupY',[],[],[]);
NavP=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',6);
 %Create P1/90 files
NavOutGeog=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',2);
NavOutProj=struct('EllipParam',[6378137 0.0818191908],'ProjParam',[0 141 0.9996 500000
0],'ProjForvFunc','gNavGeog2ProjUtm','ProjRevFunc','gNavProjUtm2Geog','TargCode',1);
[P190Head,PHead]=gNavP190DTEN2P190(SgyHead,Head,NavP,NavOutGeog,NavOutProj,'d:\8\ET3200SX512i\ReadMe\P190_Header.txt','S',('
')','1','1',' ',' ');
gNavP190Write([RootDir'ReadMe\'],P190Head,PHead,1,'NNMMMM');
 %Create coordinates catalog with step 10 pings
gNavP1902Cat(P190Head,PHead,[RootDir'ReadMe\gTraining01_Survey_Catalogue.txt'],'gTraining01_Survey',10);
 %Create and draw polyline-structure
PL=gNavP1902PL(PHead,'.-b';gMapPLDraw(101,PL);axisequal; gMapTickLabel(101,'%8.1f',8);
 %Create AutoCAD script with trackplots
gMapPL2AcadExport([RootDir'ReadMe\gTraining01_Survey_Trackplot.scr'],PL,[7 0 0 0 3],[1 1000],[6 0 5000 0],2,1,1);
end;
% mail@ge0mlib.com 02/18/2020
```