

THE THREE-DIMENSIONAL GEODETIC NETWORKS ADJUSTEMENT AUTOMATION USING MATLAB

Ana-Maria LOGHIN –Ph.D Student, Eng., „Gheorghe Asachi” Technical University of Iasi, Faculty of Hydrotechnical Engineering, Geodesy and Environmental Engineering, loghin.anamaria@yahoo.com

Ajin.R.S – Independent Researcher, ajinares@gmail.com

Valeria Ersilia ONIGA –Lecturer, Ph.D Eng., „Gheorghe Asachi” Technical University of Iasi, Faculty of Hydrotechnical Engineering, Geodesy and Environmental Engineering, Department of Terrestrial Measurements and Cadastre, ersilia.oniga@tuiasi.ro

Abstract: A three-dimensional geodetic network is composed by a system of points located on earth's surface, on an engineering work territory, points which have a known position in an unitary reference coordinate system. Geodetic networks play a significant role in the attainable accuracy of surveying and mapping applications. This paper presents a three-dimensional geodetic network in form of a closed-loop traverse, which begins and ends at the same point station. In order to determine the traverse station points coordinates, the least squares method of indirect observations was used. The primary objective of this paper is writing a Matlab program with procedures and functions, which automatically calculates the unknown points coordinates, respectively their accuracy, thus streamlining the measurements processing.

Keywords: three-dimensional geodetic network, closed-loop traverse, least squares method, Matlab

1. Introduction

It is already known that no observation made to a thing or a phenomena will never be exact. Every observation contains errors. These statements are fundamental and accepted universally. Therefore, surveyors, who are measurement specialists, should have a thorough understanding of errors. They must be familiar with the different types of errors, their sources and their expected magnitudes and they must be able to adopt procedures for reducing error sizes when making their measurements and to analyze and adjust their data.

In recent years, the least square method of adjusting spatial data has been rapidly gaining popularity as the method used for analyzing and adjusting surveying data. This fact should not be surprising, because this method is the most rigorous adjustment procedure available [1].

Geodetic surveying is concerned with two different types of surveying: the first one for heights and the second one for horizontal elements. In recent years, with the technology evolution, there was developed and implemented the concept of three-dimensional geodetic network, that is a fusion between horizontal and vertical parts. Therefore, a three-dimensional geodetic network includes both types of measurements: for heights and also for the horizontal elements being composed by a system of points, located on the earth's surface, on an engineering work territory, points which have a known position in an unitary reference coordinate system.

With the development of the technology of Electronic Distance measurement, traversing has become an important method widely used in surveys. A traverse is represented by a series of straight lines called traverse legs, which are used to connect a set of selected points on the earth, called traverse stations. The surveyor makes length and direction measurements and uses them to compute the relative positions of the traverse stations in a coordinate system. Meantime, the surveyor must make the adjustment work of the indirect observations that compose the traverse and for this operation the least square method of indirect observations is used. During the entire process of the network adjustment, computing the approximate points coordinates is one of the important steps [2].

For the present study, the three-dimensional geodetic network is represented by a closed-loop traverse, which starts at a point and ends at the same point station. In order to minimize the effect of the accidental errors made in measurements, they are all adjusted by several computations. For the present situation, the type of closed-loop traverse is preferred, because it provides a check over the fieldwork, computations and starting data. Meantime, it provides a basis for comparison, in order to determine the overall accuracy of the entire work.

Since the main purpose of a traverse is to locate points relative to each other on a common grid, there are needed certain elements of starting data. For this study, the initial data is represented by the coordinates of the starting/ending point station.

The present work develops an adjustment application of the geodetic measurements, designed with the help of the Matlab programming language. It is presented the mathematical model of least squares method and all the phases in achieving the results of the proposed algorithm. For a data set consisting in three different types of observations, like: length observations, direction observations and angles, there were determined the most probable values for the points coordinates with the accuracy of their determination and the most probable values for the measurements, of course with their associated accuracy [3].

The current work presents the steps taken to create a program that automatically processes the measurements from a three-dimensional geodetic network, in which points are defined by all the three coordinates (3D Space).

2. Presentation of the mathematical algorithm

2.1. Starting data

For the present work the geodetic network is represented by a closed-loop traverse, located approximately in the center of Iasi city (Romania), surrounding the “Dosoftei House” Museum of Old Moldavian Literature (Fig. 1).

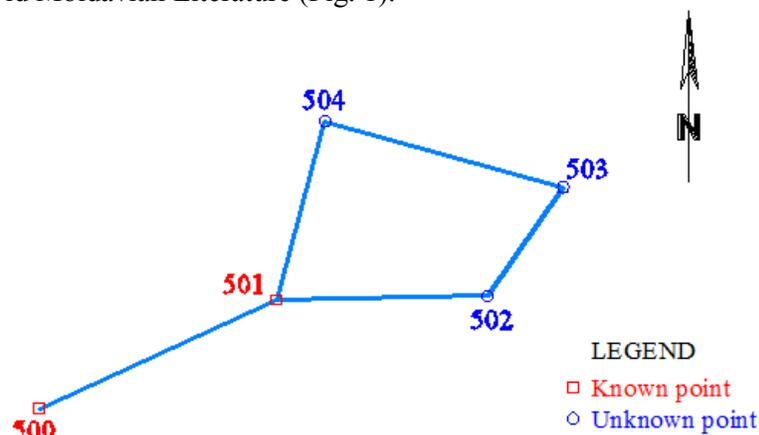


Fig. 1. The closed - loop traverse scheme

The traverse has two points with known coordinates (500 and 501) and also three unknown points (502, 503, 504) that must be computed. The known points coordinates resulted from a previous stage, using GNSS real time kinematic measurements, made with a South S82 – V receptor. During the traverse, the horizontal angles, vertical angles, and distances are measured, using a total station, with an accuracy of 7" for angles respectively of 2 mm + 2 ppm for distances.

2.2. Data processing

In order to create the adjustment application, the Matlab programming language was used. Matlab is a high-performance technical computing environment, developed by MathWorks, which combines both math and graphic functions with a powerful high-level language.

Some of the most important features of this language are: efficient matrix and vector computations, easy creation of scientific and engineering graphics, plotting of functions and data, algorithms implementation, application development including graphical user interface building, object-oriented programming and interfacing with programs written in other languages, including C, C++, Java, Fortran and Python.

Matlab can be used for a range of applications, including signal processing and communications, image and video processing, control systems, test and measurement, computational finance and computational biology. Because of its important feature of matrix and vector computation and of algorithm manipulation, it is primarily used in producing solutions to complex systems of equations, in modeling, simulation and prototyping and also in data analysis, exploration and visualization. More than a million engineering and scientists in industry and academia use Matlab, the language of technical computing [4].

2.3. Mathematical support

In order to compute the most probable values for points coordinates with the accuracy of their determination and also the corrections applied to the observed angles and measured distances, the least square method of indirect observations was used. This is the most rigorous adjustment procedure, based on the mathematical theory of probability that the sum of the squares of the errors times their respective weights are minimized. When v_i is assumed as a small correction value to be added to each observed value and p_i as a weight of observed values, each value of v_i is determined in a manner to satisfy the following formula [5]:

$$\sum_i p_i v_i^2 = \text{minimum} \quad (1)$$

In this program, one observation equation is developed for one set of observations and after theoretically evaluating the weight of observations, each equation is solved by the least squares method in order to satisfy equation (3) [6].

An important aspect of the least square method is that it permits to all kind of observations, regardless of their number or type, to be included into the adjustment process and simultaneously used in the computations. Thus, the adjustment combines distances, horizontal angles, azimuths, zenith or vertical angles, coordinates and also GNSS observations.

The entire adjustment process is composed by several steps, such as: computing the provisional values of the coordinates, preparing and solving the functional model of corrections, computing the adjusted elements and finally the accuracy analysis.

The functional-stochastic model for the indirect weighted measurement method, containing a number of r equations with n unknown parameters is represented by the following equation [5]:

$$B_{r-n} X_{n-1} + L_{r-1} = V_{r-1}, \text{ with } P_{r-r} \text{ weighting matrix} \quad (2)$$

where: B_{r-n} is the coefficient matrix of the correction equations,

X_{n-1} is the vector matrix of the unknown parameters (the corrections of the unknown point coordinates in a local geodetic system),

L_{r-1} is the vector-matrix of the free elements,

V_{r-1} is the correction matrix.

All the correction coefficients and also the free terms were calculated for the input data set composed by: distances, horizontal and vertical angle measurements.

The detailed algorithm of the three-dimensional geodetic network adjustment with its mathematical implementation can be entirely found in [7].

3. Results and discussion

3.1. Logical support

Before starting the implementation process of the adjustment in the Matlab programming language, the mathematical algorithm was carefully analyzed and also it was realized the flow chart of the program. A rough flow of the program is presented in Fig. 2.

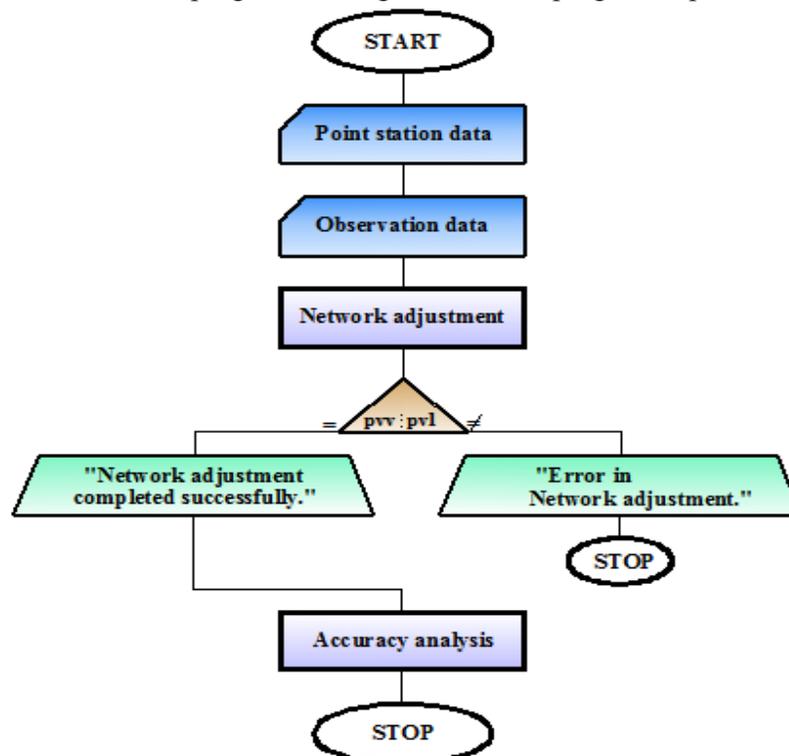


Fig. 2. Rough flow of the program

As input data, there was used two text files: one with the known point coordinates and the second one containing the measurements acquired with the help of a total station.

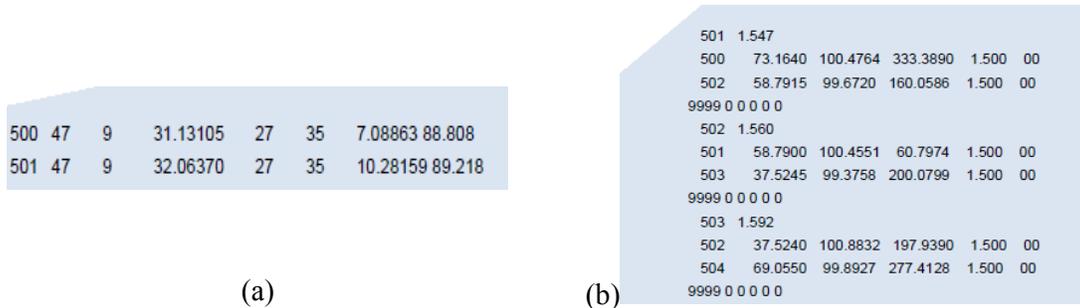


Fig. 3. Model of the input files, (a) point coordinates text file (*Points.txt*), (b) Measurements text file (*Traverse.txt*)

The present program retains all measured data from the text file in a cell-array *D*, having on its columns the observations in the following order: the name of the points, distances, vertical angles, horizontal directions, reflector elevation, code. In the total stations output text files, all observations are organized by point stations, being delimited by the “99999 0 0 0 0 0” sequence. Therefore, in order to store the data from each point station, the following code lines sequence was implemented:

```

fid = fopen("Traverse.txt","rt");
C = cell2mat(textscan(fid,'%f%f%f%f%f%f'));
fclose(fid);
idx = [1,diff(C(:,1))~=9999];
idy = find(idx<0)-find(idx>0);
idy(2,:) = 1;
D = mat2cell(C,idy(:),size(C,2));
D(2:2:end) = [];
    
```

Fig. 4. Extracting the observations from each point station

Next, there were defined the start, intermediate and the end point station points respectively, in order to determine the traverse direction. The dialog box allows the user to input the points’s names from the keyboard (Fig. 5).

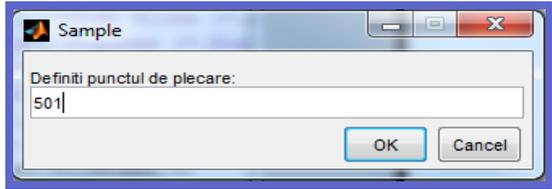


Fig. 5. Input dialog box

In the next steps, the application computes the provisional coordinates for all the unknown points of the traverse. Then, following the mathematical algorithm, the program realizes the adjustment process of the three-dimensional geodetic network. For each stage, there were created new script files, that were implemented as functions in the main program. Therefore, there were created several functions such as: the calculation of the horizontal angles function, the measurement errors calculation function, conversion function between geodetic and rectangular coordinates, azimuth angle function, coefficient calculation functions, accuracy analysis functions.

3.2. Conversion function between geodetic and rectangular coordinates

Based on the geodetic coordinates (B , L) and the constant coefficients a_{00} , a_{10} , a_{20} , a_{30} , ..., b_{01} , b_{11} , b_{21} , there were computed the plane rectangular coordinates in the National Projection System, called „Stereographical on unique secant plan” and the normal altitudes in the „Black Sea 1975”. The conversion algorithm is described in [8].

Firstly, using the particular function “dms_degrees” there were computed the geodetic coordinates from degrees, minutes, seconds into degrees with their decimal part, the results being stored in the ‘Points_BLH.txt’ text file.

```
function [ degreesB,degreesL ] = dms_degrees( dmsB,dmsL )
disp('Degrees, minutes seconds to degrees')
%input data
Coord =dlmread('Points.txt');
dmsB=Coord (:,2:4);
dmsL=Coord (:,5:7);
He=Coord (:,8);
%end of input data
format long g
degreesB = dms2degrees(dmsB);
degreesL = dms2degrees(dmsL);
coord_BLH(:,1)=Coord_cun(:,1);
coord_BLH(:,2)=degreesB;
coord_BLH(:,3)=degreesL;
coord_BLH(:,4)=Coord_cun(:,8);
dlmwrite('Points_BLH.txt',coord_BLH,'delimiter','\t','newline','pc','precision,9);
end
```

Fig. 6. The conversion function from degrees, minutes, seconds into degrees

The Matlab function uses the constant coefficients, stored in two different arrays par_a and par_b . Further, there are defined the geographical coordinates of the central point of Romania (φ_0 , λ_0) and there were created the B and L vectors that contain the points latitudes and longitudes respectively. Also, there were computed the differences between points coordinates and the central point coordinates, that were stored in the f and l vectors.

In the next stage, there were computed the additional arrays Fa , Fb , La , Lb and finally, the X_{tg} and Y_{tg} vectors that contain the rectangular coordinates on the tangent plane were calculated, using the following equations [8]:

$$\begin{aligned} X_{TG} &= (a_{00} + a_{10}f + a_{20}f^2 + a_{30}f^3 + a_{40}f^4 + a_{50}f^5 + a_{60}f^6) 1.000 + \\ &\quad (a_{02} + a_{12}f + a_{22}f^2 + a_{32}f^3 + a_{42}f^4) l^2 + \\ &\quad (a_{40} + a_{14}f + a_{24}f^2) l^4 + \\ &\quad (a_{06}) l^6 + \dots \dots \dots \\ Y_{TG} &= (b_{01} + b_{11}f + b_{21}f^2 + b_{31}f^3 + b_{41}f^4 + b_{51}f^5) 1,000 + \\ &\quad (b_{03} + b_{13}f + b_{23}f^2 + b_{33}f^3) l^3 + \\ &\quad (b_{05} + b_{15}f) l^5 + \dots \dots \dots \end{aligned} \quad (3)$$

These coordinates are passed into the secant plane of the projection, by multiplication with the coefficient of reduction to scale ($C = 0.999750$) and adding the translation of the coordinates system axes ($X = X_{tg} C + X_0$; $Y = Y_{tg} C + Y_0$).

```

function [ coord_stereo ] = conversionBLH_XYZ (Coord)
par_a = dlmread('parametri_a.txt');
par_b = dlmread('parametri_b.txt');
pct = Coord; j = size(pct,1);
B0=46; L0=25;
B=pct(:,2); L=pct(:,3);
for i=1:j
    f(i,1)=3600*(B(i,1)-B0)*10^-4; l(i,1)=3600*(L(i,1)-L0)*10^-4;
end
for i=1:j
    F(:,i)=[f(i,1)^0;f(i,1)^1;f(i,1)^2;f(i,1)^3;f(i,1)^4;f(i,1)^5;f(i,1)^6;f(i,1)^7];
    Fb(:,i)=[f(i,1)^0;f(i,1)^1;f(i,1)^2;f(i,1)^3;f(i,1)^4;f(i,1)^5;f(i,1)^6];
    La(:,i)=[l(i,1)^0;l(i,1)^2;l(i,1)^4;l(i,1)^6;0;0;0];
    Lb(:,i)=[l(i,1)^1;l(i,1)^3;l(i,1)^5;l(i,1)^7;0;0,0];
end
Pa=par_a*F; Xtg=dot(Pa,La);
Xsec=Ntg*0.999750; X=500000+Xsec;
Pb=par_b*Fb; Ytg=dot(Pb,Lb);
Ysec=Ytg*0.999750; Y=500000+Ysec;
STEREO (:,1)=pct(:,1); STEREO (:,2)=X; STEREO (:,3)=Y;STEREO (:,4)=pct(:,4);
dlmwrite('Points_stereo.txt',STEREO,'delimiter','\t','newline','pc','precision',9);
Points_stereo=[STEREO (:,1) STEREO (:,2) STEREO (:,3) STEREO (:,4)];
end

```

Fig. 7. Conversion function between geodetic and rectangular coordinates

3.3. Population of the weighting matrix procedure

```

P=zeros(n,n);
for i=1:length(d)
    for j=1:length(d)
        if i==j
            P(i,j)=weight_s; k=i; end; end
    end
    for i=(k+1):(k+length(weight_beta))
        for j=(k+1):(k+length(weight_beta))
            if i==j
                P(i,j)= weight_beta(i-n); k=i; end; end
        end
    for i=(k+1):n
        for j=(k+1):n
            if i==j
                P(i,j)= weight_z; end; end
        end
    end
end

```

Fig. 8. Population of the weighting matrix procedure

The weighting matrix is a diagonal matrix that contains the weight values for each measured data. The procedure contains three for instructions, in order to select the weight values for distances, horizontal and vertical angles respectively.

4. Conclusions

In recent years automation, computer science and applied informatics domains have known a great evolution, being used in almost all fields of activity, such as: engineering, archaeology, surveying, medical and chemical industries, design projects.

The aim of this study was to obtain an automated method for processing the measurements from a three-dimensional geodetic network, in the form of a closed-loop traverse, using Matlab programming language.

In order to prepare the entire program, both knowledge of geodesy (three-dimensional networks, processing of the measurements from a three-dimensional geodetic network) and programming (Matlab programming language) are required. Firstly, the mathematical algorithm of adjustment was carefully analyzed and there was created the flowchart of the program.

After processing, the program offers the solutions: the most probable values for the points coordinates with their accuracy determination and the most probable values for the measurements also with their associated accuracy.

This program has more advantages, such as: it has a friendly interface being easy to understand, utilizes simple notations which allows focusing on mathematical algorithm, has a great computing speed, being composed by more particular functions and procedures.

In the execution process of the program, there were achieved important properties, like: correctness, compatibility, portability, efficiency, clarity.

Programming represents an essential domain of informatics. It helps us to understand the hipper-connecting world we are living in and to represent what happens in the backside of the displays, in a coded language.

5. References

1. Ghilani D.C. - *Adjustment computations. Spatial data analysis. Fifth edition*, John Wiley & Sons, Inc., Hoboken, ISBN 978-0-470-46491-5, New Jersey, S.U.A, 2010
2. Shi Li-mei, Chen Yi-jin, Gong Hui, Wang Ning – *Algorithm design and realisation of the approximate coordinate of points in the computation of traverse network indirection adjustment*, University of Mining and Technology, Engineering of surveying and mapping, Beijing 100083, China, June 2006
3. A. Călin, A. Gînța, P. D. Dumitru – *Programming the free adjustment of the precise leveling measurements*, Journal of Geodesy and Cadastre RevCAD, no. 16, pag. 26-33, Alba-Iulia, Romania, ISSN 1583-2279, 2014
4. Matlab user guide. Source: https://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf
5. Nistor Gh., *Teoria prelucrării măsurătorilor geodezice*, Universitatea Tehnică “Gh. Asachi”, Iași, 1995
6. Takehisa Harada – *Universal program for adjustment of any geodetic network (PAG-U)*, Journal of the Geodetic Society of Japan, Vol. 26, No. 3, pp.147-171, 1980
7. C. Chirilă – *“Geodezie matematică – Îndrumător de lucrări practice și proiect”*, Publishing house Tehnopress, Iași, Romania, 2014
8. Moca Valeriu, Chirilă Constantin - *Cartografie matematică. Întocmire și redactare hărți, Lecture Notes, Technical University „Gheorghe Asachi” of Iasi, Faculty of Hydrotechnical Engineering, Geodesy and Environmental Engineering, 2002*