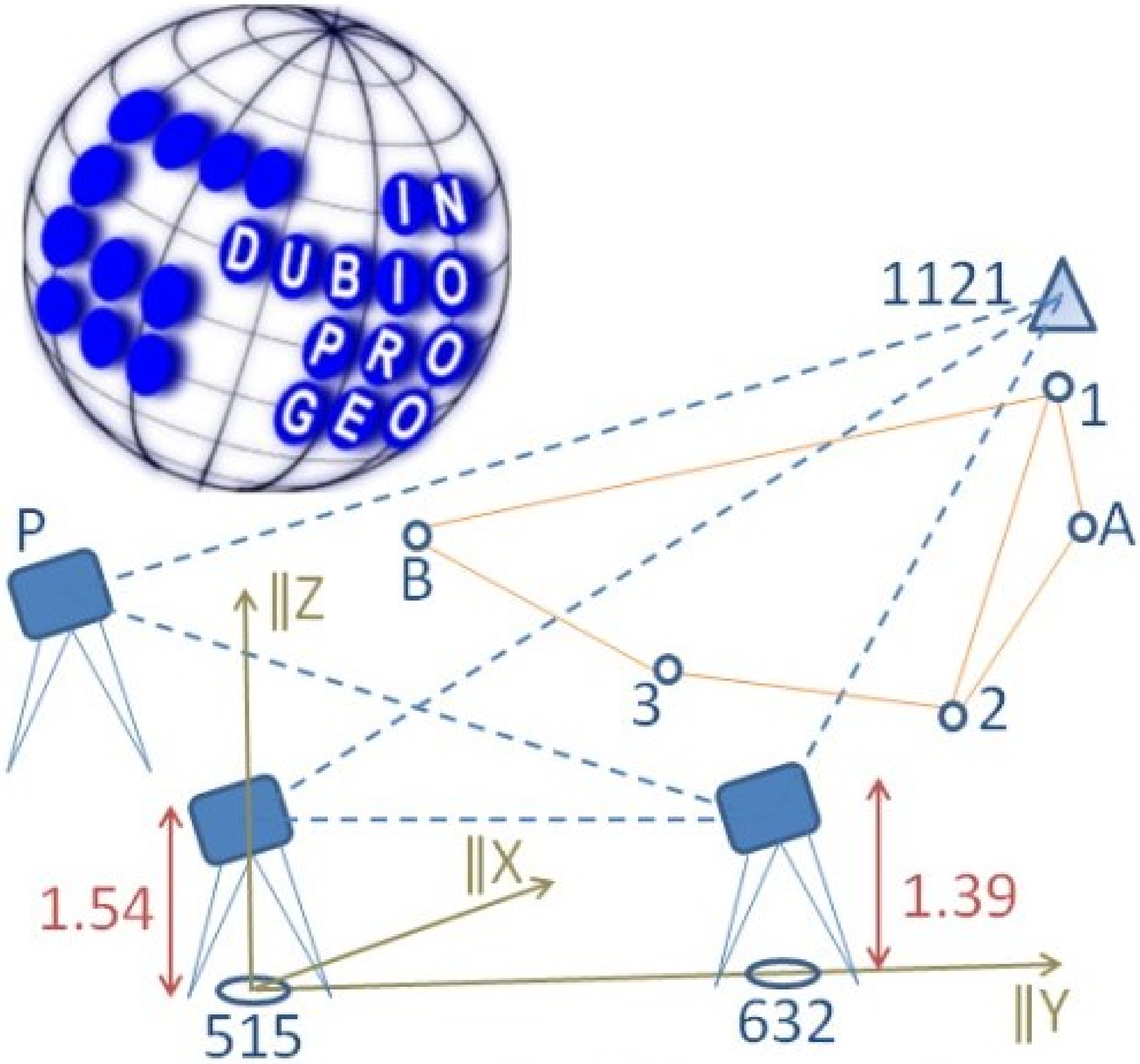
IN DUBIO PRO GEO

Geodetic Cloud Computing

Handbook



A project of HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES Faculty of Spatial Information

Author: Prof. Dr.-Ing. Rüdiger Lehmann
University of Applied Sciences Dresden
Faculty of Spatial Information
Friedrich-List-Platz 1
D-01069 Dresden
GERMANY

Visitor's address:
Friedrich-List-Platz 1, D-01069 Dresden
Room Z737
https://w3w.co/develop.closes.wires

Tel +49 351 462 3146
Fax +49 351 462 2191
mailto:Ruediger.Lehmann@htw-dresden.de
https://www.researchgate.net/profile/Ruediger_Lehmann2

http://www.in-dubio-pro-geo.de

Version: September 2019

IN DUBIO PRO GEO Geodetic Cloud Computing

Deutsch | First steps | Settings

Project managem.

Current project ...

Load | Duplicate
Clear | Save
Overview | Download

Load template project for HTW

Coordinate lists

Edit View list 1

Edit View list 2

Compare/merge both lists

Find neighbouring points

Find convex hull

Create lattice points

Plane and sphere

Planar triangles

Spherical triangles

Planar quadrangles

Circular arcs

Vertical triangles

Planar polygons

Arithmetic

Roots of a polynomial

Definite integrals

Matrix computations

Tutorials

Equilateral triangular lattice

Area partitionment

Circular arc stakeout

Hansen´s problem

Spatial line section

Rectangle through five points

Square through four points

Cylinder through seven points

Trigonometric levelling line

3D space

Spatial polygons

Spatial geodetic sections

Satellite orbits

Ephemeris computations

Ellipsoid of rotation

Coordinate conversion

Geodesics

Ellipsoidal polygons

Latitude dependent quantities

Length of meridian arc

Meridian convergence

Grid scale factors

Normal gravity formulae

Coordinate transformations

Transf. by parameters

Transf. by control points

Concatenated transformations

Registers

Geodetic journals

Geodetic abbreviations

Milestones of Geodesy

Famous Geodesists

World Geodetic System 1984

Glossary

Links

Lecture notes

(Autor: R. Lehmann)

Ebene Geodätische Berechnungen

Räumliche Geodätische Berechnungen

Geodätische Berechnungen auf dem Rotationsellipsoid

Tacheometry

Sets of angles and distances

Station centring

Atmospheric correction

Traverses

Universal computer

Geodetic statistics

Repeated measurements

Double measurements

Critical values for hypothesis tests

Error ellipses and ellipsoids

Sigma points

Geodetic adjustment

Adjustment with observation equations

Vertical networks

Trilateration

Adjusting curves

Adjusting surfaces

Library

Search

List of authors

Most wanted documents

Published 2018 (selection)

Documents with >300 pages

Adjustment textbooks

Life is a quiz

Picture quiz - easy

Unbelievable, but true - or not? - easy - medium

Milestones quiz - medium

Abbreviations quiz - difficult

and more

Inaccessible point with auxiliary triangles
Incompletely connected traverse
Adjustment of a triangle
Geodätische Messabweichungen
Fehler- und Kovarianzfortpflanzung
Ausgleichungsrechnung
List of guides
Bag of tricks
Report a problem
Timeline

Did you know? *If javascript is off, you are still able to use almost all features. IN DUBIO PRO GEO is doing without cookies.*

IN DUBIO PRO GEO Guide : List of guides

Page contents

⚠ The solutions to the problems sketched in the tutorials and guides refer to the case that you work with the standard Settings . Otherwise, small variations may occur.

General, project management and lists

Basics of working with IN DUBIO PRO GEO

If you have a geodetic or geometrical problem, there is mostly a way to solve it with IN DUBIO PRO GEO.

Load, duplicate, clear or save an IN DUBIO PRO GEO project or download it to your computer. Define project related Settings.

Coordinate lists are lists of coordinate records, a special kind of tabular data records. Edit, filter, sort and save coordinate lists for later use in the computations.
☆ GPS reference point of HTW Dresden

Measurement lists are lists of measurement records, a special kind of tabular data records. Measurements are various types angles, distances, height (differences) and much more.

Equidistant points on a straight line (1D), a rectangular lattice (2D) or cuboidal lattice (3D) are created and may be postprocessed (e.g. rotated) with other computation tools.
☆ 2D lattice for Großer Garten Dresden
☆ Loxodrome from Dresden (Saxony) to Dresden (Ontario)

Computations

Planar polygons are computed from given coordinates of vertices: planar polygonal angles, azimuths and lengths of sides, area, perimeter, barycentres, etc.
☆ Surrounding polygon for Großer Garten Dresden

Various computations are performed with a matrix or one of its submatrices: Inversion, Cholesky, LU and eigenvalue decomposition, determinant, norms, etc.
☆ Orthogonal matrix
☆ Arithmetic expressions in matrices

Satellite orbits — page 32

From ephemeris or almanac data of GNSS satellites (e.g. GPS) discrete orbit points are computed at a specified time grid. The computation is based on the GPS Interface Specification IS-GPS-200 and the GALILEO Signal in Space Interface Control Document.

☆ Orbit computation from a GPS almanach

Grid scale factors — page 35

The point scale factor of grid systems at the points of Coordinate lists and the line scale factor along the lines between consecutive points of such lists are computed.

Normal gravity formulae — page 38

The normal gravity at a point of given ellipsoidal latitude and height for the level ellipsoids GRS67,GRS80 and World Geodetic System 1984 is computed, optionally including an Error propagation.

☆ gravity benchmark at the geodetic laboratory of HTW Dresden

Transformation by parameters — page 40

Points in the plane and in 3D space are transformed by given parameters. A sequence of up to 13 individual transformation steps can be performed. In this way, all conceivable transformations can be configured.

☆ Rotate cuboid about centre axis

Transformation by control points — page 45

Control points are used for the computation of transformation parameters between two coordinate systems. All planar or spatial transformations are computed, which are computable from these points. In both systems there may be given non-control points, which are transformed by the computed parameters.

☆ Cuboid through four vertices

Sets of angles and distances — page 52

On a station there may be measurements in sets to the same targets. Measurement values may be given in arbitrary succession and may be arbitrarily missing. Set means, instrument errors and accuracy estimates are computed. The results may be processed with further IN DUBIO PRO GEO computation tools.

☆ Pure processing of horizontal angles
☆ Processing of zenith angles with slope distances
☆ Joint processing of all measurements

Station centring — page 56

Eccentrially measured sets of polar measurements are computationally transferred to a new centre. You obtain the values, which would have been measured on the centre, optionally including an Error propagation. If all required values are given, a spatial centring is computed.

☆ Eccentric angular measurements to remote targets

Atmospheric correction — page 58

The refractivity of air for specified atmospheric conditions in the visual and near infrared spectrum is computed, optionally including an Error propagation. A distance measurement value may be corrected.

☆ Leica TS30, correction of erroneous settings

Traverses — page 61

From point coordinates and polar measurements it is tried to compute a classical traverse with proportioning of misclosures. In arbitrary measurements the longest possible traverse is detected. All that is rationally evaluable in one way or another, will be evaluated.

☆ Branched traverse with spatial intersection

Universal computer page 65

From point coordinates and polar measurements all possible quantities are computed. Computation rules between these quantities are set up and consecutively applied until no new values are obtained. This is done in any possible way. By this procedure you often get many different results, which can be compared to detect gross errors. In this case the medians of the computed values represent the results of a robust estimation.

- ☆ Polar values computed from cartesian coordinates
- ☆ Inaccessible point with auxiliary triangles
- ☆ Planar trilateration network
- ☆ Trigonometric levelling line
- ☆ Computation of the orthocentre of a triangle

Repeated measurements page 75

In geodesy and in other measuring professions a quantity is often measured multiple times under the same conditions, to increase accuracy and reliability. It is assumed that the measured values differ only due to the effect of random independent identically distributed errors. Such measurements are comprehensively evaluated, including all applicable statistical tests. Instead of geodetic measured values also other random samples of that kind may be evaluated.

- ☆ Repeated height determination of a point
- ☆ Try this computation tool using normally distributed random numbers
- ☆ Try this computation tool using different random numbers

Double measurements page 80

Various quantities are all measured twice, possibly with different accuracies and/or with systematic difference. Such measurements are comprehensively evaluated, including all applicable statistical tests, e.g. whether the data show a systematic difference.

- ☆ Double determination of point heights by fast static GNSS measurements

Adjustment with observation equations page 83

The general adjustment problem with observation equations is solved in the least squares sense, optionally with constraints for parameters and functions of adjusted quantities.

- ☆ Adjusting square through four vertices

Vertical networks page 89

All kinds of vertical networks are adjusted, both spirit levelling networks (height differences & lengths of lines) as well as trigonometric vertical networks (zenith angles & distances), both free as well as connected networks, even single levelling lines.

- ☆ Campus network of HTW Dresden as a free spirit levelling network
- ☆ Trigonometric levelling line
- ☆ Inaccessible point with auxiliary triangles

Adjusting surfaces page 96

Through given data points in 3D-space an adjusting (i.e. best fitting) surface (plane, sphere, ellipsoid or general quadric) is computed. Also a plane through 3 points, a sphere through 4 points etc. can be computed. Additional points may be projected onto the surfaces.

- ☆ Terrain model approximation and interpolation at Großer Garten Dresden

Miscellaneous

Library page 100

Here you have access to a collection of links to scientific documents in the World Wide Web regarding important subjects of Geodesy. At the moment there are 2824 documents with a total number of 90000 printing pages and 5 GByte available. Latest update with full link checking: 2018/09/26.

Tutorial
page 101

Tutorials explain the functions of IN DUBIO PRO GEO by means of practical examples. The solutions of the problems are comprehensible on the basis of pre-filled forms. In the select areas not all options are available. The forms may be submitted by the button Compute and the results may be viewed. For the sake of convenience the results are given in extracts in the tutorial. The most important intermediate and final results are highlighted by golden boxes. The solutions are commented in further golden boxes.

Sorry, at the moment tutorials are only in German.

Bag of tricks
page 103

IN DUBIO PRO GEO provides more than you probably expect. This bag of tricks contains a selection of valuable tricks, which make the work easier. They are sorted by level of user (beginner ⇒ expert).

Information criteria
page 107

Information criteria are computed for your models. This enables you to select the optimal model. Some computation tools compute various models unasked and list the corresponding information criteria.

Earth curvature
page 109

☆ Effect of earth curvature for horizontal sighting

Error propagation
page 111

☆ Radius of circular arc
☆ Arcs intersection

IN DUBIO PRO GEO Guide : First steps

Page contents

How do you get help?

Click on for a guide appropriate to occasions.
After the sign you find practically instructive examples.
After the sign tricks for beginners and experts are disclosed.
Place mouse pointer over to get hints.
Place mousepointer over terms marked with › to get their meaning.

If you have questions relating to your computation, e.g. because it did not yield the desired result, you may ⇩Save project and send the link for loading it via Problem report together with your question and your contact address. You will receive an answer within some hours.

Using the input area at the bottom of the start page you search IN DUBIO PRO GEO by the Google™ search engine.

IN DUBIO PRO GEO Handbook
Last change: Sept 2019 (9 MB, 110 pages)

Please note

If you run IN DUBIO PRO GEO in different tabs of the same browser window then you work on the **same project** in all those tabs! This may be undesired.

If webpages contain computation results then IN DUBIO PRO GEO has computed these results immediately before the page is displayed. If you navigate in the browser by "history" or "back" to a page with computation results and if you have changed some data in between, e.g. some settings, then the computations are **repeated with the changed data** . You may not see exactly the same page as before! This may be undesired. It is safer not to use "history" or "back", but START , ▲Top of page , ▼Bottom of page and Modify input .

Input areas

The characters & $ < > \ { | } are not allowed in input areas and will be replaced by † (dagger).

Blue-grounded input areas may be left blank.

Numerical values in input areas may be numbers with decimal point or decimal comma or integers. Also the exponential and percentage notation is possible.

Arithmetic expressions in input fields

Instead of numerical values, like e.g.

```
16.1063    16,1063    161063e-4    1610.63%
```

you may always also give arithmetic expressions, like e.g.

```
8.1+8.0063      (3,3009-1)*7,0         8.1+80063e-4    pi*16.1063/pi
161063/10000    log(exp(16.1063))      2,3009*7,0      sqrt(16,1063*16.1063)
3,3009*7,0-7    asin(sin(0.161063))*100                (16.1063^(-0.5))^(-2)
```

All 10 arithmetic expressions yield the same numerical value. This also works in tabular data records like measurement lists or coordinate lists as well as matrices.
In these examples the chosen output decimal separator is not in effect.

The following mathematical functions are supported: `abs acos acosh asin asinh atan2 atan atanh cos cosh exp log10 log sin sinh sqrt tan tanh`

⚠ Arguments of trigonometric functions are always expected in the angle unit radian, no matter which unit has been used elsewhere.

☹ Arithmetic expressions in input areas are not workable in the ⇓angle unit `DegMinSec`.

See ☆ Arithmetic expressions in matrices.

Tabular data records

Coordinate lists and Measurement lists as well as other tabular data records like matrices and vectors may be copied to the textarea via clipboard. For most of the textareas there is a button for file upload. Here textfiles with a maximum size of 30 kByte are supported. The used character set is automatically recognized, of course also a multibyte character set.

Each data record is written in a separate row. The succession of rows is sometimes arbitrary. Every input starting from "//" on a row as well as blank rows are ignored. Consecutive elements of each record are separated by tabulator, space or semicolon. Multiple spaces are treated as one space. However, e.g. " 1; ;2 " is treated as three elements, the second is empty. For numerical values or arithmetic expressions the rules given for ⇑Input areas apply.

Example: The following inputs yield identical results.

16,10 17,11 23,06 14,02	16,10 17.11 23.06 14,02	16.10;17.11 //extra 23.06 ; 14.02	16.10 ; 17+11/100 //extra 23.06 14.02

If you want to import coordinate or measurement lists from various data formats, a text editor is useful that can mark also columns of data. We recommend the freeware editor notepad++ . Also very powerful is textpad.

Unit of length

It is assumed that all distances, cartesian and grid coordinates, instrument and target heights etc. as well as the radius of Earth curvature in the project are given in the same unit and all areas are in the square of this unit. Also standard deviations of these quantities must be specified in this unit and are obtained in this unit. The unit metre is common. If a predefined reference ellipsoid is used in a computation, then the length unit must be definitely metre.

It is not allowed to append a length unit to an input number.

Unit of angle

The angle unit can be chosen separately for latititude/longitude and other angles. For some computations not all units of angle are selectable. Western longitudes and southern latitudes must be given negative.

For angles it is acceptionally possible to specify values with `°`,`'`,`"` , but not in arithmetic expressions. Valid notations are displayed in the right table.

Example: For the angle `16.1063°` the notations given at the right are possible on input, instead of decimal point also with decimal comma. The output of angles in result tables is always by the first notation. E.g. `16°6'22.7"` is given in the setting `DegMinSec` as `16.06227`.
If by Settings the output decimal separator is set to comma, then one gets `16,06227`.

⚠ If you use Arithmetic expressions in input fields with trigonometric functions, angle arguments must always be given in radian.

☹ If angles are involved in a computation, then all quantities, which appear in alternative angle notation, are converted to the standard notation, even though they are not angles. Be sure to give only angles in this angle notation here.

Degrees	`16.1063` or `2.30009*7` or `16.1063°`
Gradians	`17.895889` or `17+0.895889`
Arcminutes	`966.378` or `966.378'`
Arcseconds	`57982.7` or `57982.7"`
DegMin	`16.06378` or `16°06.378'` or `16°6.378'`
DegMinSec	`16.06227` or `16°06'22.7"` or `16°6'22.7"`
invalid	`16° 06' 22.7"` or `2.30009°*7` or `15°66'22.7"` etc.
Radian	`0.28110797`
Turns	`0.00447397`

Examples for angle specifications

Settings: In this table the chosen output decimal separator is not in effect.

Javascript, HTML5, CSS3 etc.

IN DUBIO PRO GEO works without java or flash etc. and as much as possible also without javascript. Only the following features require that javascript is enabled in your browser:

- directly loading examples and clearing in forms
- hiding paragraphs directly on the page (via Settings works nonetheless)
- Transfering data to a new tab
- graphical output of the computation figures on a canvas
- special features only for students of geomatics at HTW Hochschule für Technik und Wirtschaft Dresden University of Applied Sciences connected to HTW network.

The computation figures are only displayed if the browser supports HTML5. Some styles require CSS3. Modern browsers supports HTML5 and CSS3! The Internet Explorer Version 8 and older are not recommended.

If Javascript is disabled, the buttons not workable appear grey .

The HTML5 and CSS3 codes of IN DUBIO PRO GEO are regularly validated with the W3C validation services .

W3C HTML 5.0 W3C CSS

IN DUBIO PRO GEO does not request or collect personal data.

Did you know? *IN DUBIO PRO GEO also runs on the smartphone or tablet computer (in the browser)*

☰ IN DUBIO PRO GEO Guide : Get in touch with IN DUBIO PRO GEO

Page contents

If you have a geodetic or geometrical problem, there is mostly a way to solve it with IN DUBIO PRO GEO.

☰ What is geodesy?

Geodesy is a term coined by the Greeks in order to replace the original term ''geometry'', which had meanwhile lost its original meaning of **earth or land measuring** (surveying) and acquired the new meaning of an abstract ''theory of shapes''. Aristotle tells us in his ''Metaphysics'' that the two terms differ only in this respect: ''Geodesy refers to things that can be sensed, while geometry to things that they cannot''. Many centuries afterwards the word geodesy was set in use anew, to denote the determination of the shape of initially parts of the earth surface and eventually, with the advent of space methods, the shape of the whole earth. Thus it remained an applied science, while facing at the same time significant and challenging theoretical problems, in both physical modeling and data analysis methodology.

after Dermanis & Rummel 2000

☰ What is IN DUBIO PRO GEO?

IN DUBIO PRO GEO is a geodetic cloud computing software. It offers tools for geodetic computations and adjustment

- ☺ scientifically proper, but easily comprehensible
- ☺ with guides and tutorials
- ☺ with library and registers
- ☺ free and manufacturer-independent
- ☺ no advertisement, no registration, no cookies
- ☺ plattform-independent (runs also on smartphones etc.)
- ☺ no installation at your computer required
- ☺ no plugin needed, runs without Javascript
- ☺ english and german
- ☺ is under constant development

☰ The IN DUBIO PRO GEO principle

‟ Make it as simple as possible, but not simpler. ”
Albert Einstein

IN DUBIO PRO GEO tries to reconcile scientific correctness with ease of use, also for beginners.

From the start values given or uploaded by the user it is attempted to compute as much as possible. If nothing reasonable can be computed from these values, nothing is computed. To a great extent, the terms and symbols are aligned to the ⇛Wikipedia .

The diffent computation tools are cross-linked, such that the results of one computation can be used as start values of a subsequent computation, whenever this makes sense.

Feedback / Report a problem

This website is unter permanent development. Mistakes cannot be excluded. All is done to detect and fix bugs. If you detect any bug or problem and want to support this, please send a short report. Hints and suggestions are always welcome. Problem report

Prof. Dr.-Ing. Rüdiger Lehmann
HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES
Faculty of Spatial Information
PF 120701
D-01008 Dresden

Tel +49 351 462 3146
Fax +49 351 462 2191
Ruediger.Lehmann@htw-dresden.de

⇛Researchgate score: 22.36
h-index: 9 (excluding self-citations)
⇛run 100 km in < 10 h

Did you know? *IN DUBIO PRO GEO is free of charge, but if you use results, we demand a reference to the source. Thank you!*

IN DUBIO PRO GEO Guide : Project management

Page contents

Introduction

IN DUBIO PRO GEO preserves your inputs and settings in the workspace. As soon as you made inputs or settings, you see the options ⇩Save project and ✗Clear project in the navigation panel. Then IN DUBIO PRO GEO has created a **project** .

⚠ If you run IN DUBIO PRO GEO in different tabs of the same browser window, then in all those tabs you work on the **same project** ! This may be undesired.

⚠After 24 min of **inactivity** a session is closed automatically by the browser. The storage is cleared automatically. But if you have saved or downloaded the project, you can restore it.

Save a project on the IN DUBIO PRO GEO Server

If you want to finish your session and save your work, click on the ⇩Save project option in the navigation panel. IN DUBIO PRO GEO saves your project on the server and provides you with a link, which you should keep, e.g. as a browser bookmark. Alternatively, you may memorise the five-digit ID. When saving a project again, the previous version is overwritten!

⚠ The link and the ID are only valid for 30 days. The period of validity restarts as soon as the project is changed and saved again.

⚠ Only those data are saved or downloaded, which you sent to the server before. If you make an input on the current page and click ⇩Save project then those input is not saved.

Load and duplicate a server-saved project

If you want to resume your work, use the link or the ID provided for loading the project from any device connected to the internet.

If you do not want to overwrite a loaded or previously saved project when saving (again), you can duplicate it. When saving the project, a new ID is assigned, such that the old project is kept untouched.

Download and upload a project

Alternative to saving the project on the IN DUBIO PRO GEO server the project can also be downloaded and saved locally. However, this is only possible, if the project does not contain too many data, such that the size of the project file does not exceed **90kB** . Otherwise, the project is saved on the server nonetheless.

The local data is named **in-dubio-pro.geo** and may be arbitrarily renamed.

Using ⇧Load project the locally saved project can be uploaded again. For this purpose select the project file on your local drive.

⚠ When loading or uploading a project, all current settings and inputs are overwritten!

⚠ IN DUBIO PRO GEO is under constant development. There is no guarantee that the project file is fully compatible with future versions of IN DUBIO PRO GEO. Possibly some data cannot be used anymore. Last resort: Problem report

Template and exemplary projects

IN DUBIO PRO GEO provides you with some template and exemplary projects:

- the template project for HTW (can only be loaded if you are connected to HTW campus network)
- exemplary projects from Ebene Geodätische Berechnungen (2018)
- exemplary projects from Räumliche Geodätische Berechnungen (2018)
- exemplary projects from Geodätische Berechnungen auf dem Rotationsellipsoid (2019)
- exemplary projects from Fehler- und Kovarianzfortpflanzung (2019)

These projects are save protected. When loading, they are automatically duplicated. Also here all current inputs are overwritten.

Clear project

For clearing a project in the workspace, click on the ✗ Clear project option in the navigation panel. All inputs will be discarded. All settings will be reset to the standard settings. The last version of the project saved on the server, is preserved, if any.

⚠ There is no undo feature.

Did you know? *You may get all numbers either with decimal point or with decimal comma.*

IN DUBIO PRO GEO Guide : Coordinate lists

Page contents

Coordinate lists are lists of coordinate records, a special kind of tabular data records.
Edit, filter, sort and save coordinate lists for later use in the computations.

Introduction

At the moment coordinate lists are used by the following computation tools:

Find convex hull	Meridian convergence	Transf. by control points
Planar polygons	Grid scale factors	Adjusting surfaces
Coordinate conversion	Spatial polygons	Universal computer
Ellipsoidal polygons	Transf. by parameters	Traverses Trilateration

Additionally, such lists are generated by

Create lattice points	Satellite orbits	Geodesics

General structure of a coordinate record

In each non-empty row of a coordinate list one point is defined. Each row consists of three up to five fields separated by semicolon or white space (column separator):

- one **pointname**, it
  - is a string of arbitrary length (overlong pointnames)
  - starts with a letter or digit.
  - It is case sensitive.
  - special characters are valid, but not the generally prohibited characters .
- perhaps one **code**
  - a string of arbitrary length
  - does not contain generally prohibited characters
  - currently used only for Satellite orbits
- two or three **coordinates**
  - numerical values or arithmetic expressions, cf. Input areas
  - Values not matching such an expression cause a warning and will be ignored
- as well as further columns, which are ignored.

Empty rows and pure comment rows are ignored.

Working with coordinate lists

The **succession of rows** in a coordinate list may be arbitrary, except for polygon computations and for the line scale factor, where the succession defines the polygon or the line.

- In one list, points with **two and three** coordinates may be mixed. This makes sense if for some points heights are known, for other points they are not.

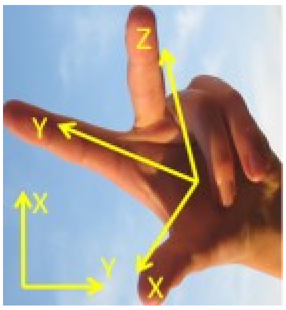
- If a point has two coordinates and three are required during a computation, then it terminates with an error message.
- If a point has three coordinates and two are required during a computation then a warning is issued and the third coordinate is ignored.
- Everything behind a third coordinate on a line is ignored.

Every coordinate list requires a **name of coordinate system**, i.e. an almost arbitrary string with restrictions as for point names, a **type of coordinate system** and a **column format**.

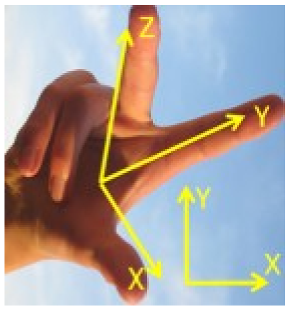
Type of coordinate system

The following types are accepted:

Type of system	What is this?	not available in
XYZ or YXZ lefthanded	cartesian lefthanded system, e.g. topocentric horizont system	Ellipsoidal polygons Meridian convergence Grid scale factors Coordinate conversion Satellite orbits
XYZ or YXZ righthanded	cartesian righthanded system, e.g. geocentric equator system or mathematical system	Ellipsoidal polygons Meridian convergence Grid scale factors Universal computer Traverses
Northing Easting Height or Easting Northing Height	Grid system like UTM or Gauss-Krüger referring to the "reference ellipsoid" and more parameters chosen under Settings	Adjusting surfaces Ellipsoidal polygons Satellite orbits
Longitude Latitude Height or Latitude Longitude Height	ellipsoidal (or spherical) geocentric system referring to the "reference ellipsoid" and the "Latitude/Longitude unit" chosen under Settings	Planar polygons Spatial polygons Transf. by control points Transf. by parameters Adjusting surfaces Trilateration Satellite orbits Traverses Universal computer



lefthanded system



righthanded system

The following rules must be obeyed:

- XYZ and YXZ etc. differ only in the succession of specification of the coordinates. In both cases the thumb-axis is X or North, the forefinger-axis is Y or East and the middle-finger-axis is Z or Height. Grid systems like UTM are therefore always lefthanded systems, no matter which coordinate is specified first in the input table.
- For planar computations the third coordinate (Z or Height) is ignored.
- Azimuts *t* count from the X-axis or north axis to the Y-axis (zero) or east axis (π/2 = 90° = 100 gon).

Column format and automatic point naming

The following formats are accepted:

pointname coordinates

Example:

```
P1 23.06 16.10 17.11
Q2 14.02 19.63 17.05
007 63.3 44 //2D point
```

pointname code coordinates

Example:

```
P1 Code1 23.06 16.10 17.11
Q2 Code1 14.02 19.63 17.05
007 Code2 63.3 44 //2D point
//Code is currently ignored
```

coordinates

Example:

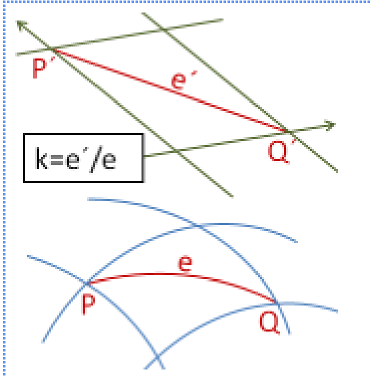
```
23.06 16.10 17.11
14.02 19.63 17.05
63.3 44 //2D point
//Points are auto-
//matically named.
```

All three lists are identical, but in the latter list the points are automatically named. The standard automatic pointnames are `1,2,3,…` Different options can be selected via Settings . Please note the following **examples**:

initial name	increment	automatical pointnames
1 (Standard)	1 (Standard)	1, 2, 3, …
10	100	10, 110, 210, …
10	-100	10, -90, -190, …
abc10	100	abc10, abc110, abc210, …
abc10	-100	abc10, abc-90, abc-190, …
abc	100	abc0, abc100, abc200, …

Coordinate scale factor for grid systems

Grid systems (type of system: Northing Easting Height or Easting Northing Height) are based on a local grading of the curved ellipsoid of rotation by Gaussian mapping (= Transversale Mercator projection on the ellipsoid). Due to the inevitable deformations the unit of grid coordinates does not coincide with the metric length unit . As a result of the conformity of the Gaussian mapping, the **grid scale factor** k is nearly constant for a small point area and can be computed automatically.



grid scale factor k*,*
ground distance e*,*
grid distance e'

Details are found here: Grid scale factors

⚠ When doing computations with grid coordinates, it is supposed that all other metric values and scales are **not** distorted by the grid scale.

Details are found here: Other metric values and scales

If you want to compute **an accurate grid scale for each individual point** you should use Grid scale factors.

☆ GPS reference point of HTW Dresden

The HTW HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES provides a reference point for testing your mobile navigation device. In the World Geodetic System 1984 with ellipsoidal height `h` the point has the following coordinates:

`λ = 13.734806°,  φ = 51.033778°,  h = 160.5`

or `λ = 13°44'.0884,  φ = 51°2'.0267,  h = 160.5`

or `λ = 13°44'5".304,  φ = 51°2'1".602, h = 160.5`

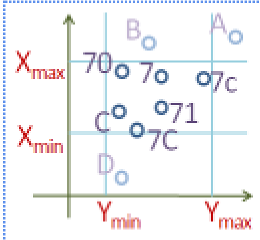
or `E = 33U 411287.9 m, N = 5654342.8 m, h = 160.5`

Below we give a number of **equivalent notations** for the coordinates of this point. Alternative notations of latitude and longitude explains units of measurement . In this table the chosen output decimal separator is not in effect.

GPS reference point of HTW Dresden

Settings	**Notation of coordinates**	**Comment**
Type of system **Longitude Latitude Height** , unit of latitude/longitude **Grad** , column format **Point Coordinates**	HTW-reference-point 13.734806 51.033778 160.5	
and now with column format **Point Code Coordinates**	HTW-reference-point pillar 13.734806 51.033778 160.5	
and now with column separator **semicolon**	HTW-reference-point;pillar; 13.734806;51.033778;160.5	
and now with column format **Coordinates**	13,734806 51,033778 160.5	Input optionally with **decimal comma**
and now with unit of latitude/longitude **DegMin**	13.440884 51.020267 160.5	Format **ggg.mmddddd**
and now with unit of latitude/longitude **DegMinSec**	13.4405304 51.0201602 160.5	Format **ggg.mmssddd**
and now without height specification	13.4405304 51.0201602	not possible with all computing tools
and now with alternative notation for angles	13.44'05.304" 51°02'01.602"	units of measurement
and now with type of system **Latitude Longitude Height**	51.0201602 13.4405304 160.5 m	What follows the third coordinates on a row, is ignored (here "m")
and now with type of system **Easting Northing Height**	411287.9 5654342.8 160.5 m	
and now with type of system **Northing Easting Height**	5654342.8 411287,9 160,5	Input optionally with **decimal comma**
and now with easting preceeded by the **zone number**	5654342.8 33411287.9 160.5	For the easting only 6 digits before the decimal separator are used.
and now with easting preceeded by the last digit of the **zone number**	5654342.8 3411287.9 160.5	
and now with abbreviated coordinates	54342.8 11287.9 160.5	Use false northing= -5600000 and false easting= -400000
and now without height specification	54342.8 11287.9	not possible with all computation tools
and now with type of system **X Y Z righthanded**	3904281.170 954274.291 4936033.017	
and now with type of system **Y X Z righthanded**	954274.291 3904281.170 4936033.017	

Filter, save and load coordinate lists



Filter and sort points: Ascending sort by X yields 7C C 71 7c 7 70. Ascending sort by pointname yields 7 7C 7c 70 71 C.

Coordinate lists may be

- given directly for each computation or saved once and **loaded** whenever needed.
- **saved** as results of computations. If a list exists already then it is overwritten.
- **filtered** by specification of limits for pointnames and coordinates. Then only points are saved, which are inside these limits.
- **sorted** by pointnames or coordinates.

Filtering and sorting does not alter the rows of a coordinate list, but only their number and succession.

Two saved coordinate lists can be managed at the same time. When filtering and sorting by pointnames the comparison is lexicographic, this means that e.g. 1610 is between 10 and 20. The comparison is case-sensitive. The comparison of coordinates is of course numerical. Filtering and sorting may be undone.

The third coordinates (Z or height) of a list may be manipulated as follows:

- All values are removed.
- All values are replaced by a given value.
- Empty fields are filled by a given value.
- All values are changed additively or subtractively by a given value.
- All values are multiplied by a common factor.

More decimal digits and overlong pointnames in coordinate lists

If in the coordinate lists of the computed points not enough decimal digits are displayed for you, it is recommended to load this list in a new browser tab or to a new coordinate list. Then you see more decimal digits.

Beim Speichern einer Liste werden eventuell zuvor schon gespeicherte Listen überschrieben. Aber zum Ansehen der Liste in der Eingabemaske von XY Edit coordinate list 1 or XY Edit coordinate list 2 When saving a list, possibly a previously saved list is overwritten. But for display of a list, this is not necessary.

If you use overlong pointnames, they are truncated in tables mostly to 12, sometimes to 9 leading characters. A warning is issued. To see the pointnames in full length, the same trick can be used.

See ⇑ Filter, load and save coordinate lists.

***Did you know?** If javascript is off, you are still able to use almost all features. IN DUBIO PRO GEO is doing without cookies.*

IN DUBIO PRO GEO Guide : Measurement lists

Page contents

Measurement lists are lists of measurement records, a special kind of tabular data records. Measurements are various types angles, distances, height (differences) and much more.

Introduction

At the moment measurement lists are used by the following computation tools:

Vertical networks
Sets of angles and distances
Station centring
Traverses
Universal computer
Trilateration

General structure of a measurement record

In each non-empty row of a coordinate list one measurement set is defined. Each row consists of one up to seven fields separated by semicolon or white space (column separator):

- one or two **pointnames**, each of them
  - is a string of arbitrary length (overlong pointnames)
  - starts with a letter or digit.
  - It is case sensitive.
  - special characters are valid, but not the generally prohibited characters .
- up to five **measurement values** in a user-defined succession
  - numerical values or arithmetic expressions, cf. input areas
  - Non-arithmetic expressions cause a warning and will be ignored.
- followed by more **unused entries**, e.g. codes, which are ignored like comments.

Blank rows as well as pure comment rows are ignored.

Measurement lists for “Traverses” and “Universal computer”

(see Traverses / Traverses / Universal computer / Universal computer)

- They start with a **station row** (one stationname, followed by a maximum of four station measurement values)
- followed by an arbitrary number of **target rows** (targetname, followed by target measurement values).
- If more station setups follow then a **separation row** is inserted now. Its first printable character must be a special character (not letter or digit) except '';'' or ''/'', therefore cannot be a blank row. The content of the separation row is ignored. Multiple separation rows act as *one* separation row.

- Now a further station row, further target rows, possibly a further separation row etc. may follow.

The succession of stations in a measurement list and the succession of target rows belonging to one station is arbitrary. This is true even for Traverses.

On each point there may be multiple station setups, such that this point recurs in station rows. However, for Traverses only the first station setup is used, which is indicated by a warning.

Each target point can be measured at a station multiple times, usually in the case of so-called Sets of angles and distances. In this case it is recommended to process each station setup separately by Sets of angles and distances , where in conjuction with the determination of instrument errors, set means and standard deviations are computed. Otherwise only simple means are computed.

Measurement lists for "Sets of angles and distances" and "Station centring"

(see Sets of angles and distances / Sets of angles and distances / Station centring / Station centring)

are in principle built as described in the preceding section, but consist only of target rows, all belonging to the same station. There is only one station. The succession of all rows is arbitrary.

Measurement lists for "Vertical networks" and "Trilateration"

(see Vertical networks / Vertical networks / Trilateration / Trilateration)

Each data record contains the data belonging to one network line followed by up to five measurements and/or accuracy measures. All rows start with two pointnames: startpoint and endpoint of the network line. The succession of all rows is arbitrary.

Quantities in the station rows may be:

***orientation angle** o*
: specifies the ⇓ azimuth of the horizontal circle scale index of the station.

***instrument height** ih*
: specifies the height of the tacheometer tilting axis above the station marking (usually zero, if the station point is not marked)

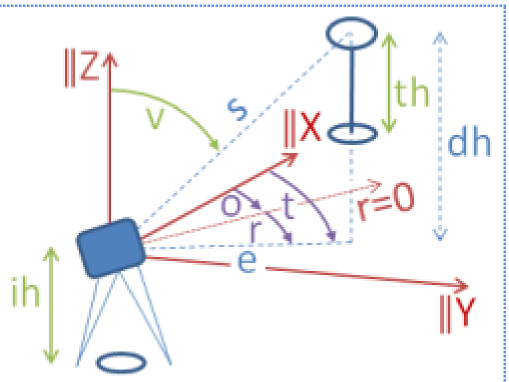
***default target height** th*
: replaces all missing target heights in this measurement list (default value)

Quantities in the target rows may be:

***horizontal angle** r*
: is the angle between the horizontal circle scale index of the station and the target, measured in the horizontal plane of the station (horizontal angle). If the associated zenith angle is missing then the face 1 is assumed.

***azimuth** t*
: is the horizontal angle between grid north and the target

***horizontal distance** e* ***or slope distance** s*
: horizontaler oder schräger Abstand von Stand- und Zielpunkt oder Anfangs- und Endpunkt



Coordinates and polar measurements

***zenith angle** v*
: is the angle between the zenith and the target. Elevation angles are not supported yet.

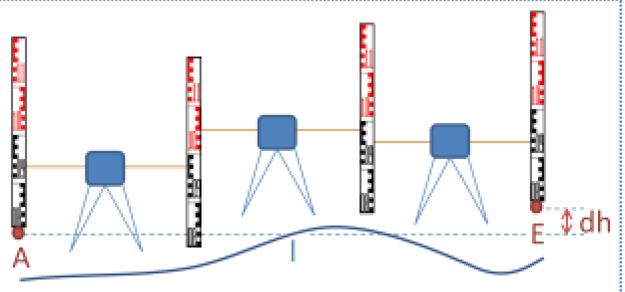
target height *th*
: specified the height of the reflexion point or targeted point above the target marking (usually zero, if the station point is not marked)

height difference *dh*
: specifies the height of the reflexion point or targeted point above the tacheometer tilting axis (negative if the tilting axis is higher)

Angles are oriented from the X or north axis to the Y or east axis, this is for the lefthanded system, seen from above, clockwise (⇑ figure).

Quantities for network lines may be:

specific for spirit levelling networks

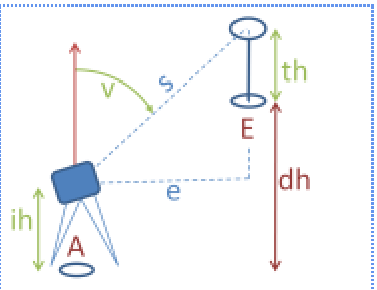


levelled network line

Height difference *dh* ***(required)***
: is the difference between endpoint and startpoint height of a line. If from startpoint to endpoint it goes uphill, height differences are specified positive, otherwise negative or zero.

Length of levelling line *l* ***(optional)***
: specifies the approximate length of a levelling line and may be used to define weights. The values must not be negative.

specific for trigonometric vertical networks



trigonometric network line AE

Zenith angle *v*
Horizontal distance *e* ***or slope distance*** *s*
Instrument height *ih*
Target height *th*
: have the same meaning as in the station and target rows.

All values are always required. Missing values *ih, th* are padded by the default values, if any.

for both kinds of vertical networks

Standard deviation (a priori) σ_{dh} ***or weight*** p_{dh} ***of the height difference***
: are used in the adjustment for the stochastic model. Only one of both values can be present. A default value pads all missing values in this column, or all values. Otherwise a missing accuracy measure is treated as zero standard deviation or infinite weight, which means that in the adjustment the related height difference acts as a constraint.
⚠ Also for trigonometric vertical networks these values refer to the adjustable height difference $dh = s \cdot cot(v) + ih - th$, i.e. they include measurement errors in the instrument height and target height.

Code (optional)
: is currently ignored.

specific for trilateration networks

horizontal distance *e* ***or slope distance*** *s*
: have the same meaning as in the station and target rows.

Standard deviation (a priori) σ_e ***or*** σ_s ***or weight*** p_e ***or*** p_s ***of the height difference***
: are used in the adjustment for the stochastic model. Only one kind of accuracy measure can be present. A default value pads all missing values in this column, or all values. If accuracies are entirely missing, all weights are set to unity, while a warning is issued.

Units

All angular quantities *o, r, t, v* are expected in the specified angle unit.

All other quantites *e, s, dh, l, ih, th* are always expected in the natural length unit , i.e. in case of a grid system the grid scale must not be applied beforehand.
ExceptionLengths of levelling lines *l* may have a different unit of length, i.e. it may be kilometres, while the other quantities are specified in metres. Moreover, a grid scale can be neglected here.

Measurement lists with distances etc. in grid scale

In Measurement lists the unit for metric quantities *e, s, dh, l, ih, th* must always be the natural length unit .

If for the grid system in a measurement list the grid scale is always applied nonetheless, please temporarily change the system type to cartesian (XYZ or YXZ) lefthanded and perform the computations. Now everything is computed in the grid scale. Afterwards reset the system type back to grid system (northing easting height or easting northing height).

By the way, the same trick works for translation parameters as well as for edge lengths and lattice spacings for Create lattice points.

Missing measurements

Measurement values can be missing in any way, in the Universal computer all measurements can be missing and only pointnames are given. Blind targets. If a value in missing inside the row, the ";;" notation can be used, exemplary in Zylinder durch sieben Punkte.

⚠ Missing measurements are always treated as unknown, also instrument and target heights! Here and there a default value may be given.

Did you know? *You may switch between german and english language at all times.*

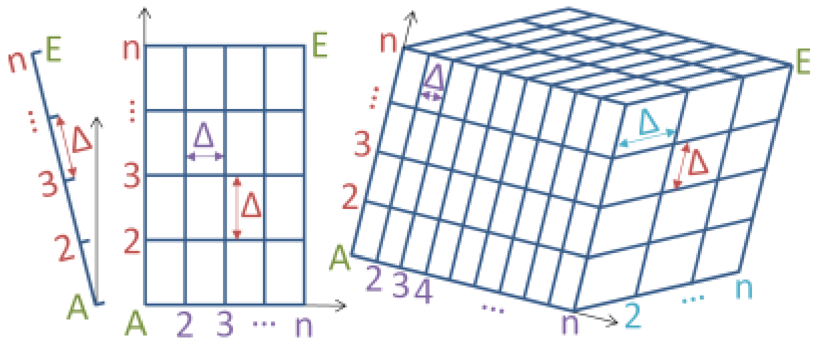
IN DUBIO PRO GEO Guide : Create lattice points

Page contents

Equidistant points on a straight line (1D), a rectangular lattice (2D) or cuboidal lattice (3D) are created and may be postprocessed (e.g. rotated) with other computation tools.

Introduction

By "lattice" we denote the set of regularly arranged points. We avoid the term "grid" here because it is used in geodesy for a grid system.



1D lattice (left), 2D lattice (centre) and 3D lattice (right)

The lattice is defined by a start point A and an end point E. The first created lattice point is the point A and the last one is the point E. If A and E are swapped, you get the same lattice points in opposite succession.

Lattice	Property	planar or spatial
line (1D) ⇓ Read more	On the connecting line AE, equidistant intermediate points are created. The line may be located arbitrarily oblique in space.	A and E must both be 2D or both be 3D points, and so will be created the intermediate points.
rectangle (2D) ⇓ Read more	In the axis parallel horizontal rectangle spanned by A and E, equidistant intermediate points are created in both axis directions.	If both A and E are 2D points, so will be created the intermediate points. If A or E has three coordinates, the third coordinates is transferred to all intermediate points. If both A and E are 3D points, the third coordinates must be identical.
cuboid (3D) ⇓ Read more	In the axis parallel cuboid spanned by A and E, equidistant intermediate points are created in all three axis directions.	A and E must both be 3D points, and so will be created the intermediate points.

⚠ Only for cartesian coordinates (XYZ oder YXZ) the lattice points are always located on spatial straight lines and are equidistant in the sense of Euklid. Otherwise this property is only fulfilled after mapping to the coordinate plane. E.g. for ellipsoidal coordinates (latitude, longitude) they are located on meridians and parallels or on the loxodrome. Exact equidistancy is here obtained only in the equirectangular projection. See ⇑ type of coordinate system and ☆Loxodrome from Dresden (Saxony) to Dresden (Ontario).

Input quantities

For each line direction (1D lattice) or axis direction (2D or 3D lattice) three input quantities are desired in arbitrary succession. The following five quantities are selectable:

- coordinate of A = first lattice border
- coordinate of E = second lattice border
- point count n in line or axis direction, positive integer
- lattice spacing Δ in line or axis direction, positive or negative
- edge length $l=(n-1)\times\Delta$, positive, negative or zero

A given edge length must always be combined with exactly one coordinate of A or E. For a negative edge length the lattice points run contrary to the related axis direction, i.e. they start with the largest coordinate.

Point count and lattice spacing should not be zero. Otherwise they are treated as a missing input.

Example: The lattice with the coordinates $X=11;21;31;41$ can be defined in the following ways:

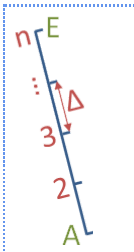
$XA=11;\ XE=41;\ n=4$	$XA=11;\ n=4;\ l=30$	$XA=41;\ XE=11;\ n=4$	$XA=41;\ n=4;\ l=-30$
$XA=11;\ XE=41;\ \Delta=10$	$XE=41;\ n=4;\ l=30$	$XA=41;\ XE=11;\ \Delta=-10$	$XE=11;\ n=4;\ l=-30$
$XA=11;\ n=4;\ \Delta=10$	$XA=11;\ \Delta=10;\ l=30$	$XA=41;\ n=4;\ \Delta=-10$	$XA=41;\ \Delta=-10;\ l=-30$
$XE=41;\ n=4;\ \Delta=10$	$XE=41;\ \Delta=10;\ l=30$	$XE=11;\ n=4;\ \Delta=-10$	$XE=11;\ \Delta=-10;\ l=-30$
Result: $X=11;21;31;41$		Result: $X=41;31;21;11$	

If the edge length is not an integer multiple of the lattice spacing, the lattice spacing is adapted. So you get the exemplary lattice above also with $XA=11;\ XE=41;\ \Delta=9$ or $XE=41;\ \Delta=11;\ l=30$.

Lattice spacing and edge length in grid scale

⚠ Also for the grid system, lattice spacing and edge length are in metric scale, i.e. they are **not** multiplied with the grid scale factor . If this is undesired, please declare the system temporarily as a cartesian system XYZ or YXZ. See also Measurement lists with distances etc. in grid scale .

1D lattice



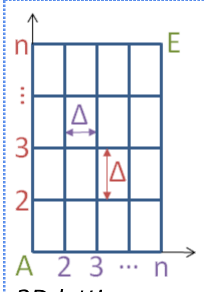
1D lattice

The lattice points are **equidistant points on the straight line segment AE**, which may be oblique in space. Here the input quantities must always be coordinates of A and B as well as either the point count n or the lattice spacing Δ . The first two columns of the input values must be filled in completely, where the second point count is set to $n=1$ or the lattice spacing is set to $\Delta=0$.

- If the last column is empty, all lattice points get only two coordinates (2D points).
- If the last column contains only one coordinate, it is used for all lattice points (3D points).
- If the last column contains two coordinates, all lattice points are computed on the oblique line AE (3D points).

The lattice spacing is the spatial separation of the neighboring points. For ellipsoidal systems, the height, if present, is not used for computation of the separation. Siehe ☆Loxodrome from Dresden (Saxony) to Dresden (Ontario).

2D lattice



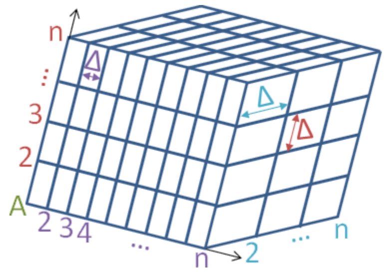
2D lattice

The lattice points are located on the crossings of equidistant coordinate lines, and as such form an **axis parallel rectangular lattice**. The first two columns of the input values must be filled in completely.

- If the last column is empty, all lattice points get only two coordinates (2D points).
- If the last column contains only one coordinate, it is used for all lattice point (3D points).
- If the last column contains two coordinates, they must be equal. The result conincides with the previous one.

After creation the lattice can be ⇓rotated . For different transformations you can transfer the lattice points to Transf. by parameters . See ☆2D lattice for the Großer Garten Dresden.

3D lattice



3D lattice

The lattice points are located on the crossings of equidistant coordinate planes, and as such form an **axis parallel cuboidal lattice**. All nine input values must be given completely. On the canvas only one lattice plane is given (third coordinate fixed).

You can also create a vertical 2D lattice, namely as a 3D lattice with point count $n=1$ for the first or second coordinate. However, the two corresponding coordinates of A and E must coincide. Moreover, you can also create a vertical 1D lattice by letting the first and second point count be equal to 1 and letting the points A and E differ only in the third coordinate. However, the same lattice is obtained more directly as a true 1D lattice.

After creation the lattice can be ⇓rotated about a vertical axis. For different transformations you can transfer the lattice points to Transf. by parameters .

Lattice point list

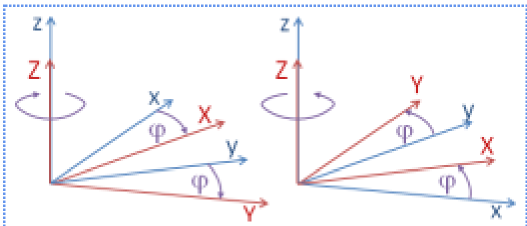
The first created lattice point is the point A and the last one is the point E. If A and E are swapped, you get the same lattice points in opposite succession.

The created lattice points are collected to a coordinate list . You define a system name and a type of system . The column format is always "coordinates", such that the created lattice points are automatically named . Lattice point lists can be saved and processed by other computation tools, e.g. they may be rotated.

Rotation of lattice points

This rotation is not possible for ellipsoidal coordinates (latitude, longitude). Otherwise, you can choose, if lattice points should be rotated **about a vertical axis** (parallel to z axis), and about which point. The following are selectable:

- vertex A = first point of coordinate list
- vertex E = last point of coordinate list
- the barycentre S of all lattice points
- the origin O of the system



spatial lefthanded (left) and righthanded (right) system, rotation about z axis

After creation of the axis parallel lattice or 1D lattice you specify the angle of rotation between `-π = -180° = -200 gon` and `π = 180° = 200 gon`. Before you should have selected the angle unit in the Settings . Rotations are defined

- for lefthanded systems positive clockwise,
- for righthanded systems positive counterclockwise.

The viewing direction for this is opposite to the rotation axis, i.e. from above.

For different transformations, e.g. rotations about other axes or shear mappings, you can transfer the lattice points to Transf. by parameters and configure arbitrary transformation steps and parameters.

2D lattice for the Großer Garten Dresden

The Großer Garten Dresden has a nearly rectangular shape with edge lengths `1900 m` and `950 m`. The northernmost vertex has the UTM coordinates (zone 33U) easting = `412734 m` and northing = `5655664 m`. The azimuth of the short edge is `35 gon ≈ 31°`. A lattice with the lattice spacing `190 m` must be computed.

2D lattice for the Großer Garten Dresden

Firstly, for the type of grid system `Easting Northing Height` you create an axis parallel 2D lattice on a rectangle with edge lengths `1900 m; 950 m` and the lattice spacings `190 m` in both axis directions. For the lattice to end at the northermost point E, you may give the first edge length and the first lattice spacing negative, such that the first coordinate (east) runs opposite to the axis direction. After creation you obtain `11×6=66` lattice points. Finally, you must rotate the lattice by `31°` about edge point E.

Load example and click "Create" and then rotate by clicking "go"

Loxodrome from Dresden (Saxony) to Dresden (Ontario)

We consider the following points in ellipsoidal coordinates referring to World Geodetic System 1984:

	ellipsoidal		
point	**latitude**	**longitude**	**height**
Dresden (Saxony), centre point of the central building of the HTW (Hochschule für Technik und Wirtschaft Dresden, University of Applied Sciences)	51.037512°	13.735186°	120 m
Dresden (Ontario), St. Andrews Presbyterian Church	42.590278°	-82.181667°	183 m

The loxodrome connecting these points should be realised by intermediate points. For the distance of the intermediate points we choose 1°.

Load example and click "Create"

Exercise: See for yourself that the created points are not equidistant on the ellipsoid, by transferring the 97 points to Ellipsoidal polygons and computing as an open polygon. The side lengths vary between `70826 m` and `82532 m`. Moreover, you see that the loxodrome is not the shortest path between the points on the ellipsoid, because all polygonal angles are smaller than `π = 180° = 200 gon`. The smallest polygonal angle is the first, which amounts to only `199.12 gon = 179.21°`.

Exercise: Instead of this we have created equidistant intermediate points on a straight line in equirectangular projection. To prove this, transfer the created points to a coordinate list and change the system type to XYZ or YXZ before saving. Now transfer the saved list to Planar polygons and compute as an open polygon. The heights are now ignored. The side lengths are now all equal to `1.0030010552°`. The difference to the desired value $\Delta=1°$ is due to the fact that the specified point spacing was not an integer multiple of the point distance, such that the spacing had to be slightly adapted. All planar polygonal angles are now exactly equal to `π = 180° = 200 gon`.

Exercise: Now, also load the list to Spatial polygons to see that the points are equidistant also in 3D space. The distance of `1.1986138574` is now, however, a strange mixture of degree and metre. All polygonal angles are again exactly equal to `π = 180° = 200 gon`.

Did you know? *The tools for ellipsoids of rotation also operate on the sphere.*

IN DUBIO PRO GEO Guide : Planar polygons

Page contents

Planar polygons are computed from given coordinates of vertices: planar polygonal angles, azimuths and lengths of sides, area, perimeter, barycentres, etc.

Introduction

A planar polygon is a planar curve consisting of straight line segments (sides). It is defined by a sequence of vertices in a plane joined by sides. A polygon may be **open or closed** . In the second case the last and the first vertex are connected by an side, such that the polygon bounds a planar piece of surface. The vertices are given as a coordinate list , which also defines the succession of points.

Cartesian systems (XY or YX): A Z coordinate is ignored, if any, while a warning is issued. This means that the points are projected onto a horizontal plane and are treated as polygonal vertices there.

Grid systems (Northing Easting or Easting Northing): Heights, if present, are used to compute the grid scale . Consequently the polygon is computed in the horizontal grid plane at the mean height of all points with specified heights (or at zero height if all heights are missing). For all lengths and for the area the grid scale is applied. If you desire to compute the polygon at zero height, then all heights must be set to zero or omitted. (In German real estate affairs the sizes of areas refer to zero height.)

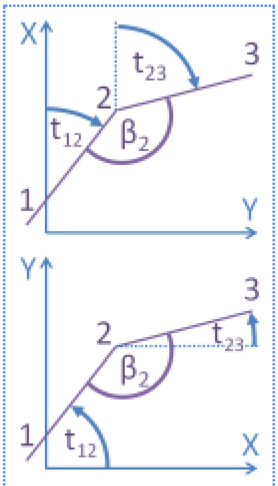
Ellipsoidal systems (Longitude Latitude or Latitude Longitude) are not directly applicable, but require a Coordinate conversion.

See also types of coordinate system.

Results of the computation

Polygonal angles are always located right of the polygon and may for the closed polygon be internal angles (moving clockwise) or external angles (moving counterclockwise).

Azimuthal angles, also known as azimuths, are angles from the X or northing axis to the polygonal sides in moving direction. The sense of rotation is counterclockwise for the cartesian righthanded system and clockwise for all other types of system. Therefore, the X or northing axis has azimuth zero and the Y or easting axis has azimuth `90° = 100 gon`.

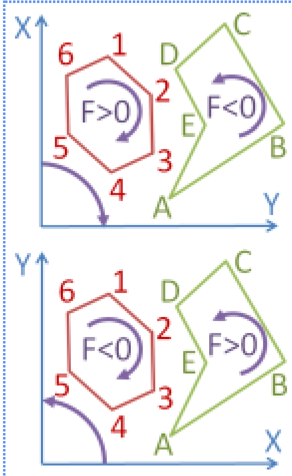


Azimuths t and polygonal angles β. top: lefthanded system, bottom: righthanded system

Special points, i.e. barycentres and circle centres are computed only is possible. The barycentre of area is the centre of mass of the area of the closed polygon. The barycentre of sides is the centre of mass of the sides of the polygon. The barycentre of vertices is the centre of mass of the vertices of the polygon. The centres and radius of the circumscribed and inscribed circles are only computed for closed triangles. (For other polygons such circles do in general not exist.) All those computed coordinates refer to the system of the given coordinates of the vertices.

⚠ The barycentre of area of the polygon crossing itself can be situated very much outside the polygon and is practically rarely useful.

Further results for closed polygons



Sign rule of the area

The **size of area** refers to the polygon. The following sign rule is established:

size of area F	sequence of vertices clockwise	sequence of vertices counterclockwise	polygon crosses itself
lefthanded	$F > 0$	$F < 0$	difference of the area
righthanded	$F < 0$	$F > 0$	segments is obtained

The **polygonal diameter** equals the maximum distance of all pairs of vertices. Example: The polygonal diameter of a rectangle equals the diagonal.

☆ Surrounding polygon for Großer Garten Dresden

The “Großer Garten Dresden” has been surrounded by a polygon consisting of seven vertices. For those vertices the UTM coordinates (zone 33U) have been determined. The succession has been chosen clockwise, in order to get the area positive and the polygonal angles are internal angles of the polygon.

Aerial image Großer Garten Dresden

The computation of the polygon results in an area of `1811260 m² ≈ 1.8 km²` and a perimeter of `5767 m` . The longest polygonal side is the side 1-2, it has the length `1903 m` . Opposite sides differ in the azimuth by about `π = 180° = 200 gon`, they are therefore nearly parallel. At the sides 1,2,3 the internal angles are nearly right angles. The vertices with the largest distance are 2 and 7, it amounts to `2174 m`.

Heights have been omitted in this example. Therefore, lengths and areas refer to the sea level. The Großer Garten has a mean height of approximately 120 m. We could tentatively add height values 120 to the vertex coordinates. (One would be enough because for the. grid scale is deduced from the mean of all specified heights.) The new size of the area at ground level of `1811328 m²` turns out to be `68 m²` larger than the area at sea level. This height effect seems to be significant. However, if we add 1 m to the northing of side 1 then we get an area at sea level of `1812341 m²`. This increase by far exceeds the previous one.

Due to this change of height the lengths increase by some centimetres. The perimeter increases e.g. by `0.11 m` . Angles and coordinates of barycentre do not change. Since the sides are not defined with metre accuracy here, it is not mandatory to use heights in this computation.

Load example and “Compute” (The Settings for the grid system are thereby adapted automatically.)

We add the midpoints of the two long polygonal sides as new vertices and recompute. At these points the polygonal angles amount to `π = 180° = 200 gon`. The side lengths are cut into halves. The other values do not change. However, if you divide the closed polygon along the line through both midpoints into a western and an eastern partial polygon, by deleting or commenting out (i.e. prepending '//') the remaining points in the coordinate list , and compute those partial polygons separately, then you recognize that in the western part the sides shrink by up to 1 mm and in the eastern part they stretch by up to 1 mm. The sum of the areas of both partial polygons is smaller than the area of the total polygon by `0.023 m²` . The reason is that during all computations a constant grid scale of the polygons has been assumed. When splitting the area, this approximation is enhanced.

Here, this deviation is comparatively small, because Dresden is situated close to the central meridian.

Großer Garten, two partial polygons

Load example and “Compute” (The Settings for the grid system are thereby adapted automatically.)

Circle through three points

Planar polygons and Spatial polygons also compute a circle through three points, this is the case if the closed planar or spatial polygon consists of exactly three points. Below “Special points” you find the centre of the circumscribed circle M_3.

The radius R is found if the polygon is loaded in Planar triangles and computed. By the way, the same applies to the centre of the inscribed circle M_4 and the corresponding radius r.

Did you know? *After 24min of idle time the session is automatically closed by the server. Unsaved data will be lost.*

IN DUBIO PRO GEO Guide : Matrix computations

Page contents

Various computations are performed with a matrix or one of its submatrices: Inversion, Cholesky, LU and eigenvalue decomposition, determinant, norms, etc.

☆ Orthogonal matrix

This example shows an orthogonal matrix of three rows and three columns. The inverse equals the transposed matrix. The singular values are all equal to 1. The matrix is not a rotation matrix, because the determinant is equal to -1, not +1. In addition to the rotation there is a reflexion (mirroring).

```
0.0 -0.8  -0.6
0.8 -0.36 0.48
0.6 0.48 -0.64
```

Load example and "Compute"

☆ Arithmetic expressions in matrices

This example shows a matrix of four rows and three columns, who's elements are all identical to 16.1063. It shows the broad range of arithmetic expressions accepted bei IN DUBIO PRO GEO.

```
161063e-4      8.1+80063e-4      pi*16.1063/pi
1610.63%       161063/10000      log(exp(16.1063))
8.1+8.0063     (3,3009-1)*7,0    sqrt(16.1063^2)
2,3009*7,0     3,3009*7,0-7      asin(sin(0.161063))*100
```

See Arithmetic expressions in input fields.

Load example and "Compute"

Exercise: Create a 4×4 matrix by appending the third column right once with "Select and/or re-order columns". Note that the rank of this matrix is equal to one because in the LU decomposition the matrix U has obviously rank one. This is shown be the fact that this triangular matrix has only *one* main diagonal element essentially different from zero.

Did you know? *The LU decompositon of a square matrix A generates a product of a lower unitriangular matrix L and an upper triangular matrix U. In general row swaps by a permuation matrix P are required, such that PA=LU holds.*

IN DUBIO PRO GEO Guide : Satellite orbits

Page contents

From ephemeris or almanac data of GNSS satellites (e.g. GPS) discrete orbit points are computed at a specified time grid. The computation is based on the GPS Interface Specification IS-GPS-200 and the GALILEO Signal in Space Interface Control Document.

Geodetic constants

As usual in geodesy, the geocentric gravitational constant GM refers to the total mass of the Earth, including the atmosphere.

At the time of introduction of GPS, the geocentric gravitational constant GM was not as exactly known as it is today. At this time the value of $GM = 3986005 \cdot 10^8\ m^3/s^2$ was used, while today it is mostly replaced by the more accurate value of $GM = 3986004.418 \cdot 10^8\ m^3/s^2$, which is defined in the World Geodetic System 1984 . This new value is also used for GALILEO. To allow older GPS receivers to compute the orbits correctly nonetheless, the broadcast ephemeris data of GPS satellites are still adapted to the old value.

The angular velocity of Earth's rotation must refer to the fixed stars. In the World Geodetic System 1984 the value of $\omega_E = 7.2921151467e\text{-}5\ rad/s$ is defined and should routinely be used.

Algorithm of orbit computation

The algorithm is described in the official document GPS Interface Specification IS-GPS-200 in table 30 and in the official GALILEO Signal in Space Interface Control Document in table 58. The symbols used there differ slightly from ours.

Keplerian elements at reference time

By the following six orbital elements an unperturbed Keplerian orbit is defined, i.e. a spatial ellipse fixed with respect to the stars and the current position (known as ''anomaly'') within this ellipse.

- a semi-major axis of orbital ellipse
- e numerical eccentricity of orbital ellipse
- i orbital inclination angle
- ω argument of perigee
- Ω ascension of ascending node
- M mean anomaly

There are different options for specification of the size of the ellipse and the anomaly. The official GPS definition uses die square root of the semi-mayor axis $\sqrt{a}$ and the mean anomaly M Input options of alternative orbit parameters are offered as well:

instead of semi-major axis of orbital ellipse

T_0 revolution period

n_0 mean motion

instead of mean anomaly

E_0 eccentric anomaly

v_0 true anomaly

The Keplerian elements refer to the time instance t_{oe}, which is given in seconds after the beginning of the GNSS week, ⚠ but not the right ascension of the ascending node Ω_o which refers to the beginning of the GNSS week $t=0$.

Change rates, correction values and correction terms

If an unperturbed Keplerian orbit is desired, only these six Keplerian are needed. Change rates, correction values and correction terms of these elements are needed to model satellite orbit perturbations.

For mean precisions, e.g. for the computation of satellite's rise and set times, at least the rate of right ascension $d\Omega/dt$ is required, except if the orbit should be computed at the beginning of the week. For GPS the reference value `-2.6 semi-circles/s = -8.168e-9 rad/s` is used, which specifies the approximate nodal precession caused by the Earth's flattening. For GALILEO the corresponding value is `-0.02764398°/d = -5.584e-9 rad/s`.

For high precisions, e.g. for the computation of the receiver position from observations, also the other change rates, correction values and correction terms are required. They are provided as broadcast ephemeris data in the navigation message or as precise ephemeris data by an orbital service.

Computation of the orbit points

We compute the positions of the satellite reference point in the earth fixed righthanded cartesian coordinate system at the desired GNSS system time instances. Those time instances are defined by three parameters:

- start time of orbit computation t_1 (=1st orbit point) in seconds referring to the beginning of the current GNSS week
- time increment of orbit computation Δt in seconds
- number of orbit points to be computed

If only *one* should be computed, the last two parameters can be missing.

The beginning of a GNSS week $t=0$ is always sunday 0:00:00 in the GNSS system time. Note that this time differs from the universal time coordinated (UTC) by leap seconds.

If j is the counter of the points, the time instances with respect to t_{oe} as origin of the time axis are generated according to the following scheme:

$$\delta t_j = t_1 + (j\text{-}1)\cdot\Delta t - t_{oe}$$

The following column formats determine the appearance of the Coordinate lists:

- **pointname coordinates**: The times since beginning of the week are used as point names.
- **pointname code coordinates**: The points are automatically named. The times since beginning of the week are used as point codes.
- **coordinates**: The points are automatically named. The times are not displayed.

☆ Orbit computation from a GPS almanach

Given is the following GPS almanach in YUMA format:

```
******** Week  297 almanac for PRN-02 ********
ID:                          02
Health:                      000
```

```
Eccentricity:                0.9529113770E-002
Time of Applicability(s):    589824.0000
Orbital Inclination(rad):    0.9551331376
Rate of Right Ascen(r/s):   -0.8183198006E-008
SQRT(A)   (m 1/2):           5153.635742
Right Ascen at Week(rad):    0.1038484770E+001
Argument of Perigee(rad):    1.827911506
Mean Anom(rad):              0.2496773193E+001
Af0(s):                     -0.2574920654E-004
Af1(s/s):                    0.0000000000E+000
week:                        297
```

We desire the positions of satellite 02 at the full hours of the last three days of the GPS week `297`.

Load example and "Compute"

Satellite orbit in the starfixed system

Satellite orbits usually computes discrete orbit points in the earth fixed (rotating) righthanded cartesian coordinate system ECEF. However, if you want to obtain the orbit points in the star fixed (quasi inertial) system ECSF, simply specify for the angular velocity of Earth's rotation $\omega_E = 0$. Then you obtain a system with axes coincident with the earth fixed system at the beginning of the week and then kept star fixed.

Satellite orbit velocitiy

Satellite orbits are computed in the form of Coordinate lists of discrete orbit points on a time grid. If you want to obtain the orbit velocity, transfer the coordinate list to Spatial polygons and compute it as an open polygon. Then you obtain the spatial distances of consecutive orbit points as side lengths of the polygon. If you chose e.g. 1 second as a computation time increment Δt then the side lengths are immediately velocities in metres/second.

Usually you obtain the velocities in the earth-fixed (rotating) coordinate system. If you want to obtain them in the starfixed system, please apply the previous trick. If you do this for the ☆ Orbit computation from a GPS almanach (time increment 1h), then you obtain orbit velocities between `13600 km/h` and `14000 km/h`.

Did you know? *Planar polygons and Spatial polygons also compute a Circle through three points.*

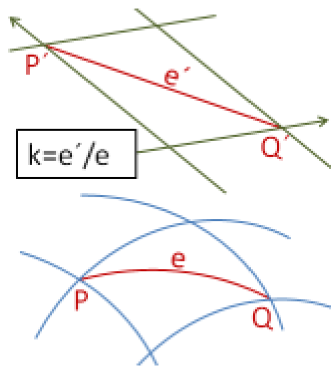
IN DUBIO PRO GEO Guide : Grid scale factors

Page contents

The point scale factor of grid systems at the points of Coordinate lists and the line scale factor along the lines between consecutive points of such lists are computed.

Introduction

Grid systems (type of system: Northing Easting Height or Easting Northing Height) are based on a local grading of the curved ellipsoid of rotation by Gaussian mapping (= Transversale Mercator projection on the ellipsoid). Due to the inevitable deformations the unit of grid coordinates does not coincide with the metric length unit . This is corrected by a **grid scale factor** k . It holds $k > 1$, if grid distances (=map distances) e' are longer than ground distances (=natural distances) e , and $k < 1$ in the opposite case.



grid scale factor k , ground distance e , grid distance e'

⚠ The correct computation and consideration of k requires that the Settings for the coordinate system parameters are correct. Be careful when using **abbreviated** coordinates. The settings false easting and false northing must be changed by the same amount! This is explained in the ☆ GPS reference point of HTW Dresden.

Approximate correction in small point areas

As a result of the conformity of the Gaussian mapping, in a small point area the **grid scale factor** k is nearly constant and can be computed automatically as follows:

$$k=k_o\cdot\frac{(\bar{E}-E_o)^2 - 2\bar{h}R}{2R^2}$$

$\bar{E}$ = mean easting at point site

$\bar{h}$ = mean ellips. height at point site

E_o = false easting , initially it is set to `500000`.

k_o = scale factor at the central meridian , initially it is set to `0.9996`.

R = radius of Gaussian mean curvature

The mean height $\bar{h}$ is only computed from points with specified heights. If no heights are specified at all then the mean height is assumed to be 0.000, while a warning is issued. In the plains the influence of the height is small. It amounts up to 15 pmm for a height up to 100 m.

This correction is used by IN DUBIO PRO GEO for all computations with grid coordinates, except for Grid scale factors. This computation tool serves for the much more accurate determination of the correction. This will be explained in the following sections.

In the informations to a grid coordinate list the computed scale factor is reported. If the point area is large, the extreme values of the scale factor are reported instead. This serves as a warning that the computation may be inacurate, because only *one* scale factor per grid coordinate list, i.e. the mean value.

If the points are distributed over a large area then the grid scale factor is not representative in the entire point area. This happens typically for a great expansion of the area in east-west direction or in vertical direction. In such a case you should filter the list, if possible, such that points outside the area of interest are truncated.

In the Settings it can be specified, how many decimal places of k must be matching within the point area. Default is 3 decimal places. In this case, e.g. $min(k)=0.999267,\ max(k)=0.999783$ would enable a computation, otherwise an error occurs. Initially, 3 decimal places are in effect.

Other metric values and scales

⚠ When doing computations with grid coordinates, it is supposed that all other metric values and scales are **not** distorted by the grid scale. This concerns the following quantities:

Computation tool	not distorted by the grid scale
Create lattice points	lattice spacings and edge lengths
Planar polygons Spatial polygons	side lengths, perimeters, areas, radii of circles
Transf. by parameters Transf. by control points	translation parameters, transformation scales
Adjusting surfaces	radii of adjusting sphere, semiaxes of adjusting ellipsoid and hyperboloid etc.
Universal computer Traverses Trilateration	horizontal and slope distances

If you want to specify those data in grid scale nonetheless, apply the following trick:
Measurement lists with distances etc. in grid scale .

Point scale factors

The computation tool Grid scale factors serves for the **much more accurate determination** of the scale factors using a sufficiently long series expansion. However, the accuracy is reached only in a small vicinity of a point. Given a grid or ellipsoidal coordinate list, **for each point** an associated scale factor is computed. This factor relates to horizontal distances in the height of the point.

For the given coordinate list it is roughly estimated, how large the worst case error of the series expansion can be.

Line scale factors

Between all consecutive points of the coordinate list the ratio of the length of the geodesic and it's image in the grid space is computed. For this purpose a sufficiently accurate quadrature formula is used. Each obtained line scale factor refers to horizontal distances in the mean height of the corresponding line.

For the given coordinate list it is roughly estimated, how large the worst case error of the quadrature formula can be.

☆ Points on the 51° parallel

The UTM grid scale factor for equispaced points on the 51° parallel is computed. The point spacing is 0.2° ≈ 14 km. The point scale factor varies between 0.9996 and 1.000144. The line scale factor varies between 0.9996 and 1.000109.

⚠ If you load the example, the ⚙Settings for the grid system are reset to standard values.

Load example and “Compute”

__Did you know?__ On the start page you search IN DUBIO PRO GEO by the Google search engine.

IN DUBIO PRO GEO Guide : Normal gravity formulae

Page contents

The normal gravity at a point of given ellipsoidal latitude and height for the level ellipsoids GRS67,GRS80 and World Geodetic System 1984 is computed, optionally including an Error propagation.

Introduction

The normal gravity field is a coarse approximation of the true gravity field of the Earth. It serves as a simple easy-to-compute model. In geodesy and geophysics we use rotationally symmetric normal gravity fields, where a surface of equal gravity potential (known as equipotential or level surface) coincides with a geodetic reference ellipsoid, e.g. GRS80 or World Geodetic System 1984

The value of the gravity acceleration in the normal gravity field is called **normal gravity** γ . It depends on the latitude φ and the ellipsoidal height h above the level ellipsoid. It decreases from the poles to the equator by about `0.052 m/s²` and in vertical direction by about `0.003 m/s²` per kilometre of altitude. The following normal gravity formulae are implemented:

International normal gravity formula 1967

$$\gamma_o(\varphi) = \gamma_e \cdot (1+5.2891 \cdot 10^{-3} \cdot \sin(\varphi)^2 - 5.9 \cdot 10^{-6} \cdot \sin(2\varphi)^2)$$

$$\gamma(\varphi, h) = \gamma_o(\varphi) \cdot (1-(3.15704 \cdot 10^{-7} - 2.10269 \cdot 10^{-9} \cdot \sin(\varphi)^2) \cdot h + 7.37452 \cdot 10^{-14} \cdot h^2)$$

with γ_e = 9.780318 m/s² and h in the unit metre.

Formula of Somigliana for the normal gravity at the level ellipsoid h=0

$$\gamma_o(\varphi) = \gamma_e \cdot (1+k \cdot \sin(\varphi)^2) (1-e^2 \cdot \sin(\varphi)^2)^{-½}$$

with values for

GRS80: γ_e = 9.7803267715 m/s², k =0.0019318513548, e^2 =0.00669438002290
WGS84: γ_e = 9.7803253359 m/s², k =0.0019318526464, e^2 =0.00669437999014

Height dependence for GRS80 and WGS84

$$\gamma(\varphi, h) = \gamma_o(\varphi) \cdot (1-2(1+f+m-2f \cdot \sin(\varphi)^2) \cdot (h/a)+3(h/a)^2$$

with the semi-major axis of the level ellipsoid a =6378137 m and the values for

GRS80: f =1/298.257222101, m =0.00344978600308
WGS84: f =1/298.257223563, m =0.00344978650684

☰ ☆ Gravity benchmark at the geodetic laboratory of HTW Dresden

We compute the normal gravity for the gravity benchmark at the geodetic laboratory of the HTW HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES. The ellipsoidal latitude equals `51.03361°`. The height is `114 m` above the reference surface DHHN92. The height of the reference surface DHHN92 above the ellipsoid WGS84 equals `35 m`. This results in a ellipsoidal height above WGS84 of about `149 m`. We compute a value of $\gamma(\varphi, h) = 9.811161\ m/s^2$. By the way, an absolute gravity value measured at the geodetic laboratory is obtained as `9.811193 m/s²` (rounded).

We further compute the vertical gradient of the normal gravity at this benchmark. For this purpose we use the Error propagation utility with a height deviation of exactly 1 m. The deviation of gravity at a height deviation of 1 m equals the value of the vertical gradient (sign is always minus). The latitude is held fixed. In this case it is arbitrary if you specify the height deviation as a maximum or standard deviation. The value of $3.085 \cdot 10^{-6} s^{-2}$ is obtained.

Load example and “Compute”

Did you know? *In geodesy an alternative unit of gravity is used: 1 Gal = 0.01 m/s² ; 1 m/s² = 100 Gal*

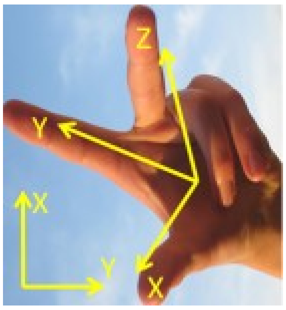
IN DUBIO PRO GEO Guide : Transformation by parameters

Page contents

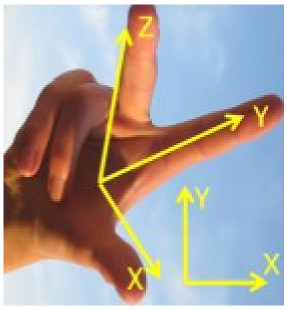
Points in the plane and in 3D space are transformed by given parameters. A sequence of up to 13 individual transformation steps can be performed. In this way, all conceivable transformations can be configured.

Introduction

IN DUBIO PRO GEO computes planar and spatial coordinate transformations, i.e. conversions of point coordinates of an **inital system** to coordinates of a **target system**. We do not require any special approximate alignment of the axes of the systems. Coordinates are specified via Coordinate lists .



lefthanded system



righthanded system

Cartesian lefthanded, cartesian righthanded systems `(XYZ)` and grid systems `(northing,easting,height)` (or other succession of coordinates) can by transformed immediately. Ellipsoidal systems `(latitude,longitude,height)` must first be converted with Coordinate conversion . See coordinate system type.

The coordinates in the initial system are always denoted by x,y,z and in the target system by X,Y,Z . For grid coordinates we identify north = x and/or X, east = y and/or Y and height = z and/or Z.

Transformation steps

Each transformation can be imagined as a sequence of elementary transformation steps. By arbitrary combination of the steps

- **Translation** = shifting
- **Scaling** = scale change, for grid systems not including the grid scale factor
- **Rotation**
- **Transvection** = shear mapping
- **Reflexion** = mirroring from lefthanded to righthanded system or vice versa

using the parameters

shift and rotation parameters		**scaling and shear parameters**	
t_x	translation along x axis	m_x	scale factor of x axis
t_y	translation along y axis	m_y	scale factor of y axis
t_z	translation along z axis	m_z	scale factor of z axis

		m_{xy}	scale factor of x and y axis
ε_x	Eulerian angle for rotation about x axis	m	scale factor of all axes
ε_y	Eulerian angle for rotation about x axis	f_{xy}	shear factor for y vs. x axis
ε_z	Eulerian angle for rotation about z axis	f_{yx}	shear factor for x vs. y axis
ε	2D rotation angle (identical to ε_z)	f_{xz}	shear factor for z vs. x axis
	rotation angle about Eulerian axis	f_{zx}	shear factor for x vs. z axis
e_x	Eulerian axis, x vector component	f_{yz}	shear factor for z vs. y axis
e_y	Eulerian axis, y vector component	f_{zy}	shear factor for y vs. z axis
e_z	Eulerian axis, z vector component	τ_{xy}	shear angle for y vs. x axis
		τ_{yx}	shear angle for x vs. y axis
q_0	quaternion, 0th component	τ_{xz}	shear angle for z vs. x axis
q_1	quaternion, 1st component	τ_{zx}	shear angle for x vs. z axis
q_2	quaternion, 2nd component	τ_{yz}	shear angle for z vs. y axis
q_3	quaternion, 3rd component	τ_{zy}	shear angle for y vs. z axis

all relevant transformation types can be represented. In total up to 13 elementary steps can be arbitrarily combined. They are processed in the given succession. E.g., it is possible to perform a translation and a rotation followed by a further translation. The reflexion is automatically invoked, if the initial system is a lefthanded system and the target system is a righthanded system or vice versa. ⚠ In many cases the transformation result depends on the succession of the transformation steps.

The transformation steps involving the z coordinate require, that all points have three coordinates.

Translation = shifting

For planar transformations two translation parameters t_x , t_y , for spatial transformations three translation parameters t_x , t_y , t_z can be given. They are expected in the natural length unit , i.e. for grid systems not in grid scale. However, if this is desired differently, then grid systems must be redeclared to cartesian systems, see Bag of tricks. Missing translation parameters are treated as zero.

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} x \\ y \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} + \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Scaling

Scale parameters are scale factors applied to the coordinates. For planar transformations at most two scale parameters m_x , m_y , for spatial transformations at most three scale parameters m_x , m_y , m_z are accepted. All scale parameters must be positive. They do not include the grid scale factors . However, if this is desired differently, then grid systems must be redeclared to cartesian systems, see Bag of tricks. Missing scale parameters are treated as one.

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} m_x \cdot x \\ m_y \cdot y \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \end{pmatrix} = m \cdot \begin{pmatrix} x \\ y \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} m_x \cdot x \\ m_y \cdot y \\ m_z \cdot z \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} m_{xy} \cdot x \\ m_{xy} \cdot y \\ m_z \cdot z \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = m \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Rotation

The transformation equation of the **planar** rotation reads:

$$\begin{pmatrix}X\\Y\end{pmatrix} = \begin{pmatrix}cos(\varepsilon) & -sin(\varepsilon)\\ sin(\varepsilon) & cos(\varepsilon)\end{pmatrix} \cdot \begin{pmatrix}x\\y\end{pmatrix}$$

For all **spatial** transformations there is a spatial rotation, which is described in the following three alternative ways:

(a) with **Eulerian angle** ε_x or ε_y or ε_z :

$$\begin{pmatrix}X\\Y\\Z\end{pmatrix} = \begin{pmatrix}1 & 0 & 0\\ 0 & cos\varepsilon_x & -sin\varepsilon_x\\ 0 & sin\varepsilon_x & cos\varepsilon_x\end{pmatrix} \cdot \begin{pmatrix}x\\y\\z\end{pmatrix} \qquad \begin{pmatrix}X\\Y\\Z\end{pmatrix} = \begin{pmatrix}cos\varepsilon_y & 0 & sin\varepsilon_y\\ 0 & 1 & 0\\ -sin\varepsilon_y & 0 & cos\varepsilon_y\end{pmatrix} \cdot \begin{pmatrix}x\\y\\z\end{pmatrix} \qquad \begin{pmatrix}X\\Y\\Z\end{pmatrix} = \begin{pmatrix}cos\varepsilon_z & -sin\varepsilon_z & 0\\ sin\varepsilon_z & cos\varepsilon_z & 0\\ 0 & 0 & 1\end{pmatrix} \cdot \begin{pmatrix}x\\y\\z\end{pmatrix}$$

The total rotation can be viewed as a sequence of three rotations about coordinate axes. Any succession of rotations can be used. Changing the succession of rotations usually changes the result.

(b) with the quadruple of the **unit quaternion** (q_0 , q_1 , q_2 , q_3) . A quaternion usually permits a more elegant description of rotations in three dimensions:

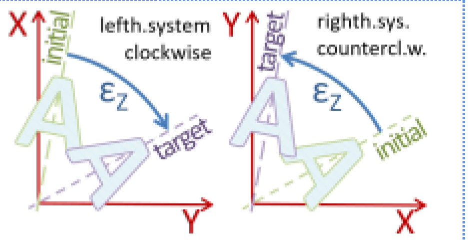
$$\begin{pmatrix}X\\Y\\Z\end{pmatrix} = \begin{pmatrix}q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1 \cdot q_2-q_0 \cdot q_3) & 2(q_1 \cdot q_3+q_0 \cdot q_2)\\ 2(q_1 \cdot q_2+q_0 \cdot q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2 \cdot q_3-q_0 \cdot q_1)\\ 2(q_1 \cdot q_3-q_0 \cdot q_2) & 2(q_2 \cdot q_3+q_0 \cdot q_1) & q_0^2-q_1^2-q_2^2+q_3^2\end{pmatrix} \cdot \begin{pmatrix}x\\y\\z\end{pmatrix}$$

If no unit quaternion with $q_0^2+q_1^2+q_2^2+q_3^2=1$ is given then it is scaled to unity.

(c) with **Eulerian axis** (e_x , e_y , e_z) as a unit vector and **rotation angle** ε about it:

$$\begin{pmatrix}X\\Y\\Z\end{pmatrix} = \begin{pmatrix}cos\varepsilon+e_x^2(1-cos\varepsilon) & e_xe_y(1-cos\varepsilon)-e_zsin\varepsilon & e_xe_z(1-cos\varepsilon)+e_ysin\varepsilon\\ e_xe_y(1-cos\varepsilon)+e_zsin\varepsilon & cos\varepsilon+e_y^2(1-cos\varepsilon) & e_ye_z(1-cos\varepsilon)-e_xsin\varepsilon\\ e_xe_z(1-cos\varepsilon)-e_ysin\varepsilon & e_ye_z(1-cos\varepsilon)+e_xsin\varepsilon & cos\varepsilon+e_z^2(1-cos\varepsilon)\end{pmatrix} \cdot \begin{pmatrix}x\\y\\z\end{pmatrix}$$

The rotation is here described about an oblique axis through the origin of the coordinate system, the so-called Eulerian axis. If no unit vector with $e_x^2+e_y^2+e_z^2=1$ is given then it is scaled to unit length.



Rotation direction for $\varepsilon>0$, viewing direction opposite to the rotation axis

All angles $\varepsilon, \varepsilon_x, \varepsilon_y, \varepsilon_z$ are expected between `-π = -180° = -200 gon` and `π = 180° = 200 gon` and in the chosen angle unit. They are defined

- for lefthanded systems positive clockwise,
- for righthanded systems positive counterclockwise.

The viewing direction for this is opposite to the rotation axis.

⚠ There are alternative conventions in use, mainly in the non-geodetic branches.

Both the parameters q_0, q_1, q_2, q_3 and the parameters $e_x, e_y, e_z, \varepsilon$ form a compound and must be given in the list of transformation parameters as a consecutive sequence, where the internal succession of the group is arbitrary.

☰ Transvection = shear mapping

The shear mapping is an affine mapping of the plane onto itself, which preserves the area of geometric figures, but angles may change. The transformation equations read:

$$\begin{pmatrix}X\\Y\end{pmatrix}=\begin{pmatrix}1 & f_{xy}\\0 & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\end{pmatrix}\quad \begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}1 & f_{xy} & 0\\0 & 1 & 0\\0 & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}\quad \begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}1 & 0 & f_{xz}\\0 & 1 & 0\\0 & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}\quad \begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\f_{yx} & 1 & 0\\0 & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}$$

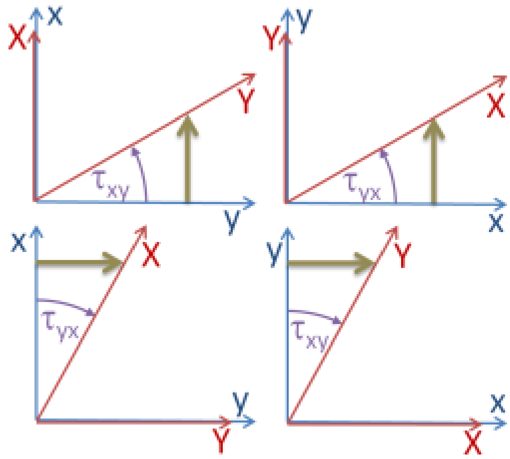
$$\begin{pmatrix}X\\Y\end{pmatrix}=\begin{pmatrix}1 & 0\\f_{yx} & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\end{pmatrix}\quad \begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\0 & 1 & f_{yz}\\0 & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}\quad \begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\0 & 1 & 0\\f_{zx} & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}\quad \begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & f_{zy} & 1\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}$$

$$f_{xy}=tan(\tau_{xy})\quad f_{xz}=tan(\tau_{xz})\quad f_{yz}=tan(\tau_{yz})$$

Either the shear parameters f or the shear angles τ may be specified. All angles are expected between `-π/2 = -90° = -100 gon` and `π/2 = 90° = 100 gon` and in the chosen angle unit. Missing shear parameters are treated as zero.

Clue: Two consecutive shear mappings with $f=f_{yx}=-f_{xy}\neq 0$ are not identical with a rotation with $\tau=arctan(f)$. Angles are not preserved:

$$\begin{pmatrix}X\\Y\end{pmatrix}=\begin{pmatrix}1 & f_{xy}\\f_{yx} & f_{yx}\cdot f_{xy}+1\end{pmatrix}\cdot\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}1 & -f\\f & 1-f^2\end{pmatrix}\cdot\begin{pmatrix}x\\y\end{pmatrix}$$



lefthanded (left) and righthanded (right) system, shear mapping of y vs. x axis (top) and vice versa (bottom)

Planar transformations (2D)

Amongst others, the transformation steps can be combined to the following planar transformations:

Transformation	Parameter	Number
Affine	$t_x , t_y , m_x , m_y , \varepsilon , \tau_{xy}$	6
5-Parameter type 1	$t_x , t_y , m_x , m_y , \varepsilon$	5
5-Parameter type 2	$t_x , t_y , m , \tau , \varepsilon$	5
5-Parameter type 3	$t_x , t_y , \varepsilon , m_x , m_y$	5
5-Parameter type 4	$t_x , t_y , m , \varepsilon , \tau$	5
Helmert	$t_x , t_y , m , \varepsilon$	4
with fixed scale	t_x , t_y , ε	3

Spatial transformations (3D)

Amongst others, the transformation steps can be combined to the following spatial transformations:

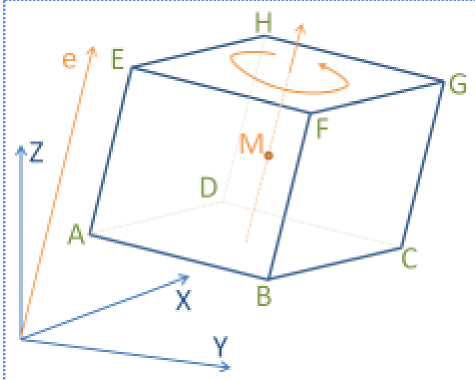
Transformation	Parameter	Number
Affine	$t_x , t_y , t_z , m_x , m_y , m_z, \varepsilon_x , \varepsilon_y , \varepsilon_z , \tau_{xy} , \tau_{xz} , \tau_{yz}$	12
9-Parameter Type 1	$t_x , t_y , t_z , m_x , m_y , m_z , \varepsilon_x , \varepsilon_y , \varepsilon_z$	9
9-Parameter Type 2	$t_x , t_y , t_z , \varepsilon_x , \varepsilon_y , \varepsilon_z , m_x , m_y , m_z$	9
Helmert	$t_x , t_y , t_z , m , \varepsilon_x , \varepsilon_y , \varepsilon_z$	7
with fixed scale	$t_x , t_y , t_z , \varepsilon_x , \varepsilon_y , \varepsilon_z$	6

As an alternative to the Eulerian angles $\varepsilon_x, \varepsilon_y, \varepsilon_z$ also a quaternion (q_0, q_1, q_2, q_3) or Eulerian axis (e_x, e_y, e_z) and rotation angle ε about it are accepted.

☰ ☆ Rotate cuboid about centre axis

The following points with coordinates in a cartesian lefthanded system form the vertices of a cuboid (ABCD=bottom surface, EFGH=top surface):

	X	Y	Z
A	14.034	17.043	8.067
B	23.605	29.759	5.522
C	42.146	16.239	7.807
D	32.585	3.537	10.349
E	14.281	20.222	24.877
F	23.842	32.924	22.335
G	42.393	19.418	24.617
H	32.841	6.711	27.167



Rotate cuboid about centre axis

It must be rotated by 45° about its centre axis parallel to AE (⇑ figure) counterclockwise, as seen from above. The point to be held fixed during rotation may be the centre point M of the cuboid

```
M   28.2159 18.2316 16.3426
```

computed as the mean of all coordinates e.g. with Spatial polygons , or the centre point of the bottom or top surface. The Eulerian axis e may be described by the vector

$$AE=\begin{pmatrix}14.281-14.034\\20.222-17.043\\24.877-8.067\end{pmatrix}$$

Since for lefthanded systems, when looking opposite to the rotation axial direction, the rotation angle is positive clockwise (⇑ Rotation), the rotation angle $\varepsilon=-45°$ must be specified. As an alternative, you could use EA instead of AE as an axis vector. The you should use $\varepsilon = 45°$.

Load example and "Compute"

The result is

$$\begin{pmatrix}X\\Y\\Z\end{pmatrix}=\begin{pmatrix}-2.33842866\\23.6949266\\-4.44667340\end{pmatrix}+\begin{pmatrix}0.70716782 & 0.69550488 & -0.12722668\\-0.69393365 & 0.71721800 & 0.06367434\\0.13553508 & 0.04325843 & 0.98982774\end{pmatrix}\cdot\begin{pmatrix}x\\y\\z\end{pmatrix}$$

PName	X	Y	Z
A	18.4131166747	26.6934690300	6.1776196987
B	34.3492519040	29.0099229729	5.5057885573
C	37.7669114811	6.5924075628	9.6956467711
D	21.8479666784	4.2788643123	10.3664693981
E	18.6601166747	29.8724690300	22.9876196987
F	34.5790614774	32.1860122804	22.3167970717
G	38.0139114811	9.7714075628	26.5056467711
H	22.0968358509	7.4485422141	27.1853915435

Exercise: Perform a Transf. by control points using these coordinate lists. All residuary misclosures must be zero, independent of the weighting.

Did you know? *You must not sign in to IN DUBIO PRO GEO, therefore you work anonymously.*

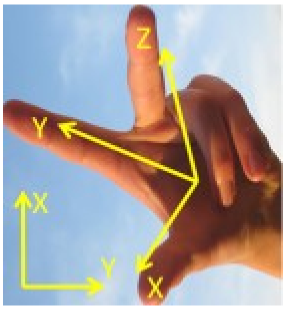
IN DUBIO PRO GEO Guide : Transformation by control points

Page contents

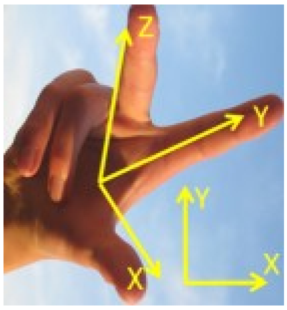
Control points are used for the computation of transformation parameters between two coordinate systems. All planar or spatial transformations are computed, which are computable from these points. In both systems there may be given non-control points, which are transformed by the computed parameters.

Introduction

IN DUBIO PRO GEO computes planar and spatial coordinate transformations, i.e. conversions of point coordinates of an **inital system** to coordinates of a **target system**. We do not require any special approximate alignment of the axes of the systems. Coordinates are specified via Coordinate lists .



lefthanded system



righthanded system

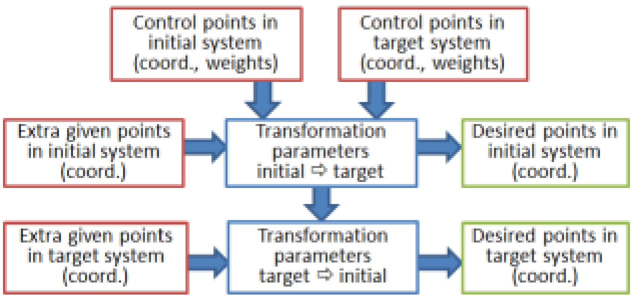
Cartesian lefthanded, cartesian righthanded systems `(XYZ)` and grid systems `(northing,easting,height)` (or other succession of coordinates) can by transformed immediately. Ellipsoidal systems `(latitude,longitude,height)` must first be converted with Coordinate conversion . See coordinate system type.

The coordinates in the initial system are always denoted by x,y,z and in the target system by X,Y,Z . For grid coordinates we identify north = x and/or X, east = y and/or Y and height = z and/or Z.

Control points and new points to be transformed

Control points are identified automatically by identical pointnames. Whenever possible, transformation parameters are computed from those coordinates.

Additionally, there may be points with coordinates given only in one of the systems (new points). The computed transformation is applied to them. There may be new points in both systems at the same time.



Transformation by control points

All points can be specified in the coordinate lists in arbitrary succession.

System of transformation equations

All implemented geodetic coordinate transformations are computed, which fulfill the following three conditions.

- A sufficient number of control points must be provided.
- The control points are configured, such that a unique transformation solution exists, i.e. a regular system of equations is obtained.
- If the parameters need to be computed iteratively: The control points match at least approximately (small residuary misclosures).

All transformations use the following **template of a system of transformation equations** :

$V = t + T \cdot v$	v , V are the position vectors of points in the initial and target system. t is the translation vector (= shift vector). T is the transformation matrix.

⚠The succession of the components of v, V, t, T is always `xyz`, even if the coordinates in the Coordinate lists have a different succession. Depending on the type of transformation, T effects

- **Reflexion** = mirroring from lefthanded to righthanded system or vice versa
- **Rotation**
- **Transvection** = shear mapping
- **Scaling** = scale change, for grid systems not including the grid scale factor

For each computed transformation all transformation equations are given in the above form, such that all additional computations can be performed, e.g. the transformation of additional points. Often it is appropriate to specify the transformation by transformation parameters. Unfortunately, there is a multitude of possibilities, how to do this, which are supported here almost completely:

It is easiest to represent T as a product of at most three matrices:

- **orthogonal matrix** Q , effects a rotation (rotation matrix, if $detQ = +1$) and possibly an additional reflexion (if $detQ = -1$)
- **upper unitriangular matrix** S (this is a triangular matrix with all diagonal elements equal to unity), effects a shear mapping
- **diagonal matrix** M , effects a scale change, possibly all scales are equal, such that only one scalar factor m is in effect, i.e. $M = m \cdot I$ with identity matrix I .

The different types of transformation differ with respect to the set of factors, which compose T , and their succession. For the affine transformation, T can be decomposed into all three factors in arbitrary succession:

$$T = Q_1 M_1 S_1 = Q_2 S_2 M_2 = M_3 S_3 Q_3 = S_4 M_4 Q_4$$

The factors themselves depend in general on the succession of the decomposition, e.g. you get $S_1 \neq S_2 \neq S_3 \neq S_4$. All above-listed decompositions are computed.

Planar transformations (2D)

Z coordinates or heights are not influenced by planar transformation, they may therefore be missing. If heights are given for some or all points in grid systems then they are used to determine the grid scale factor .

Whenever possible, the following transformations are computed one after the other:

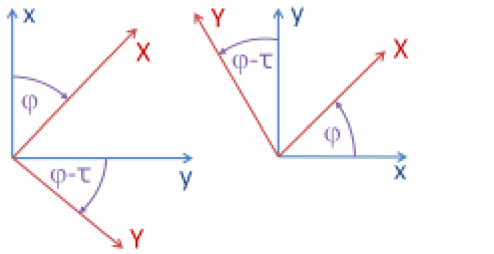
Transformation	Parameter	Number	Contr.P.	Transformation matrix T	
Affine	$t_x, t_y, m_x, m_y, \varepsilon, \tau$	6	≥ 3	is a general `2×2` matrix	
5-Param. type 1	$t_x, t_y, m_x, m_y, \varepsilon$	5	≥ 3	has orthogonal row vectors	$T=MQ$
5-Param. type 2	$t_x, t_y, m, \tau, \varepsilon$	5	≥ 3		$T=mSQ$
5-Param. type 3	$t_x, t_y, \varepsilon, m_x, m_y$	5	≥ 3	has orthogonal column vectors	$T=QM$
5-Param. type 4	$t_x, t_y, m, \varepsilon, \tau$	5	≥ 3		$T=mQS$
Helmert	t_x, t_y, m, ε	4	≥ 2	is the scalar multiple of an orthogonal matrix	$T=mQ$
with fixed scale	t_x, t_y, ε	3	≥ 2	is an orthogonal matrix	$T=Q$

The **translation vector** t and the matrices Q, S, M have the following representation:

$$t=\begin{pmatrix} t_x \\ t_y \end{pmatrix}$$ without reflexion: $$Q=\begin{pmatrix} cos(\varepsilon) & -sin(\varepsilon) \\ sin(\varepsilon) & cos(\varepsilon) \end{pmatrix}$$ with reflexion: $$Q=\begin{pmatrix} sin(\varepsilon) & cos(\varepsilon) \\ cos(\varepsilon) & -sin(\varepsilon) \end{pmatrix}$$

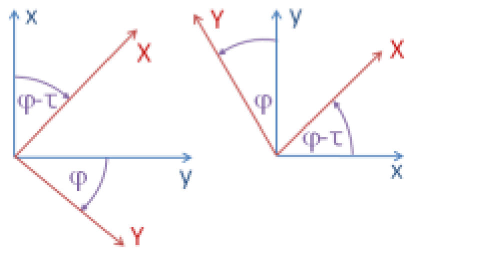
$$S=\begin{pmatrix} 1 & tan(\tau) \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} 1 & f \\ 0 & 1 \end{pmatrix} \qquad M=\begin{pmatrix} m_x & 0 \\ 0 & m_y \end{pmatrix}$$

The **rotation angle** ε is the rotation angle from the initial system to the target system about the origin. In the lefthanded system $\varepsilon>0$ means a clockwise rotation, in the righthanded system conversely.



planar lefthanded (left) and righthanded (right) system, shear mapping & rotation

After a shear mapping, the axes of the initial system do not enclose a right angle, but an angle diminished by the **shear angle** τ or enlarged by $-\tau$. In the computation protocol the shear factor $f = tan(\tau)$ is also reported.



planar lefthanded (left) and righthanded (right) system, rotation & shear mapping

The **scale factors** m_x, m_y are always positive and for some transformations of equal size. $m_x = m_y = m > 1$ means that the points are pulled apart. The grid scale factors are not included. E.g., if you change the `grid` to `cartesian`, the scale factors may also change.

Spatial transformations (3D)

Three coordinates must be given for all points. If the transformation type `auto-detect` is chosen, then spatial transformations are computed whenever this condition is satisfied, otherwise planar.

Whenever possible, the following transformations are computed one after the other:

Transformation	Parameter	Number	Contr.P.	Transformation matrix T	
Affine	$t_x, t_y, t_z, m_x, m_y, m_z, \varepsilon_x, \varepsilon_y, \varepsilon_z, \tau_{xy}, \tau_{xz}, \tau_{yz}$	12	≥ 4	is a general 3×3 matrix	
9-Param. Type 1	$t_x, t_y, t_z, m_x, m_y, m_z, \varepsilon_x, \varepsilon_y, \varepsilon_z$	9	≥ 3	has orthogonal row vectors	$T=MQ$
9-Param. Type 2	$t_x, t_y, t_z, \varepsilon_x, \varepsilon_y, \varepsilon_z, m_x, m_y, m_z$	9	≥ 3	has orthogonal column vectors	$T=QM$
Helmert	$t_x, t_y, t_z, m, \varepsilon_x, \varepsilon_y, \varepsilon_z$	7	≥ 3	is the scalar multiple of an orthogonal matrix	$T=mQ$
with fixed scale	$t_x, t_y, t_z, \varepsilon_x, \varepsilon_y, \varepsilon_z$	6	≥ 3	is an orthogonal matrix	$T=Q$

The **translation vector** t and the matrices S, M have the following representations:

$$t=\begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \qquad S=\begin{pmatrix} 1 & tan(\tau_{xy}) & tan(\tau_{xz}) \\ 0 & 1 & tan(\tau_{yz}) \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & f_{xy} & f_{xz} \\ 0 & 1 & f_{yz} \\ 0 & 0 & 1 \end{pmatrix} \qquad M=\begin{pmatrix} m_x & 0 & 0 \\ 0 & m_y & 0 \\ 0 & 0 & m_z \end{pmatrix}$$

The **3D shear mapping** represented by S may be viewed as a sequence of three planar shear mappings, in particular

1. a shear of y and z axis with shear angle τ_{yz}
2. a shear of x and z axis with shear angle τ_{xz}
3. a shear of x and y axis with shear angle τ_{xy}

If one interchanges the 1st and 2nd or the 2nd and 3rd planar shear mapping, one approximately obtains the same shear angles τ_{ij} , if they are small. In the computation protocol the shear factors $f_{ij} = tan(\tau_{ij})$ are also reported.

The **3D scaling** has up to three scale factors. The two scale factors of planar transformations are completed by a 3rd scale factor m_z .

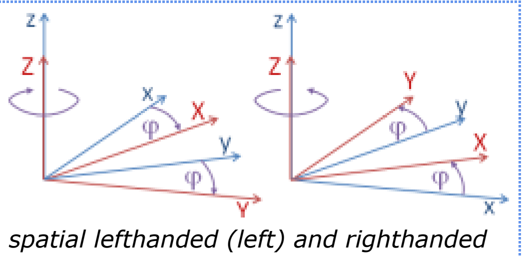
The matrix Q in the case without reflexion is a **rotation matrix** and can be represented by parameters in the following three different ways:

(a) with three **Eulerian angles** $\varepsilon_x, \varepsilon_y, \varepsilon_z$:

$$Q=\begin{pmatrix} cos\varepsilon_y \cdot cos\varepsilon_z & sin\varepsilon_x \cdot sin\varepsilon_y \cdot cos\varepsilon_z - cos\varepsilon_x \cdot sin\varepsilon_z & sin\varepsilon_x \cdot sin\varepsilon_z + cos\varepsilon_x \cdot sin\varepsilon_y \cdot cos\varepsilon_z \\ cos\varepsilon_y \cdot sin\varepsilon_z & cos\varepsilon_x \cdot cos\varepsilon_z + sin\varepsilon_x \cdot sin\varepsilon_y \cdot sin\varepsilon_z & cos\varepsilon_x \cdot sin\varepsilon_y \cdot sin\varepsilon_z - sin\varepsilon_x \cdot cos\varepsilon_z \\ -sin\varepsilon_y & sin\varepsilon_x \cdot cos\varepsilon_y & cos\varepsilon_x \cdot cos\varepsilon_y \end{pmatrix}$$

The total rotation can be viewed as a sequence of three rotations about coordinate axes. We use the geodetic convention:

1. Rotation about the x axis, rotation angle ε_x
2. Rotation about the y axis, rotation angle ε_y
3. Rotation about the z axis, rotation angle ε_z



spatial lefthanded (left) and righthanded (right) system, rotation about z axis

⚠ There are alternative conventions in use, mainly in the non-geodetic branches.

(b) with the quadruple of the **unit quaternion** (q_0, q_1, q_2, q_3) :

$$Q=\begin{pmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1 \cdot q_2-q_0 \cdot q_3) & 2(q_1 \cdot q_3+q_0 \cdot q_2) \\ 2(q_1 \cdot q_2+q_0 \cdot q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2 \cdot q_3-q_0 \cdot q_1) \\ 2(q_1 \cdot q_3-q_0 \cdot q_2) & 2(q_2 \cdot q_3+q_0 \cdot q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{pmatrix}$$

A quaternion usually permits a more elegant description of rotations in three dimensions than Eulerian angles. The four parameters satisfy the condition $q_0^2+q_1^2+q_2^2+q_3^2=1$ (unit quaternion).

(c) with **Eulerian axis** (e_x, e_y, e_z) as a unit vector and **rotation angle** ε about it:

$$Q=\begin{pmatrix} \cos\varepsilon+e_x^2(1-\cos\varepsilon) & e_xe_y(1-\cos\varepsilon)-e_z\sin\varepsilon & e_xe_z(1-\cos\varepsilon)+e_y\sin\varepsilon \\ e_xe_y(1-\cos\varepsilon)+e_z\sin\varepsilon & \cos\varepsilon+e_y^2(1-\cos\varepsilon) & e_ye_z(1-\cos\varepsilon)-e_x\sin\varepsilon \\ e_xe_z(1-\cos\varepsilon)-e_y\sin\varepsilon & e_ye_z(1-\cos\varepsilon)+e_x\sin\varepsilon & \cos\varepsilon+e_z^2(1-\cos\varepsilon) \end{pmatrix}$$

The rotation is here described about an oblique axis through the origin of the coordinate system, the so-called Eulerian axis. The parameters satisfy the condition $e_x^2+e_y^2+e_z^2=1$ (unit vector).

All angles $\varepsilon_x, \varepsilon_y, \varepsilon_z, \varepsilon$ are computed between `-π = -180° = -200 gon` and `π = 180° = 200 gon` in the chosen angle unit. They are defined

- for lefthanded systems positive clockwise,
- for righthanded systems positive counterclockwise.

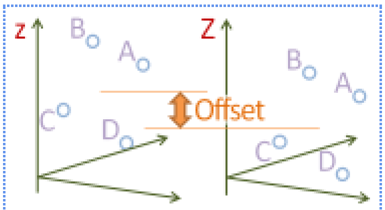
The viewing direction for this is opposite to the rotation axis.

⚠ There are alternative conventions in use, mainly in the non-geodetic branches.

If the initial system is a lefthanded system and the target system is a righthanded system or vice versa, T must additionally generate a **reflexion** . The indication of this is $det(T)<0$. We realize this by mirroring at the yz plane.

Planar transformations with adjustment of the vertical offset

In this case it is assumed that the vertical axes (Z or height) of both systems are parallel and have the same scale. From all control points with three coordinates in both systems the vertical offset is computed. At least one such point must exist. This offset is used to transform all non-control points into the opposite system. To the other two coordinates all applicable planar transformations listed above are applied.



Adjustment of the vertical offset

Least squares adjustment

If the number of coordinates of control points exceeds the number of transformation parameters in a transformation (redundancy) then an adjustment is computed by the method of weighted least squares. In this case accuracy measures are required, either a standard deviation σ or a weight p . In the first case the weight is computed by $p=1/\sigma^2$. For this purpose two modes are implemented:

Simple weighting (standard mode)
Accuracy measures can be chosen equal for each coordinate of the same kind, such that up to six measures can be specified.

Individual weighting (expert mode)
Herewith you can assign an individual accuracy measure to each coordinate of each control point.

Weight	Std	When computing the transformation parameters, the corresponding coordinate will be
0	INF	ignored.
INF or empty field	0 or empty field	used as a constraint.

After the adjustment there are often residuary misclosures. They coincide by magnitude to the residuals of least squares adjustment, but have the opposite sign:

residuary misclosure = given coordinate − coordinate computed from transformation parameters

Under the bonnet: Internally, IN DUBIO PRO GEO always works with the elements of t and T as adjustment parameters. For planar transformations we always get 6 parameters and for spatial transformations we get 12 parameters. For all non-affine transformations the restriction to special transformation matrices T is realised by restrictions for the elements of T . Only after the adjustment, scale, shear and rotation parameters are computed by QR or RQ decomposition of T In view of this, adjustment parameters and transformation parameters must be distinguished, except for the affine transformation.

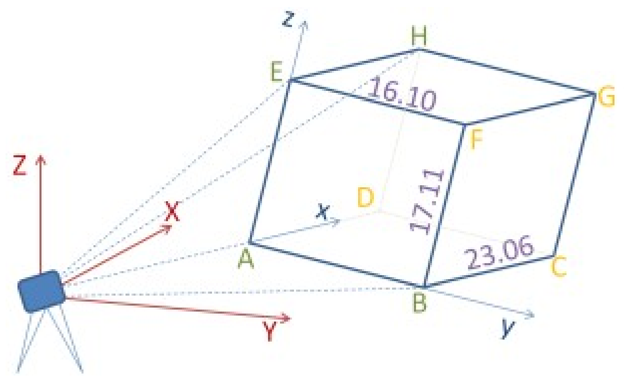
Parameters of the reverse transformation

If also transformation parameters of the reverse transformation are required, it is recommended to change the transformation direction by swapping the systems. Note that the parameters of the reverse transformation are not always obtained by reversing the signs of the angles and by inverting the scale factors.

☆ Cuboid through four vertices

Four vertices A,B,E,H of a cuboid ABCDEFGH with edges not parallel to coordinate axes are measured by tacheometry (⇒ figure). With a standard deviation of 0.02 the following coordinates in the station system (cartesian lefthanded system) X,Y,Z are obtained therefrom:

	X	Y	Z
A	14.029	17.058	8.073
B	23.616	29.751	5.516
E	14.272	20.210	24.880
H	32.863	6.737	27.163



Cuboid through four vertices

For all points C,D,F,G of the cuboid not sighted, the coordinates in the same system need to be determined.

In addition to the station system xyz we introduce an object coordinate system xyz as displayed above as a cartesian lefthanded system with axes along the edges of the cuboid. However, the edge lengths are not known. (The values in the figure above represent the true solution.) So we take these lengths as units of the coordinate axes. In this way for each coordinate axis a different length unit is defined. Now we assign object coordinates to the points A…H. They are treated as error-free.

object coordinates

	x	y	z
A	0	0	0
B	0	1	0
C	1	1	0
D	1	0	0
E	0	0	1
F	0	1	1
G	1	1	1
H	1	0	1

The spatial transformation from the object system to the station system must effect

1. three scale changes of the axes x,y,z and afterwards
2. one rotation about the Eulerian axis or
   three rotations about the coordinate axes x,y,z

Therefore, the appropriate transformation is a **9 parameter transformation type 2**.

Load example and “Compute”

From these coordinates, only the affine transformation and the 9 parameter transformation type 2 can be reasonably computed. For the other transformations, the control points do not approximately match (large residuary misclosures).

Result for the 9 Parameter Transformation Type 2 (3 scales ⇒ rotation)

The least squares method converged after 3 iterations.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 14.04018241 \\ 17.04096213 \\ 8.06932903 \end{pmatrix} + \begin{pmatrix} 18.5716594 & 9.56082510 & 0.24339293 \\ -13.4982723 & 12.72223499 & 3.18419440 \\ 2.2883147 & -2.54868580 & 16.80752749 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

comp.	object system			station system		
points	**x**	**y**	**z**	**X**	**Y**	**Z**
A	0	0	0	14.0401824	17.0409621	8.0693290
B	0	1	0	23.6010075	29.7631971	5.5206432
C	1	1	0	42.1726669	16.2649248	7.8089579
D	1	0	0	32.6118418	3.5426898	10.3576437
E	0	0	1	14.2835753	20.2251565	24.8768565
F	0	1	1	23.8444004	32.9473915	22.3281707
G	1	1	1	42.4160598	19.4491192	24.6164854
H	1	0	1	32.8552347	6.7268842	27.1651712
resid.	**x**	**y**	**z**	**X**	**Y**	**Z**
A	0	0	0	-0.0111824	0.0170379	0.0036710
B	0	0	0	0.0149925	-0.0121971	-0.0046432
E	0	0	0	-0.0115753	-0.0151565	0.0031435
H	0	0	0	0.0077653	0.0103158	-0.0021712

The residuary misclosures in the object system x,y,z are zero because they are assumed to be error-free. The residuary misclosures in the station system X,Y,Z account to a maximum of 0.017 and are therefore plausible.

Exercise: Prove that the new station system coordinates X,Y,Z of the 9 parameter transformation type 2 exactly represent a cuboid. Hint: Save coordinates and compute in Spatial polygons . Confirm that this would be only approximately true for the affine transformation.

Exercise: Use the button Swap systems to compute the reverse transformation. (Javascript must not be disabled.) Note that now the 9 parameter transformation type 1 is the desired transformation. The coordinates of the new points are identical.

Did you know? *If you use grid systems, then the grid scale factor is always applied.*

IN DUBIO PRO GEO Guide : Sets of angles and distances

Page contents

On a station there may be measurements in sets to the same targets. Measurement values may be given in arbitrary succession and may be arbitrarily missing. Set means, instrument errors and accuracy estimates are computed. The results may be processed with further IN DUBIO PRO GEO computation tools.

Introduction

When measuring with a tacheometer or a theodolite, target points are usually not sighted and measured only once, but **repeatedly and in both telescopic faces**. Such measurements are called **sets of angles and distances**. IN DUBIO PRO GEO uses geodetic adjustment to compute the best solution for angles and distances as well as for instrument errors of the tacheometer or theodolite. The scale circle must not be moved during the sets. All measurements must be taken with one and the same instrument immediately one after the other, such that we can rely on unchanged instrument errors. If steep sightings are due, than the instrument must not contain an uncorrected horizontal axis error.

Measurement values

Measurements are given as a Measurement lists . For the processing of the measurements the succession of the rows of the measurement list is arbitrary. However, for the practical measurement it is recommended to adhere to the classical rule: In every set all targets should be measured first clockwise in face I and then in reverse succession in face II. Measurement values may be **almost arbitrarily missing** . This may happen either in case of eliminated gross errors or in case of forgotten sightings.

For each horizontal angle r there must be given a zenith angle v such that it is possible to compute, how a collimation error acts on this measurement. The effect of the collimation error only slightly depends on the zenith angle, such that an approximate value is sufficient, e.g. `100 gon = 90°` oder `300 gon = 270°` for only slighlty inclined sightings. Additionally, at least one target point must have been sighted in both faces, otherwise no instrument errors are computable. Normally, for all target points there should be sightings in both faces.

Distances may be slope distances s or horizontal distances e .

All target heights th of one and the same target point must coincide, if given multiple times. If target heights are not specified for all sightings, it is assumed that the missing target heights coincide with the specified values. If for some target point no target height is given then the target height of this point remains undefined.

Set means and accuracy measures

The set means of the horizontal angles $\bar{r}$ and of the zenith angles $\bar{v}$ are computed by two separate **adjustments by observation equations** . All horizontal angles are equally weighted, and all zenith angles as well. As a result of both adjustments the a posteriori standard deviations of all measured values $\bar{\sigma}_r$, $\bar{\sigma}_v$ and of all set means $\bar{\sigma}_{\bar{r}}$, $\bar{\sigma}_{\bar{v}}$ are obtained.

If a set is complete, i.e. no angle values are missing, then the standard deviations of all set means coincide.

For **distances** no standard deviations are computed because electronic distance measurements mostly show only systematic measurement errors, such that these accuracy measures are much too optimistic. Instead of this, set means $\bar{s}$ or $\bar{e}$ and ranges Δs or Δe are computed.

Instrumental corrections

The instrument typically shows so-called instrument errors. Corrections for two kinds of such instrument errors can be computed from the measurement values: **collimation error** c and **vertical index error** i . The more targets are sighted in both faces, the higher the accuracy of the computed instrumental corrections. The set means are already corrected for those errors. If you want to correct further uncorrected measured values r',v' then you have to compute

in telescopic face I $r = r' + c \,/\, sin(v) \quad v = v' + i$

in telescopic face II $r = r' - c \,/\, sin(v) \quad v = v' - i$

Note that the obtained values represent corrections in telescopic face I. (''Errors'' would actually get the opposite sign.) If the measurement values are already corrected for those errors, then one would only get the updates for those corrections.

Horizontal axis errors are not computed because for this purpose exclusively steep sightings are required, which are rarely found in normal sets of angles.

Sets of angles and distances may also be processed by the Universal computer . However, no instrument errors are computed and no accuracies are estimated. The accuracy of the results is therefore usually worse.

Further computations with set means

Set means can be transfered to the Universal computer or in Traverses for further computations, e.g. for the determination of coordinates of station and/or some target points. If there is not yet any measurement list defined in the target computation tool then such a list can be created from the set means. Otherwise it can be overwritten by the current set means, or the set means are appended as a new measurement list. In the latter case only those data are appended which fit into the current format specification of station and target row of the target computation tool.

⚠ Note that some data may not be transfered. If the succession of the data disagrees then it is adapted. Ideally you choose the formats of the target rows consistently.

Set means can also be transfered to Station centring for further computations. But here an existing measurement list is always overwritten because this target computation tool only supports one station setup at a time.

Loading adjustment models in "Adjustment with observation equations"

Vertical networks and Sets of angles and distances can be re-adjusted with Adjustment with observation equations . This yields the following advantages:

- Weights can be changes. E.g. target points of low sighting accuracy can be downweighted.
- Outliers can be automatically detected by w- or τ-test.
- The accuracy may be tested vs. a theoretical value. For example, it may be tested statistically, if the accuracy specification of the manufacturer of the instrument is met.
- The redundancy parts, full cofactor matrices and other interesting values are displayed.
- For many values more digits are displayed, if desired.

Forthcoming: More tools will provide this option.

The parameters of the adjustment are the set means in the succession of the set means table with appended instrument error, either the collimation error c or the vertical index error i . The observations are the measured horizontal or zenith angles in the succession of the measurement input area.

Pure processing of horizontal angles

If only horizontal angles need to be processed and all sightings are all gently inclined then

- for all sightings in telescopic face I the zenith angle `100 gon = 90°` and
- for all sightings in telescopic face II the zenith angle `300 gon = 270°`

should be specified. The following horizontal angles (all gently inclined) on the station S0, from which one measured value is evidently grossly erroneous due to a target point mix-up, need to be processed:

target point	set 1 [gon] telescopic face I	set 1 [gon] telescopic face II	set 2 [gon] telescopic face I	set 2 [gon] telescopic face II
T1	16.1063	216.1104	16.1083	216.1139
T2	~~17.1165~~	223.0712	23.0697	223.0787
T3	91.0214	291.0277	91.0312	291.0303

Load example and "Compute"

As a result one obtains the three set means: `16.10973 gon; 23.07251 gon; 91.02765 gon` . The posterior standard deviation for the mean of two faces is obtained as `2.6 mgon` . The posterior standard deviations of the set means amount to `1.8 mgon` for T1 and T3 and to `2.1 mgon` for T2. This value is worse due to the missing measurement. However, the three not grossly erroneous measurement values have been used. The standard deviations of the zenith angles are all zero, because these measurements are faked, and must be ignored.

Exercise: Load this adjustment model of the horizontal angles into Adjustment with observation equations and test if the specification of the manufacturer of a priori standard deviation for the mean or two faces of `2 mgon` (corresponds to a priori standard deviation for a uncorrected single measurement of `2.8 mgon`) with a probability of type I decision error of `0.01 = 1%` may be considered as met. At the same time, check that no further gross errors are detected.

Processing of zenith angles with slope distances

If only zenith angles need to be processed then all other measurement values can be omitted.

In the following example on the station S0 zenith angles in two faces and slope distances are measured:

target point	zenith angle [gon] telescopic face I	telescopic face II	slope distance telescopic face I	telescopic face II	target height
T1	90.1866	309.8157	17.589	17.590	1.40
T2	98.5077	301.4979	23.697	23.697	1.40
T3	94.9949	305.0066	14.291	14.294	1.40

Load example and "Compute"

As a result one obtains for the three set means: `90.18545 gon; 98.50490 gon; 94.99415 gon`. The a posteriori standard deviations of the set means amount to `1.1 mgon` . The a posteriori standard deviations of the set means has the same value because only one set has been measured and it is complete. The vertical index error amounts to `-1.6 mgon` and its a posteriori standard deviation is obtained as `0.6 mgon` . Horizontal angles and zenith angles are not computed.

Exercise: Transfer the set means by the button go into Universal computer and add some arbitrary coordinates of the station S0 there and an instrument height. Start the computation and note that heights are computed also for the target points. If the station height would have been set to `100.0000` and instrument and target heights set to zero, then the heights of the target points are obtained as `101.30 99.15 99.72`. If coordinates of a target point would have been specified, then the heights of all other points would have been obtained in the same manner.

Joint processing of all measurements

The measurement of both previous examples may be introduced in a joints processing. However, the second set of the horizontal angles must be dropped because no faked zenith angles can be used here. At best you could repeat the zenith angles from set 2, this would not change the set means, but the standard deviations would be computed too small. Since in one target point row a horizontal angle is missing now, you could shift the column of horizontal angles to the end of the measurement list . This can be avoided by letting the missing value empty. For this purpose you write '';;'' in the measurement list, i.e. two separating characters, which are not joined to one.

Load example and "Compute"

Exercise: Repeat the exercises from the previous examples with the entire data set. Although only one set of horizontal angles is given and it is on top of that incomplete, an adjustment is possible. However, the total redundancy is only 1, which makes adjustment results unreliable. Note that the horizontal angle to point T2 is now fully uncontrollable. In the universal computer the target coordinates are now fully computed in the local station system. For example, using the station coordinates `S0 100.000 100.000 100.000` and with instrument height and orientation angle in the station row being both `0.000` one obtains

PName	X	Y	Z
T1	116.827452	104.351096	101.30098431
T2	122.152065	108.397866	99.15647232
T3	102.002168	114.106965	99.72268593

Did you know? *IN DUBIO PRO GEO also runs on the smartphone or tablet computer (in the browser)*

IN DUBIO PRO GEO Guide : Station centring

Page contents

Eccentrially measured sets of polar measurements are computationally transferred to a new centre. You obtain the values, which would have been measured on the centre, optionally including an Error propagation. If all required values are given, a spatial centring is computed.

Introduction

Due to setup or sighting obstacles it is oftentimes not possible to measure tacheometrically at the desired sites. In this case you can take measurements on the closest possible site and compute a station centring.

Naturally the centring computation slightly downgrades the accuracy of the measurements. This effect can be determined by a Error propagation . However, this computation is only approximate for slope sightings, which is indicated by a warning.

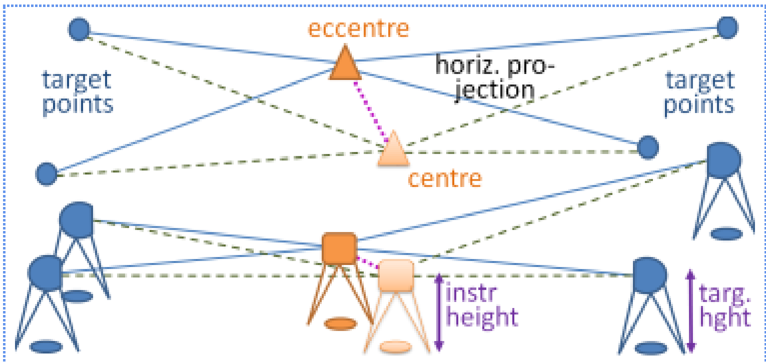
Vertical centring

For spatial centrings two options are available:

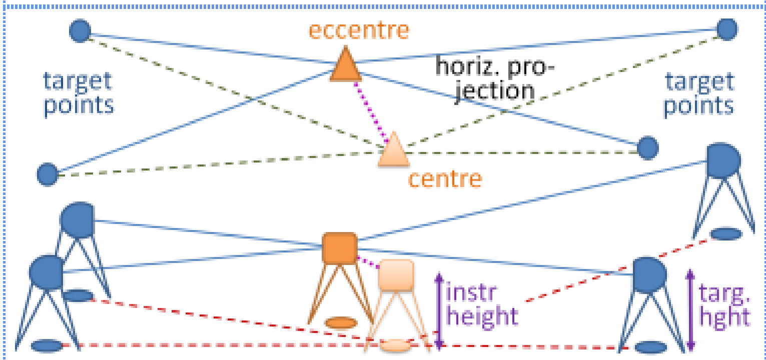
Vertical centring to

targeted measuring marks
: The centric computation results refer to the targeted measuring marks. The given target heights (or the default value) are not used, but passed through, if applicable, e.g. to Universal computer.

the actual points
: The centric computation results refer to the actual points below the targeted measuring marks (or above, if the target height is negativ). The given target heights (or the default value) are subtracted.



The centric values refer to the targeted measuring marks.



The centric values refer to the actual points.

This selection is in effect only for spatial centring. If all instrument and target heights are zero, there is no difference here.

☆ Eccentric angular measurements to remote targets

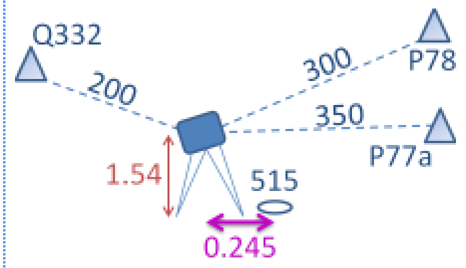
Angular measurements to remote target points `Q332, P77a, P78` have been taken, however, not on the actual station point `551`, but slightly offside, i.e. **eccentrically**. The distances to the targets are approximately known.

We try to generate measurements, as we would have taken them, if we would have measured on the actually desired station point `551` in the same instrument height. All target heights are zero, which is specified by the default value. In this case both options for ⇑ Vertical centring are equivalent.

```
Q332   0.000 97.636 200 // default target
height
P77a 177.565 98.955 350 // is set to 0.000
P78  191.295 93.342 300 //

515  210     100    0.245 1.54 // desired
                               // centre
```

Measurement lists, Winkeleinheit: Gon
Column format: point name, horizontal angle, zenith angle, slope distance, target height



Eccentric angular measurements to remote targets

Naturally the centring computation slightly downgrades the accuracy of the measurements. We try to determine the size of this effect by Error propagation . The standard deviation of the angular measurements to remote targets is assumed to be `0.002 gon` . Our distances to remote targets are only guesses, such that we assign a standard deviation of `10` to it. However, the eccentricity `0.245` is needed rather precisely, its standard deviation amounts to `0.005`. Due to the small eccentricity, the horizontal angle from eccentre to centre may be rather imprecise, the standard deviation is assumed to be `5.0 gon` . The zenith angle has been defined exactly by `100 gon` and is therefore error-free.

Load example and “Compute”

We obtain the centric computation results displayed at the right. The standard deviations of the horizontal angles are enlarged for to `0.005…0.007 gon`. By changing the input you find out that this effect is mainly due to the standard deviation of the horizontal angle from eccentre to centre being `5.0 gon` . For the zenith angles no change is found.

Target Pname	horizontal angle	zenith angle
Q332	0.0122	97.63886
P77a	177.5432	98.95436
P78	191.2798	93.33680

Did you know? *IN DUBIO PRO GEO is always up to date, without cumbrously installing updates.*

IN DUBIO PRO GEO Guide : Atmospheric EDM correction

Page contents

The refractivity of air for specified atmospheric conditions in the visual and near infrared spectrum is computed, optionally including an Error propagation. A distance measurement value may be corrected.

Introduction

The methods of electronic distance measurement (EDM) are the most important modern geodetic methods of distance measurement. They are predominantly applied for tacheometry (total stations) and for laser scanners. In any case, the **propagation velocity of the carrier wave** is required. The longer the distance to be measured and the higher the accuracy requirement for the distance measurement value, the more accurate this value must be known or determined. It more or less depends on the following quantities:

- carrier wavelength of the EDM λ
- air temperature t
- air pressure p
- air humidity

From these quantites the refractivity of air and consequently the propagation velocity in the range of validity of their values can be computed. This task is performed by the computation tool Atmospheric correction.

Carrier wavelengths of selected EDM

Today we practically apply only EDM, which use light waves in the red or near infrared (NIR) spectral range.

Spectral range: red, visible		**Spectral range: near infrared (NIR)**	
EDM/tacheometre/total station	**nm**	**EDM/tacheometre/total station**	**nm**
Kern Mekometer ME5000	633	Leica TC 2003	850
Leica TC 400 / TC 800	658	Leica TC 110	850
Leica TS30 / TM30 / TS02	658	Leica TDA / TMA 5005	850
Trimble S8	660	Trimble 3300	860
Leica TPS 110 / 1100 / 1200	670	ZEISS Rec Elta	860
(prismless+LongRange)		ZEISS ELTA4	869
Leica TPS 110 / 1100 / 1200	780	Trimble S6	870
(on prisms)		ZEISS ELTA3	910

Air temperature and air pressure

At the instant of measurement the temperature of the dry air and the air pressure along the signal path of the EDM must be known or determined. The air temperature is most critical here, because it is required relatively accurate and it is hardly constant over longer distance paths. Moreover, the air temperature changes in time. Best results would be obtained from measurements of air temperature and air pressure at equispaced points distributed over the signal path and the derivation of representative averages of these values. Unfortunately, this approach is too laborious as compared to the benefit, such that it is usually omitted. For short distances only the atmospheric measurement values at the instrument are used.

A temperature error of 1 Kelvin or a pressure error or 3 hPa produce a distance error of about `1 ppm = 1 mm/km`.

Humidity of air

The humidity of air is only relevant for measurements of highest accuracy or in cases of wet or hot weather. Optionally, one of the following specifications is possible:

Measure of humidity	Symbol	Unit
partial pressure of the water vapour	e	hPa (hectopascal)
relative humidity of air	h	% (percent)
wet bulb temperature	θ	°C (Celsius)

If this information is not available, the relative humidity of air should be kept at the default value of 60%. This value causes a distance error of at most `2 ppm = 2 mm/km` (reference: Leica Geosystems).

Normal atmosphere and distance measurement value to be corrected

Frequently there is an uncorrected distance measurement value to be corrected. This value refers

- either to some **mean** atmospheric conditions, the **normal atmosphere** , which is often defined by the manufacturer of the EDM, to avoid corrections in case of low accuracy requirements,
- or to some **erroneous** atmospheric conditions, because the correct conditions were unknown at the instance of measurement time or could not be taken into account or in case of a misapprehension.

For the correction it is required that the group refractivity of this atmosphere is known. Either the manufacture of the EDM specifies the group refractivity of the normal atmosphere or values of temperature, pressure and humidity of air, for which the uncorrected distance measurement value would be valid, are known. In the second case the group refractivity can be computed in a first run of Atmospheric correction .

Formulae

We use the formulae recommended by Buck (1981) and Rüeger (2002, p. 87) :

$$N_{gr} = 287.6155 + \frac{4.88660}{\lambda^2} + \frac{0.06800}{\lambda^4} \qquad e = \frac{h}{100} \cdot 10^x$$

$$N_{ph} = 287.6155 + \frac{1.62887}{\lambda^2} + \frac{0.01360}{\lambda^4} \qquad x = \frac{7.5 \cdot t}{237.3 + t} + 0.7857$$

$$N_L = \frac{273.15}{1013.25} \cdot \frac{N_{gr} \cdot p}{273.15 + t} - \frac{11.27 \cdot e}{273.15 + t} \qquad e_w = 6.112 \cdot \exp \frac{17.502 \cdot \theta}{240.97 + \theta}$$

$$ppm = (N_o - N_L) \;/\; (1 + N_L \cdot 10^{-6}) \qquad e = e_w - p \cdot (t - \theta) \cdot 0.00066 \cdot (1 + 0.00115 \cdot \theta)$$

$$D = D' \cdot (1 + ppm \cdot 10^{-6})$$

Symbols:

N_{gr}, N_{ph}	group refractivity and phase refractivity of the standard atmosphere $t = 0°C, p = 1013.25\ hPa, e = 0\ hPa$, CO2-Gehalt 0,0375%		
λ	carrier wavelenght in µm	t	dry bulb air temperature in °C
p	air pressure in hPa	e	partial vapour pressure in hPa
h	air humidity in %	x	auxiliary variable
N_L	group refractivity of the real atmosphere	ppm	distance measurement correction in ppm
N_o	group refractivity of the normal atmosphere	θ	wet bulb temperature
D',D	uncorrected and corrected distance measurement value, resp.	e_w	saturation vapour pressure related to wetbulb temperature in hPa

Leica TS30, correction of erroneous settings

Using a total station TS30 (manufacturer: Leica Geosystems, coaxial visible red laser, $\lambda = 658\ nm = 0{,}658\ \mu m$) a distance measurement was performed with the erroneous settings

$$t = 12°C,\ p = 1013.25\ hPa,\ h = 60\ \%$$

The result was $D'=175.989$. Actually, at the instance of measurement time along the signal path the following conditions were met:

$$t = 23°C,\ p = 990.7\ hPa,\ h = 20\ \%$$

The corrected distance is required.

Firstly, the group refractivity N_o of the atmosphere, to which the distance measurement value D' applies (normal atmosphere), must be computed.

Load example and “Compute”

The result is `286.34`. Secondly, the group refractivity of the real atmosphere N_L is computed. At the same time the distance measurement value is corrected.

Load example and “Compute”

The correction result amouts to `+16.7 ppm` , this corresponds to `+2.9 mm` . The corrected distance is then `175.9919 m` . Finally, we want to know about the influence of errors in the atmospheric parameters on the correction. Let us assume the following errors:

$$\Delta t = 2°C,\ \Delta p = 10\ hPa,\ \Delta h = 20\ \%$$

Load example and “Compute”

By Error propagation we obtain for the maximum error of the correction result an amount of `4.8 ppm` , this corresponds to `0.84 mm`.

Exercise: Determine the fraction of this error purely due to the error in the air humidity.

***Did you know?** The refractivity N relates to the refractive index n by $N=(n\text{-}1)\cdot 10^6$.*

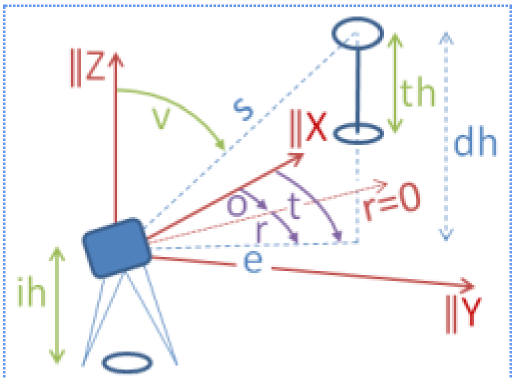
IN DUBIO PRO GEO Guide : Traverses

Page contents

From point coordinates and polar measurements it is tried to compute a classical traverse with proportioning of misclosures. In arbitrary measurements the longest possible traverse is detected. All that is rationally evaluable in one way or another, will be evaluated.

What is a computable traverse?

A traverse consists of a series of points, where consecutive points are connected by measured distances and angles as well as opposite angles. For one or more of these points, coordinates must be given. These points can be arranged at arbitrary positions in the traverse (begin and/or end and/or in between). At some points an orientation angle can be given, or such a value can be computed from angles measured between known points. These points can as well be arranged at arbitrary positions in the traverse.



Coordinates and polar measurements

If you want to compute a **free traverse** , which has no point with known coordinates, then you should define a local coordinate system yourself by assigning arbitrary coordinates to an arbitrary point. Newly computed points are then obtained in this coordinate system.

If only **a single** point with known coordinates is given and no orientation angle is given or is computable there and also all other stations come without orientation angle, then exceptionally an auxiliary orientation is defined arbitrarily. A warning is issued.

See how a ☝closed traverse is computed.

Coordinate lists and measurement lists

Coordinates and measurements are given by Coordinate lists and Measurement lists exactly as in the Universal computer. These lists can contain arbitrary extra coordinates and measurements, which are not usable for the traverse. These are ignored. The **longest possible** traverse is detected automatically and computed.

The **succession of points** in the coordinate list and the stations in the measurement list as well as the targets belonging to a station are as always arbitrary. These points must **not** be sorted along the traverse.

The **direction of the traverse**, in which the computation proceeds, follows roughly the succession of points in the coordinate list. If this is undesired, the succession can be reversed. The results do not change, but are displayed in reverse succession.

The measurement list is preliminarily computed, such that measurements belonging to targets measured multiple times on a stations are averaged. Those equi-weighted arithmetic means as well as the corresponding ranges are computed and displayed as well as possibly compared with a critical value specified in Settings .

Stations should not be occupied multiple times, otherwise only the first occupation in the measurement list is used and the rest is ignored. A warning is issued.

Orientations and station setups

The computation of the traverse starts with the angle orientation at the stations where this is possible. The orientation angles o can be given in the station row of the measurement list. Additionally, it is tried to compute them from horizontal angles r to targets with known coordinates inside or outside the traverse. If more than one such measurements are found, the equi-weighted arithmetic mean of the orientation angles is used (station setup). The range is computed and displayed as well as compared with a critical value specified in Settings .

Horizontal angle closures

If at multiple stations an orientation angle has been computed (often this happens at the beginning and at the end of the traverse), then from any extra orientation a constraint (restriction) can be derived. The corresponding misclosures are displayed. If the option "Do not change measured angles (report only misclosures)" is chosen, the inner geometry of the traverse is left unchanged. Consequently, given or measured orientations have no influence on the coordinates of newly computed points. Otherwise, the misclosures are portioned equally to the measured angles. A portioning of the misclosures also or exclusively to the orientation angles is not yet supported. The traverse is now adjusted with respect to the angles.

Coordinate closures

Now the traverse is transformed onto the known points without further change of the inner geometry. These points serve as control points. The given coordinates represent the target system, equal weights are assigned to them. Further known points are ignored. If only one point is known, a pure translation is performed. For two or more points the traverse is additionally rotated. If the option "Adapt scale of distances to coordinates of known points" is chosen, the scale of the distances is also changed, which is identical to a Helmert transformation. This gives smaller residuals, which however does not necessarily improve the results. For two known points these residuals are always equal to zero. The computed scale factor is displayed.

Height computation

The computation of the traverse can be performed purely in the horizontal plane (2D). As soon as points with three coordinates are given, as well as measurements, which permit a spatial computation, this will be performed. Such measurements are zenith angles or height differences as well as instrument and target heights. It so also possible that such coordinates and measurements are given only segmentwise, i.e. not along the whole traverse. The spatial computation is then performed only in these segments of the traverse. For this purpose, local heights are computed for all points, possibly only segmentwise. From the given heights (Z coordinates) of known points a vertical offset is computed as equi-weighted arithmetic mean of the height differences. The local heights are transformed onto the given heights. The residuals are displayed.

Transfer of results to other computation tools

At the end of the computation a coordinate list of the results can be created. As requested, it can contain:

- only newly computed points, not known before
- only points on traverse, known and newly computed
- all points of the measurement list, including those outside the computed traverse

Points of the given coordinate list, which do not appear in the measurement list, are always ignored. For the other points the coordinate list can be created using the given (old) coordinates or the newly computed coordinates, which differ by the residuals.

The created list is not displayed in the computation results, but can be transferred to other computation tools or to a new browser tab.

If unknown points outside the traverse are found in the measurement list, which are therefore not yet computed, the measurement list and the created coordinate list of all known and newly computed points can be transferred to ❓Universal computer . If possible, all remaining points are computed, without changing the points already computed.

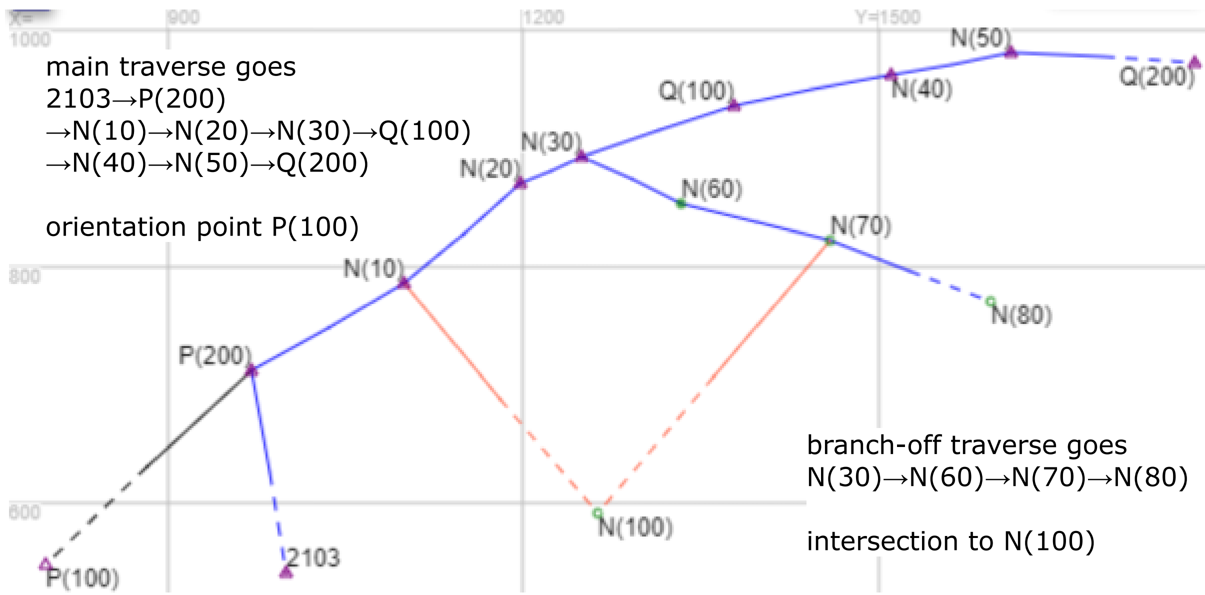
☹ If a distance scale is computed, this will not be applied automatically in the ❓Universal computer .

☰ ☆ Branched traverse with spatial intersection

Let us consider this traverse: From point `P(200)` to point `N(50)` measurements have been taken at seven stations, the first and the fifth station are known points. At the first station two known points have been targeted: `P(100)` and `2103`. At the last station the known point `Q(200)` has been targeted. Additionally, on point `N(50)` an orientation angle is given, which was possibly derived from computations.

At point `N(30)` a second traverse branches off with two additional stations `N(60)` and `N(70)` . The end point `N(80)` is a pure target point. Moreover, from `N(10)` and `N(70)` horizontal angles to point `N(100)` have been measured. These points are all unknown.

All known points except `P(100)`, which is a pure orientation point, have given heights. At all stations zenith angles as well as instrument and target heights have been measured, such that the unknown heights can be computed.



Pointnames and coordinates

```
P(100) 549.836  796.295
P(200) 712.406  971.427 116.100
2103   542.395  999.909  99.164
Q(200) 972.862 1765.642  97.506
Q(100) 936.602 1378.368 102.914
```

Type of system: XYZ lefthanded
Column format: pointname coordinates

Pointnames and measurements

```
P(200)   1.789
P(100)  68.554
2103     5.646 173.393 106.888 0
N(10)  283.126 148.800 102.987 0
+++++
N(10)    0
P(200)  66.911 148.804  97.007 1.789
N(20)  254.485 128.726  98.222 0
N(100) 355.842   ;   ;   97.943
```

Format station row: point name, instrument height
Format target row: point name, horizontal angle, slope distance, zenith angle, target height
all angles in Grads

Load example and "Compute"

The computation starts with the **station setup** at the station `P(200)`. In this way we obtain the orientation angle `183.7999 gon`. The corresponding range amounts to `26.2 mgon`. Additionally, at the point `N(50)` the given orientation angle is listed.

Since two stations are oriented horizontally, a **horizontal angle closure** can be computed. The misclosure amounts to `85.1 mgon`. It is distributed equally to all horizontal angles in the segment `P(200)→N(50)` and the angles are adjusted. The horizontal angles outside of this segment cannot be adjusted.

Now the **coordinate closures** can be computed by transformation onto the four known points, in this case by translation und rotation. (The scale is fixed.) In this way we obtain residuals up to `35 mm` Form the transformed coordinates a coordinate list is created. It can be transferred to other computation tools.

Exercise: Select the option "Adapt scale of distances to coordinates of known points" and see for yourself that the residuals amount only up to `25 mm` . The distance scale factor yields `1.00003658`.

To compute the remaining points `N(60)`,`N(70)`,`N(80)`,`N(100)` we transfer the computed coordinates and all given measurements to the Universal computer.

Compute a closed traverse

Here we have measured distances as well as horizontal angles and opposite angles also between the first and last point of the traverse, such that a closed loop is formed. In principle, there may even be more tie lines (diagonals) such that a network of traverses is created. At the moment, such an arrangement is not processed with this computation tool. However, you can cut through the closed traverse at one point and name this cut point differently at the end of the now open traverse. This point is listed twice in the coordinate list with different names and identical coordinates.

Did you know? *You may switch between german and english language at all times.*

IN DUBIO PRO GEO Guide : Universal computer

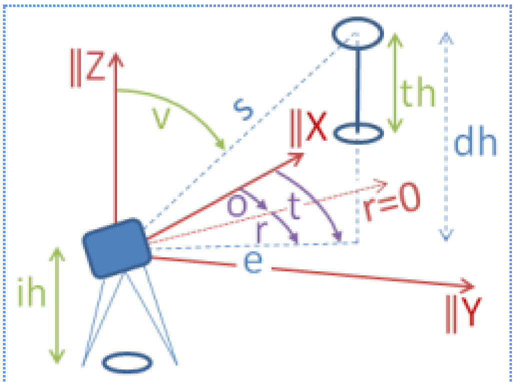
Page contents

From point coordinates and polar measurements all possible quantities are computed. Computation rules between these quantities are set up and consecutively applied until no new values are obtained. This is done in any possible way. By this procedure you often get many different results, which can be compared to detect gross errors. In this case the medians of the computed values represent the results of a robust estimation.

Introduction

Point coordinates may be cartesian or grid coordinates and are here denoted as X, Y, Z . Polar measurements may be:

- orientation angle o
- horizontal angle r
- azimuth t
- horizontal distance e
- slope distance s
- zenith angle v
- height difference dh
- instrument height ih
- target height th



Coordinates and polar measurements

The combination of coordinates and measurement may be fully **arbitrary** . It is always computed, what is computable based on the specified quantities, in fact **thoroughly** . Also all measurements can be missing, while only station and target point names are given. It is tried to compute polar values from given coordinates. ☆Polar values computed from cartesian coordinates

Adjustment, robust estimation, prevention of inappropriate sections

If **different computation schemes** result in the same quantities, all will be processed. The **count, minimum, maximum and range** of the obtained values are computed. If the range is small then the median should be considered as the final result, otherwise a gross error in the start values is probable. ⇓Outlier detection. In case of many different computation schemes the median is hardly falsified by a few gross errors in the input data. There is no adjustment in the traditional geodetic sense of least squares, but the results are excellently verifiable and even **adjusted** in the sense of a robust estimation.

If for some quantity a sufficient number of computation schemes is found, it is investigated, if the range can be decreased by prevention of **inappropriate geodetic sections** or inappropriate computations in acute triangles. If this case applies, the related values are discarded. The total count of discarded values is documented. In the ☆Inaccessible point with auxiliary triangles 1533 out of 2590 computed values have been discarded due to inappropriate section angles.

Additional options

The type of system of the coordinates of the known points must be **cartesian lefthanded** or **grid system** . The distances must be corrected for instrumental errors like prism offsets, but not for the grid scale factor.

For known points at least **both horizontal coordinates** X,Y or Northing, Easting must be specified.

⚠ Missing coordinates and measurements are always treated as unknown, including missing instrument heights *ih* and target heights *th* ! If e.g. heights for new points are required then station and target heights must be completely specified.

Whenever possible, start values are by default also computed from other start values. This facilitates a check for gross errors also here. However, if one wants to define start values as errorfree, it is possible to suppress the **recomputation of start values** . This can be done for all point coordinates (only XY or only Z or XYZ), for all polar measurements (rtesvo...) or for all start quantities (XYZrte...).

Using this Universal computer one may also process Sets of angles and distances . However, no instrument errors are computed and no accuracies are estimated. Therefore, we recommend to use the computation tool Sets of angles and distances . You can transfer the set means obtained there directly into this Universal computer .

Blind target points

Oftentimes the Universal computer computes polar values only between points related by measurements (station and target in a station setup). More results are sometimes obtained by adding blind target points without measurements. E.g. if you want to obtain the horizontal distance between two known or computed points then they may be additionally given as station and target without measurements. This value may be used in further computations, if useful. Find such a case in the ☆ Polar values computed from cartesian coordinates.

Count of start values, computation time and storage capacity limit

The **total number of start values** (specified point coordinates and polar measurements) is limited to 256. They must be counted as follows:

- Start values irrelevant for the computation are not counted.
- Quantities measured multiple times including azimuth and distance in opposite sights are counted only once, because they are averaged before the start of the main computation.
- For a known point both horizontal coordinates X,Y or Northing, Easting count only as one value.
- The default target height (default value for the target height) counts as one value at each point, where it is inserted.

The start values are divided into two groups:

- those, which are analysed in a processing engine (⇓ next section) for possible computable quantities and computation schemes, they must not exceed a number of 126 values, and
- the remaining, which are later used in a postprocessing step.

The count of start values actually used is documented.

The **total computation time** is limited to 90 s and the **total storage consumption** is limited to 128MB. Due to the laborious search for different computation schemes, in case of many start values a complete solution may take some time. The computation time can also be limited more strongly by the user. If for large scale computations this limit is eventually reached, then not all theoretically possible values are computed, but still as many as required for a reliable result in the sense of a robust estimation. A warning is issued. However, this does not mean that more computation time would really give more results.

If one of these limits is exceeded, please try to split the task. The computation time reduces also, if you suppress the ⇑recomputation of start values .

☰ Universal computer finds out for itself, what can be computed and how

Below we sketch the universal computer algorithm. If this information is unimportant for your, please skip this section.

1. From the catalogue of available **computation rules** (i.e. all angle and triangle relationships, all geodetic sections, all conversions between cartesian coordinates and polar quantities) a list of computation rules, which may in principle be useful for the computation of missing quantities, is compiled. From this list all rules are successively deleted, which are not applicable, e.g. because there is no applicable rule for the computation of some missing input quantity.
2. Now all computation rules with output quantites, which are not input quantities of other rules, are sourced out from the processing engine and move to a postprocessing step.
3. Starting from known point coordinates and polar measurements (start values) it is then tried to compute new quantities by sequential application of computation rules. In case of redundant start values almost any quantity can be computed by many different **computation schemes** and consequently assumes many different values. Here two computation schemes are only considered as different if for the two sets of used start quantities holds: One set is not subset of the other set and vice versa. Also for start quantities we often obtain new values by computing them from other start values. For each quantity it is tried to generate as many different values as possible.
4. Then all computation schemes are compiled to a **computation procedure** and are processed. The computation procedure is ⇓documented.
5. If for some computation scheme there is a **nonunique solution** , as e.g. for the arcs-intersection, it is tried to resolve the nonuniqueness by comparison with other computation schemes of the same quantity. If this fails permanently then it is tried to continue the computation in parallel with both solutions. If this case arises repeatedly, as in the ☆Planar trilateration network , then a large number of solutions may arise. Those are all computed and displayed, but only **successively** by the button Next solution » etc. A warning gives notice of this problem.
6. If for some quantity a sufficient count of computation schemes is found, it is investigated, if the extreme values resulted from inappropriate (i.e. acute-angled) geodetic sections or inappropriate computations in acute triangles. If this case applies, the values are discarded, and the investigation continues with the second smallest and second largest values, until no further values are to be discarded. The total count of discarded values is documented. In the ⇓Documentation of the computation procedure you also find the discarded values.
7. If it happens that computations do not give a result, e.g. because the arcs-intersection does not have a real point of intersection, is is not executed. In the ⇓Documentation of the computation procedure the result **not a number** is shown. This gives a clue, where the error in the input data must be searched for. If for some quantity exclusively such non-real results are obtained, a warning is issued, and the quantity is not displayed in the ⇓Table of results.

8. Now the postprocessing step is performed in the same manner, however, the extensive search for computation schemes is unnecessary.

☹ If a computation yields multiple solutions then calling for the solutions currently not displayed is only possible immediately after the computation. If in the meantime the computation procedure has been analysed or the outlier detection has been invoked or the result table has been sorted, the buttons for calling these solutions disappear. A recomputation puts things right.

Table of results

All computed quantities are displayed in a table, sorted by type of value. Within the types of value, sorting is by (station) point names, or optionally by ranges (smallest first). For horizontal coordinates, X and Y is always listed contiguously. When sorting by ranges, the sum of ranges of X and Y counts.

Azimuths t and horizontal distances e are always displayed in one direction only, i.e. such that the two pointnames are in lexicographic order. For the opposite direction you change t by `π = 180° = 200 gon`. Height differences dh and slope distances s in sight and backsight may be different, if the related instrument and target heights are not equal.

The column **values** shows, how many values for each quantity are obtained. For position coordinates of one point the values are the same and are given only once. Possibly discarded inappropriate sections and `not a number` results are not counted. A start quantity, which in addition is computed x mal berechnet wurde, erscheint als `1+x`.

Documentation of the computation procedure

The computation procedure can be retraced in detail. If you do not need this feature, skip this section. For a desired quantity click on the value(s) in column **values** to get the detailed documentation of the computation procedure and the individual results.

In addition to the symbols for the measurements listed in the ⇑Introduction the following abbreviations are used. The table shows, which result a computation step may produce.

Symbol	computation step	unique	ambiguous	bad section	not a number
Rec2Pol/ Pol2Rec	coordinate conversion cartesian⇔polar	X			
AA	arcs-intersection		X	X	X
LA	straightline-arc-intersection	X	X	X	X
LL	straightlines-intersection	X		X	X
RE	resection	X		X	X

Assume that the angle unit is `gon` . A computation step is documented like:

$t(P1\rightarrow3)_7 = r(3°2\rightarrow P1) + o(3°2)_3 \pm 200 = 69.965792398379$

This can be understood as follows: The 7th value of the azimuth t from point `P1` to point `3` is computed from the given horizontal angle r (index missing → start value) measured during the second setup on the station point `3` to point `P1` and the 3rd computed value of the orientation angle o for the second setup on the station point `3` . The result is `69.965792398379` in the chosen unit of angle `Grad` . If there is only on setup per station, the setup counter behind the station point name is omitted. Or e.g.:

$XY(2)_9 = LL(XY(1)_5, t(1\rightarrow2)_2, XY(Q3), t(2\rightarrow Q3)_4) =$ **not a number**

This can be understood as follows: The 9th computed values of the positional coordinates XY of point `2` are computed by straightlines-intersection (LL) from the 5th values of the coordinates XY of point `1` , from the 2nd computed value of the azimuth t from point `1` to point `2` , from the given values of the coordinates XY (index missing → start values) of point `Q3` and from the 4th computed value of the azimuth t from point `2` to point `Q3` . The result is `not a number` , because due to gross errors in the start values no real section of the rays exists.

In the documentation of the computation procedure also the values discarded due to ⇑inappropriate sections are displayed.

☰ Outlier detection

If a sufficient number of values per computable quantity is obtained, it can be tried to detect outliers in the start values (coordinates and measurements). Below the result table a button [outlier detection] appears. If you click it, the results are analysed for outliers.

This outlier detection shows, what would happen, if a start value would be eliminated and the computation would be repeated. This would give less results, but the remaining results would be unchanged. As a consequence, for some results we would often get smaller ranges. An outlier is identified by a drastic reduction of such ranges. Note that for outlier detection the ranges **prior** to the ⇑prevention of inappropriate sections are the decisive factors.

The outlier detection creates a table with the following information per start value:

still computable quantities	It is possible that after elimination of a start value some quantities are no longer computable. The table shows, how many percent of the quantities are still computable. Start quantities, for which no second value can be computed anymore, are counted as no longer computable here. 100% means: After elimination all quantities can still be computed, at least once.
still computable values	After elimination of a start value not all values are still computable. The table shows the percentage of the values still computable. If this number is small then this start value is very important for a reliable solution and an elimination is questionable.
smaller ranges	The table shows the percentage of the ranges getting smaller after elimination of a start value. If this number is large then this start value is probably an outlier.
maximum reduction	The table shows the size of the maximum relative reduction of a range in percent. If this number is large then this start value is probably an outlier.
outlier probability	The probability, that a start value is an outlier, is rated. ⚠ Note that even if high outlier probability is assigned to many start values, this does not mean that all of them are outliers. It can well be only one of them.

If some start quantities are missing in the table, then they cannot be investigated for outliers due to small redundancy. If the total redundancy is small, the outlier detection fails sometimes. An error message is generated.

Now a start value to be eliminated can be selected (1st column of the outlier table). A candidate is pre-selected. Clicking the button [Eliminate start value] redisplays the results remaining after the selected start value has been eliminated. (The actual computation is not repeated.) Although an eliminated start value continues to appear in the list of known points or input measurements, it is not used anymore. If the redundancy is still sufficiently large, a renewed outlier detection can be requested, etc.

An application of the outlier detection can be found in the example ☆Trigonometric levelling line.

☰ ☆ Polar values computed from cartesian coordinates

The points of the penalty area of a football pitch 1,2,3,4,5,6 must be staked out by a tacheometer, which is set up at the corner points A,B of the pitch. The instrument heights are 1.42 m at A and 1.55 m at B. At the target points a stake out reflector of height 0.15 m is used. The azimuths t , slope distances s and zenith angles v of the six points to be staked out are desired.

First of all, we define a coordinate system, favorably a lefthanded cartesian system YXZ and determine the cartesian coordinates of all points in this system in metre. For example, as can be seen from the figure below, the coordinate X of point 6 is 75.00/2-7.32/2-5.50-11.00=17.34 and the coordinate y is 5.50+11.00=16.50. Heights are all set to 0.00. Now we set up the Coordinate lists .

The Measurement lists consists of two stations A and B. The station rows must contain the instrument heights 1.42 and 1.55. Since all target points have identical reflector heights 0.15, they can be specified by the default target height (default value of the target height) in the station row and omitted in the target rows. Then the target rows consist only of the target point names. The format selected for the empty columns is irrelevant, may be e.g. ''code / unused''.

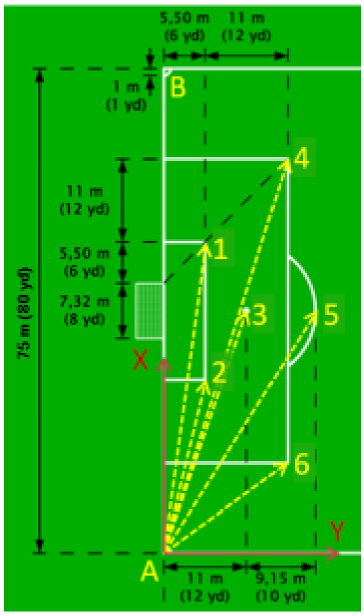
```
A 0   0 0
B 0 75 0

1   5.50 46.66 0
2   5.50 28.34 0
3 11.00 37.50 0
4 16.50 57.66 0
5 20.15 37.50 0
6 16.50 17.34 0
```

Coordinate lists
Type of system: Y X Z lefthanded
Column format: pointname coordinates

```
A 1.42 0.15
1
2    // This example
3    // shows how
4    // blind target
5    // points act.
6
~~~~ separation row
B 1.55 0.15
1
2
3
4
5
6
```

Measurement lists
Format station row: point name, instrument height, default target height
Format target row: point name



stakeout of a football pitch

Load example and “Compute”

The processing engine searches for computable quantities and finds height differences *dh* , horizontal distances *e* , slope distances *s* , azimuths *t* and zenith angles *v* , though only for the lines of sight. Nothing is computed between the target points among themselves. If e.g. the distance between 1 and 3 is desired then a further station 1 with target point 3 (blind target point) should be specified, or vice versa.

Moreover, we only get the azimuths from target to station here. This is because azimuths and horizonal distances are always given such that the two pointnames are in lexicographic order. Digits are prior ranking. Therefore, the azimuths must be changed by 200 grads. In order to get azimuths in sighting direction nonetheless, it is possible to rename the points, e.g. from 1 to P1 etc. Another possibility is to enforce the output of horizontal angles, which are always given from station to target. For this purpose, each station row must be augmented by the orientation angle 0.

Finally, after reasonable rounding, we obtain the following polar stakeout values in metres and grads:

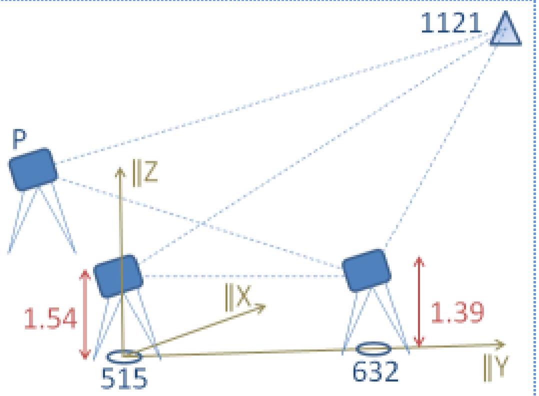
	s	t	v		s	t	v
A→1	47.000	7.470	101.720	B→1	28.903	187.797	103.085
A→2	28.897	12.203	102.799	B→2	47.004	192.530	101.896
A→3	39.101	18.165	102.068	B→3	39.105	181.835	102.280
A→4	59.988	17.743	101.348	B→4	23.977	151.580	103.719
A→5	42.590	31.390	101.899	B→5	42.594	168.610	102.093
A→6	23.970	48.420	103.375	B→6	59.991	182.257	101.486

If prefered, also other stakeout values can be taken from the result table, e.g. *e* or *dh* .

Inaccessible point with auxiliary triangles

An inaccessible point 1121 was measured from three tacheometry stations 515,632,P. The height of point 515 amounts to 107.483 m, the instrument height is here 1.54 m. The height of point 632 amounts to 107.832 m, the instrument height is here 1.39 m. The height of the auxiliary point P is unknown. An arbitrary value can be assigned to the instrument height of P, say 0.00 m. The target heights equal the instrument heights on the same point, because matching instruments and reflectors have been used. The inaccessible point 1121 was directly sighted without distance measurement, such that the corresponding target height equals 0.00 m. The height of point 1121 can be computed in both auxiliary triangles 515,632,1121 and P,632,1121. The measurement values are:

station point name	target point name	horizontal angle [grads]	zenith angle [grads]	slope distance [m]
515	1121	149.846	91.886	---
	632	187.807	99.991	952.233
632	515	260.607	100.018	952.233
	P	260.740	100.086	941.461
	1121	314.405	89.258	---
P	1121	28.449	91.684	---
	632	66.940	99.923	941.461



Inaccessible point with horizontal auxiliary triangles

Since pure levelling benchmarks are not supported, it is necessary to introduce a local coordinate system, favourably a cartesian lefthanded system XYZ. First of all we need the horizontal distance 515→632. For this aim, a simple computation using the Universal computer can be run, very much like in the preceeding example. The result is 952.233, equal to the slope distance of 515→632, because both points have nearly the same height. We define X_{515}=1000, Y_{515}=1000, X_{632}=1000, Y_{632}=1952.233. As a result, the defined Y-axis is parallel to the projection of 515→632 onto the horizontal plane. Now we set up the Coordinate lists and the Measurement lists . For the unmeasured distances to the inaccessible point 1121 we put 0. The same effect would have a negative or non-numerical value. An alternative notation of an unmeasured value can employ the ";;" method: Tabular data records.

```
515 1000      1000 107.483
632 1000 1952.233 107.832
```

```
515
1.54
1121 149.846  91.886   0     0
632  187.807  99.991 952.233
1.39
~~~~~~~~~~~~~~~~~~~~~~~~~
632
```

Coordinate lists
Type of system: XYZ lefthanded
Column format: pointname coordinates

Load example and "Compute"

Measurement lists
Format station row: point name, instrument height
Format target row: point name, horizontal angle, zenith angle, slope distance, target height

As a result, for the height Z of the inaccessible point 1121 no less than `41` different solutions are computed.The range is obtained by `8.4 mm` , which is convincing, the median amounts to `201.1106 m`.

The following should be noted:

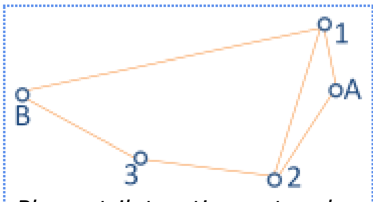
- For the height Z of 1121 as much as 84 different solutions have been computed, but 43 of them have been discarded during the ⇑ prevention of inappropriate sections . If you click on the 41 in column "values", all 84 values are shown, including computation procedure and intermediate values. The smallest and the largest value differ by no less than `0.17 m`.

- The instrument and target height of P, although being arbitrary, should not be omitted. Otherwise it cannot be supposed by the computation algorithm that both values are equal. In this case only 30 different solutions for the height Z of the inaccessible point 1121 would be obtained, 12 of them are discarded. However, the median would change only marginally.

The same example is also used for ☆ Vertical networks and may be computed. However, the results are not fully identical because Vertical networks computes a least squares adjustment.
The deviation in the final height of the inaccessible point `1121` amounts to `0.0012`.

☆ Planar trilateration network

This example shows how the Universal computer deals with ambiguous solutions. In the displayed trilateration network six horizontal distances have been measured. A and B are known points. There are exactly 2 solutions for the unknown points 1 and 2 (reflection with respect to axis AB) and exactly 4 solutions for the unknown point 3 (additional reflection with respect to axis 2B). Firstly, only one solution is computed and displayed, while a warning is issued, that there are 3 extra solutions. By means of the button [Next solution »] all the other solutions are displayed step by step.



Planar trilateration network

```
A        129.15  192.92
B        130.16  107.49
```

Coordinate lists
Type of system: X Y Z lefthanded
Column format: pointname coordinates

```
1
A        28.540  // In this example
B        94.147  // it is shown,
2        63.623  // how an ambiguous
####             // computation
2                // is treated.
A        35.714
```

Measurement lists
Format station row: point name
Format target row: point name, horizontal distance

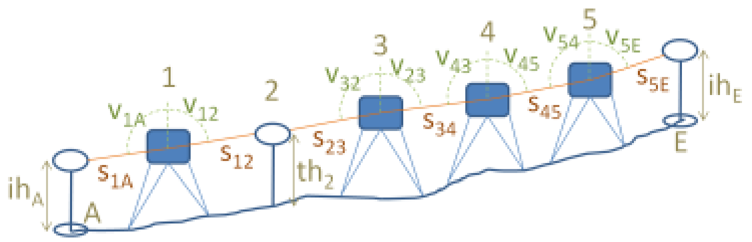
[Load example] and "Compute"

Note that arcs-intersections (AA) with bad section angles have been computed. In this example the ⇑ prevention of inappropriate sections is not effective, because there is no redundancy. Hence, this computation is unavoidable. If this is precluded then fewer solutions are obtained. Example: With the restriction to ''section angle between 15°=17 gon and 165°=183 gon'' only two solutions are obtained. Only point 1 is still computable. With the restriction to ''section angle between 75°=83 gon und 105°=117 gon'' also this point cannot be computed anymore.

Exercise: Augment the coordinate list by an arbitrary solution for point 3 and repeat the computation. See for yourself that only **one** solution remains for points 1 and 2.

☆ Trigonometric levelling line

This example shows, how pure height measurements can be processed. e.g. a trigonometric levelling line or a trigonometric levelling network. In the current version of the universal computer no pure height benchmarks without positions are supported. However, as long as these coordinates are not usable for any computation, their values are completely arbitrary. You may use approximate coordinates or simply "0 ; 0".



Trigonometric levelling line with a leap station at point 2

Pointnames and coordinates

Let us consider the displayed levelling line with a leap station at point 2, i.e. no instrument has been erected at this point and consequently at point 1 no reflector was needed. At the points 3,4,5 there was in turn a station with instrument and a reflector position. Heights are given for the levelling bechmarks A by $H=116.10$ and E by $H=123.06$. The reflector at A, 2 and E always has a target height of $th=1.40$. The points 1,3,4,5 are not marked, such that we refer the heights to the tilt axis of the instrument: $ih=0.00$. However, the reflection point is always 0.005 above the tilt axis, such that on the points 3,4,5 there is a target height of $th=0.005$ to consider. The obtained measurement values are shown in the measurement list at the right.

```
A 0 0 116.10 // Zeros are
E 0 0 141.38 // wildcards
```

Coordinate lists
Type of system: every possible
Column format: pointname coordinates

Pointnames and measurements

```
1 0.00
A 105.545 55.454 1.40
2  95.112 71.689 1.40
~~~~
3 0.00
2 103.230 49.528 1.40
4  92.751 65.666 0.005
~~~~
4 0.00
3 107.239 65.666 0.005
5  96.345 78.300 0.005
~~~~
5 0.00
4 103.645 78.300 0.005
E  99.300 45.650 1.40
```

Measurement lists
Format station row: point name, instrument height
Format target row: point name, zenith angle, slope distance, target height

Load example and “Compute”

The results in terms of the medians are given in the column `Height 1` of the table at the right. The ranges of the heights amount ot a maximum of `0.0029` and meet the expectations. (Note that the range is not to be confused with the standard deviation. The range is always much larger.)

If you want to get the classical levelling misclosure, delete either A or E in the coordinate list and re-compute. In the first case the result is `Z(E)=141.3795`, which corresponds to a levelling misclosure of `0.0005` .

The same example is also used for ☆ Vertical networks and may be computed. However, the results are not fully identical because Vertical networks computes a least squares adjustment.
The deviations in the final heights amount up to `0.0004`.

	Height 1	Height 2	Diff.
A	116.100	116.099	0.001
1	122.324	122.323	0.001
2	126.423	126.422	0.001
3	130.335	130.334	0.001
4	137.791	137.792	0.001
5	142.278	142.279	0.001
E	141.380	141.381	0.001

Height 1: original, Height 1: with implemented gross error

Now we observe, what happens, if we implement a **gross error** . We falsify the zenith angle `107.239` by 1 gon getting `108.239` and repeat the computation.

Load example and “Compute”

The results in terms of the medians are given in the column `Height 2` of the table at the right. The ranges of the heights amount ot a maximum of `1.06` and indicate a problem. However, the medians are almost unchanged, they change by no more than `0.001`! See column `Diff.` at the right. This shows clearly, how well the ⇑ robust estimation works. The reason is that less than half of the results use the falsified value. (However, this must not always be so.)

Finally, we test the ⇓ outlier detection . The outlier probability `very high` is assigned to the implemented outlier. All `37 (100%)` computed quantities are still computable after elimination, but only with 37% of their values. E.g. for the computed height of the new points we only get 5 values, previously up to 17. All ranges are decreased by the elimination, by up to 99%. The ranges decrease from `1.06` to `0.0033` . The medians are now again identical to the values before the falsification.

☆ Computation of the orthocentre of a triangle

From given vertex coordinates of a triangle ABC the coordinates of the **orthocentre** H (intersection of the altitudes) shall be computed. The solution can be obtained in a single run of the Universal computer .

The coordinate list consists of the given vertex coordinates, see right. We simulate angular measurements from the feet of the altitudes P,Q,R to all vertices and to the orthocentre H. The "measured" angles differ by 100 gon or 200 gon. One angle per station can be defined arbitrarily, here 0 gon is assigned to the rear vertex. (Note: For obtuse angled triangles the angles must be adapted.)

Load example and "Compute"

As results of the computation we obtain planar coordinates for all points P,Q,R,H. The computation can be performed even multiple times because one station in the measurement list is redundant. (Prove this by deleting an arbitrary station with the associated targets and repeating the computation.) However, the individual results for each quantity coincide, which can be verified by the ranges, because all altitudes intersect in a common point, which is demonstrated herewith for the case at hand.

By clicking on "3" in the column "values" you can retrace of the computation procedure . H is computed by straightlines-intersections (LL):

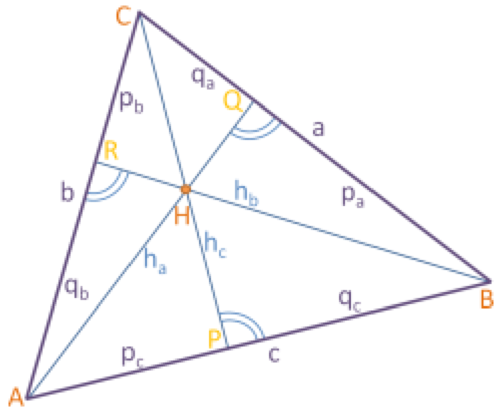
$$XY(H)_1 = LL(XY(Q)_1, t(H \rightarrow Q)1, XY(R)_1, t(H \rightarrow R)_1)$$
$$XY(H)_2 = LL(XY(P)_1, t(H \rightarrow P)1, XY(R)_1, t(H \rightarrow R)_1)$$
$$XY(H)_3 = LL(XY(P)_1, t(H \rightarrow P)1, XY(Q)_1, t(H \rightarrow Q)_1)$$

The segments of the altitudes can be taken from the result table, and the following products can be computed:

AH=3.950	HQ=2.686	AH·HQ=10.61
BH=6.786	HR=1.563	BH·HR=10.61
CH=3.924	HP=2.704	CH·HP=10.61

The equality of the products in the trailing column follows from a well-known law of geometry, which is used here as an additional check.

Exercise: Load the triangle ABC in Planar polygons and compute the barycentre of area M_2 and the centre of circumscribed circle M_3 . Now, check the well-known fact, that M_2, M_3 and H are collinear, i.e. lie on the same straight line called **Euler line** . Tip: The triangle M_2M_3H should have vanishing area.



Triangle ABC with orthocentre H

Pointnames and coordinates

```
A 14.02 17.11 // vertex
B 23.06 18.18 // coordinates
C 16.10 24.03
```

Coordinate lists
Type of system: YXZ lefthanded
Column format: pointname coordinates

Pointnames and measurements

```
P   // foot of altitude on AB
A 0
C 100
H 100  // orthocentre
B 200
~~~~~~~~~~~~~~~~~~~~~~~~~
Q   // foot of altitude on BC
B 0
A 100
H 100  // orthocentre
C 200
~~~~~~~~~~~~~~~~~~~~~~~~~
R   // foot of altitude on AC
C 0
B 100
H 100  // orthocentre
A 200  // angle unit = Grads
```

Measurement lists
Format station row: point name only
Format target row: point name, horizontal angle in Grads

***Did you know?** You must not sign in to IN DUBIO PRO GEO, therefore you work anonymously.*

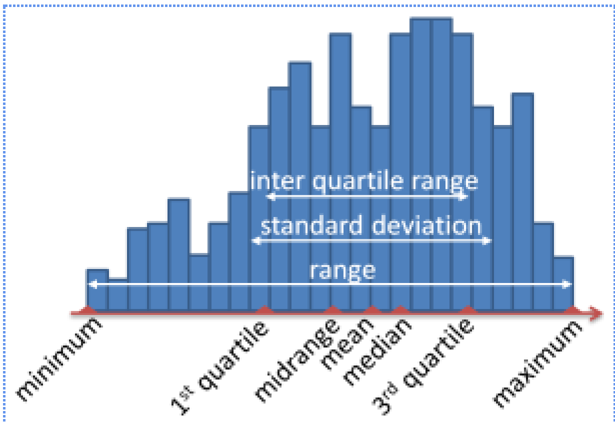
IN DUBIO PRO GEO Guide : Repeated measurements

Page contents

In geodesy and in other measuring professions a quantity is often measured multiple times under the same conditions, to increase accuracy and reliability. It is assumed that the measured values differ only due to the effect of random independent identically distributed errors. Such measurements are comprehensively evaluated, including all applicable statistical tests. Instead of geodetic measured values also other random samples of that kind may be evaluated.

Empirical measures of dispersion, skewness and excess kurtosis

The **inter quartile range** (IQR) is the difference of the 1st and 3rd quartile. This interval includes 50% of all values of a quantity. For the computation of the inter quartile range at least 12 measured values are required.



Histogram of the measured values

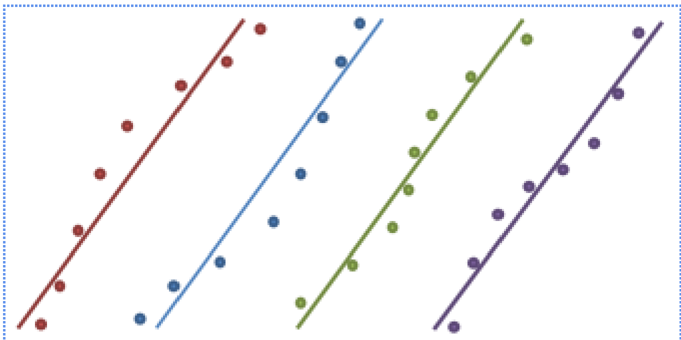
The **skewness** indicates, how symmetrical or non-symmetrical the measurement distribution is.
The **excess kurtosis** indicates, how thin or thick the tails of the measurement distribution, in comparison to the Gaussian normal distribution.

Skewness	distribution
< 0	left-skewed
= 0	symmetrical
> 0	right-skewed

excess kurtosis	distribution
< 0	thin-tailed (blunt peaked)
= 0	normal shape (Gaussian bell)
> 0	thick-tailed (sharp peaked)

Normal probability plot

This plot yields a fast graphical method to assess the deviation of a sample distribution from normality. The measured values are plotted on the horizontal axis against the normal order statistic medians on the vertical axis. In case of a Gaussian normal distribution all points in the plot should lie close to a straight line. Deviations of this may indicate non-symmetric (skewed) or thick-tailed, i.e. peak-shaped, or thin-tailed



Deviations from normal distribution (from left to right): right-skewed, left-skewed, thick-tailed, thin-tailed distributions. Isolated points at the left-bottom or right-top of the graphic are outlier-suspected.

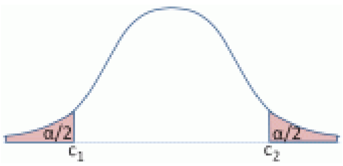
Explanations of the statistical tests

The measured values are treated as independent, identically Gaussian normal distributes random variates. If the measured values only approximately follow a normal distribution then some tests still yield correct results, provided that the number of measurements is not too small. Correlations would falsify the result.

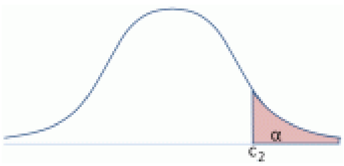
Did you know? *Although for a set of measurements all possible tests are computed consecutively, it would* ***not be correct*** *to use the results of two or more tests simultaneously without adapting the probability of type I decision error* α *. If you want to perform a* ***multiple*** *test with the same measurements, e.g. an outlier test and thereafter a test of the expectation then it has to be taken into account that the total probability of type I decision error* α *is increasing. In the simplest case of nearly independent multiple test statistics, the desired total probability must by divided by the number of individual tests according to the Bonferroni equation, to arrive at the probability* α *to be set for each individual test.*

All tests are computed, which are theoretically computable, even if a different test would less probably bring about a false decision. **Example:** If the standard deviation of the measurements is a priori correctly known then the w-Test after Baarda less probably brings about a false decision than the Pope-test and the Z-test less probably brings about a false decision than the t-test.

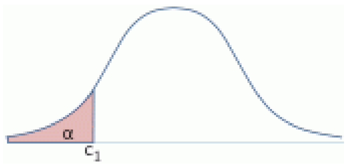
Acceptance region : The null hypothesis H_o is accepted if the test statistic



of the two-tailed test falls between the critical values c_1,c_2 .



of the right-tailed test falls below the critical value c_2 .



of the left-tailed test exceeds the critical value c_1 .

Test distributions

$N(\mu,\sigma)$	Gaussian normal distribution with expectation μ and standard deviation σ
$t(r)$	t-distribution r degrees of freedom
$\tau(r)$	τ-distribution r degrees of freedom
$\chi^2(r)$	χ²-distribution r degrees of freedom
$F(r,r')$	F-distribution r and r' degrees of freedom

Anderson-Darling-test for normality

This test checks the hypothesis of Gaussian normal distribution. The test statistic measures the difference between the empirical distribution of the sample and the normal distribution, giving more weights to the tails of the distribution than similar tests, e.g. the Cramér–von-Mises criterion.

The Anderson-Darling test is computed in up to four variants:

- with known expected value of the measurements μ_0 with known a priori standard deviation of measurements σ_0
- with known value σ_0
- with known value μ_0
- without restriction of the distribution

The first three variants work only if the required values have been given.

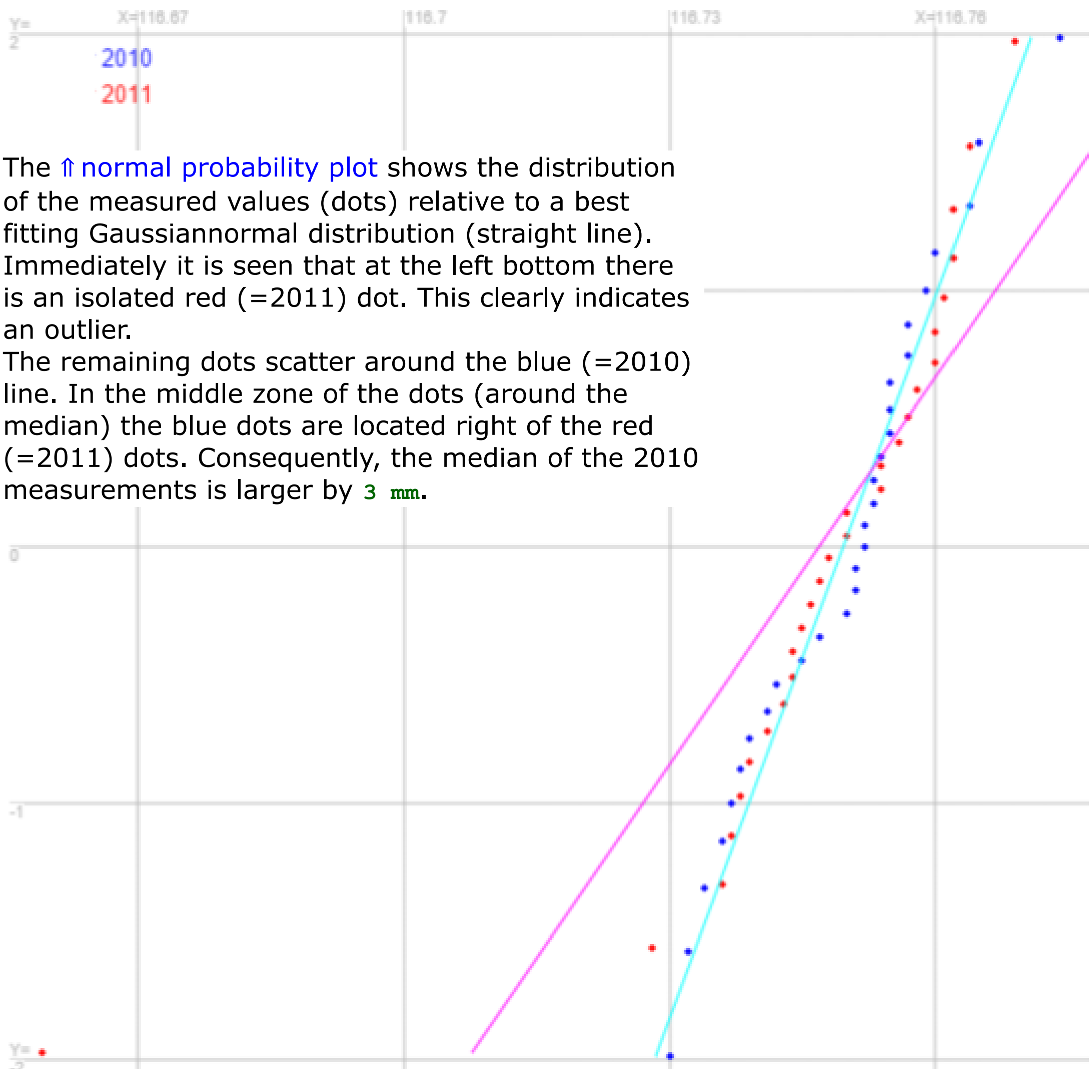
Repeated height determination of a point

Every year in the third semester of the curriculum Surveying/Geoinformatics of HTW HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES every student has to measure tacheometric point positionings of the same points. These can be treated as independent repeated measurements. In the years 2010 and 2011 the following results have been obtained for a chosen point:

Year	Determination of the point height, unit=metre
2010	116.774 116.755 116.755 116.751 116.742 116.745 116.760 116.754 116.753 116.739 116.752 116.747 116.732 116.752 116.736 116.764 116.738 116.765 116.757 116.750 116.741 116.759 116.751 116.753 116.734 116.737 116.757 116.730 116.755
2011	116.764 116.748 116.758 116.743 116.757 116.659 116.744 116.754 116.761 116.762 116.769 116.741 116.747 116.738 116.744 116.750 116.746 116.736 116.760 116.762 116.760 116.756 116.739 116.754 116.728 116.745 116.737 116.750

In the exercise network this point has the nominal height of `116.767 m` . Despite of lacking routine the students can be asked for a standard deviation of the determination of $\sigma_0 = 0.01\ m$. The repeated measurements shall be tested statistically with a probability of type I decision error of $\alpha = 0.05$.

Load example and "Compute"



The ⇑ normal probability plot shows the distribution of the measured values (dots) relative to a best fitting Gaussiannormal distribution (straight line). Immediately it is seen that at the left bottom there is an isolated red (=2011) dot. This clearly indicates an outlier.
The remaining dots scatter around the blue (=2010) line. In the middle zone of the dots (around the median) the blue dots are located right of the red (=2011) dots. Consequently, the median of the 2010 measurements is larger by `3 mm`.

First of all, it can be tested if the required measurement precision has been realized, i.e. if $H_0 : \sigma \leq \sigma_0$ can be assumed. This is done by the right-tailed **global test**, which for the measurement series `Year 2011` is rejected. This means that the measurement precision has probably not been realized. Strictly speaking, the probability, that the measurement precision has been realized and the global test is rejected nonetheless, is $\alpha=0.05$.

Also the **w-test** after Baarda detects an outlier for `Year 2011` . This is the measured value `116.659`, which is most departed from the mean. If you eliminate this value and repeat the computation, both the left-tailed global test as well as the w-test are accepted. However, it could be irritating that also the right-tailed global test $H_0{:}\sigma\geq\sigma_0$ is accepted. The reason for this phenomenon as follows: If α is chosen small enough, the null hypotheses of all statistical tests are always accepted. And for a small number of measurements, $\alpha=0.05$ is practically small.

If $\sigma_0=0.01$ would have been unknown then the outlier could have been detected by the **τ-test** after Pope.

The posterior standard deviations of a single determination of the point height are estimated by `0.0107 m` for `Year 2010` and `0.0102 m` for `Year 2011` . The answer to the question if after elimination of the outlier the measurement precision of both years should be treated as identical, is given by the two-tailed **F-test** with $H_0{:}\sigma_x=\sigma_y$. This hypothesis is accepted. Therefore, the precision in the `Year 2011` was not significantly higher.

The means amount to `116.7496 m` for `Year 2010` and `116.7501 m` for `Year 2011` . The answer of the question, if after elimination of the outlier the expected values of both series coincide, yields the **double-sample Z-test** with $H_0{:}\mu_x=\mu_y$. The hypothesis is accepted. Therefore, the expected values should be treated as identical.

The hypothesis that the nominal value of that point $\mu_0=116.767$ is identical to the mean, is tested by the two-tailed **single-sample Z-test** with $H_0{:}\mu=\mu_0$. This hypothesis is rejected for both years. The conclusion could be that the nominal value is not correct or that one control height used in all determinations of instrument heights was not correct and this has not been detected during station setup.

Since both measurement series do not show significant differences, it is possible to merge them into one series and repeat the computation:

Load example and "Compute"

As in the cases before, the nominal height value of $\mu_0=116.767$ is rejected. This value is either incorrect or the heights are biased. Moreover, the ⇑hypothesis of normal distribution with the parameters $\mu_0=116.767$ is rejected. Without this parameter the **Anderson-Darling-test** succeeds.

☰ ☆ Try this computation tool using normal distributed random numbers

Two series of Gaussian normal distributed pseudo random numbers $N(53.06;16.10)$ from ⇒www.random.org/gaussian-distributions are investigated.

Load example and "Compute"

☰ ☆ Try this computation tool using different random numbers

One series of Laplace distributed and one series of $\chi^2(1)$ distributed pseudo random numbers, 100 values each, computed by ⇒GNU Octave are investigated. The true parameters of the distributions are

series	expectation	median	stddev.	inter quartile range	skewness	excess kurtosis
Laplace	0	0	1.414	1.386	0	3
$\chi^2(1)$	1	0.455	1.414	1.222	2.828	12

Load example and “Compute”

Did you know? *All numbers are always accepted with decimal point or with decimal comma.*

IN DUBIO PRO GEO Guide : Double measurements

Page contents

Various quantities are all measured twice, possibly with different accuracies and/or with systematic difference. Such measurements are comprehensively evaluated, including all applicable statistical tests, e.g. whether the data show a systematic difference.

Introduction

In practical geodesy measurements are frequently carried out twice. Often a certain measurement configuration is prescribed for carrying out such measurements, like two different telescopic faces when measuring horizontal angles or two sessions in GNSS measurements. To be able to treat the results of two measurements L' and L'' as double measurements in the sense of geodetic adjustment, these measurements should not contain a systematic difference $d=E\{L''-L'\}$ or if so, this difference should be identical for all pairs of measurements. If this condition is fulfilled, then L' and L'' may also be other uncorrelated quantities, e.g. computed quantities like coordinates.

Measurements and accuracy measures

Measurement values are given as Measurement lists . See also the rules of tabular input. Each measurement record (row in a text area) contains a pair of measurements L' and L'' as well as maybe a name for the measured quantity L and corresponding standard deviations and weights in user-defined succession.

Standard deviations must be given in the same unit as the measured values. They can be specified separately for each measured values or jointly for both measured value of a pair. Instead of standard deviations it is possible to specify only weights. Here the unit is arbitrary. If some or all accuracy measures are missing, the are filled in by default values. If in the measurement format no column for an accuracy measure is selected, the default value is treated as a standard deviation. If neither accuracy measures in the text area nor default values are given, all weights are assumed to be equal to unity, where a warning is issued.

Incomplete measurement records or those with weight zero are ignored.

If standard deviations σ are given, they are transformed into weights by $p=1/\sigma^2$. Finally, each measurent value must be equipped with a weight.

Results

To begin with, the differences $L''-L'$ and by weight propagation also their weights are computed.

The processing is based on two different models:

without systematic difference of the measurements $d = 0$:
  The expectations of the measurement are equal, i.e. systematic measurement errors are assumed to be equal, e.g. they are excluded, i.e. they are identical to zero.

with systematic difference of the measurements $d \neq 0$:
  The expectations of the measurement are not equal.

In both models residuals and adjusted measurements are estimated. In the model without systematic difference adjusted measurements are simply the weighted means. In the model with systematic difference the adjusted value of the difference $\bar{d}$ is additionally estimated.

In both models a posteriori standard deviations $\bar{\sigma}$ for the original and adjusted measurements are estimated. In the model with systematic difference the a posteriori standard deviation $\bar{\sigma}_{\bar{d}}$ of the adjusted difference $\bar{d}$ is additionally estimated.

Optionally, more digits can be displayed for accuracy estimates.

Statistical tests

Distributions

If the measurements are only affected by uncorrelated normal distributed measurement errors, then the differences of the measurements are a statistical sample of a random variable from a known family of distribution.

given	**standard deviations**		**only weights**
weights/stddev.	all equal	different	
to be investigated	differences $\Delta L = L'' - L'$	normalised differences $\Delta L / \sigma_{\Delta L}$	studentised differences $\Delta L / \bar{\sigma}_{\Delta L}$
distribution of the ΔL	normal distribution		t distribution
if $d = 0$	$N(0, \sigma^2)$	$N(0,1)$	$t(r-1)$

d = true systematic difference σ = standard deviation of the measurements

r = total redundancy $\sigma_{\Delta L}$ = standard deviations of the differences ΔL

In this way it is possible to test statistical hypotheses. The following tests are possible with the computation tool Repeated measurements applied to those (normalized/studentized) differences:

Test for normal distribution

The Anderson-Darling-test for normality tests for the (normalised) differences the hypothesis of normal distribution. If parameters of the distribution are a priori known, they are used in the test.

Global test

This test is only possible if standard deviations of the measurements are given. Then it is possible to test statistically, if the a posteriori standard deviation is identically to the a priori value.

Outlier tests

Here it is tested statistically, if the measurements are free of outliers. If standard deviations of the measurements are given, the outlier test of Baarda (w test) can be used. If only weights are given, the outlier test of Pope (τ test) should be chosen.

Test of significance $d \neq 0$

Here it is tested statistically, if the measurements show no systematic difference, i.e. $d = 0$ can be assumed. If standard deviations of the measurements are given, the single sample Gauss test (z test) is suited. If only weights are given, the single sample t test should be chosen.

Double determination of point heights by fast static GNSS measurements

For 8 points with a time separation of 10 years, heights are determined by fast static GNSS measurements. The a priori standard deviations in the year 2009 are all assumed to be 30 mm while the a priori standard deviations in the year 2019 are all expected to be only 20 mm .

Point	height 2009	height 2019
62-x-81	115.232 m	115.252 m
62-x-82	113.345 m	113.357 m
62-x-83	113.203 m	113.215 m
62-x-84	117.232 m	117.230 m
62-x-85	119.733 m	119.720 m
62-x-86	112.400 m	112.434 m
62-x-87	114.220 m	114.206 m
62-x-88	114.004 m	114.009 m

Load example and "Compute"

If a systematic difference by **uniform height change in the entire area is excluded**, then the adjusted heights in the second column of the following table is obtained. The a posteriori standard deviation of these heights is 7.7 mm.

If a systematic difference by **uniform height change in the entire area is assumed** then for this difference the adjusted value 6.7 mm is obtained. The adjusted heights in the third and fourth column of the following table is obtained. The a posteriori standard deviation of these heights is 8.5 mm für 2009 und 7.8 mm for 2019.The a posteriori standard deviation of the difference is estimated to 5.9 mm .

	$d = 0$	$d \neq 0$	
Point	**height**	**height 2009**	**height 2019**
62-x-81	115.246 m	115.241 m	115.248 m
62-x-82	113.353 m	113.349 m	113.355 m
62-x-83	113.211 m	113.207 m	113.213 m
62-x-84	117.231 m	117.226 m	117.233 m
62-x-85	119.724 m	119.719 m	119.726 m
62-x-86	112.424 m	112.419 m	112.426 m
62-x-87	114.210 m	114.206 m	114.212 m
62-x-88	114.007 m	114.003 m	114.010 m

To decide, if the hypothesis $d = 0$ can be accepted, a statistical hypothesis test is required, in this case a two-tailed single-sample Z-test of the normally distributed sample of the height differences. For this purpose, the height differences must be transferred to Repeated measurements and computed. The test statistic evaluates to $z = 0.53$. With a probability of type I decision error $\alpha = 0.05 = 5\%$ the critical values are $-1.96;1.96$. Therefore the null hypothesis $d = 0$ is accepted.

Did you know? *You may get all numbers either with decimal point or with decimal comma.*

IN DUBIO PRO GEO Guide : Adjustment with observation equations

Page contents

The general adjustment problem with observation equations is solved in the least squares sense, optionally with constraints for parameters and functions of adjusted quantities.

Introduction

The (linearised) residual equations $l + v = A\,x$ are solved in the **least squares sense**: $v^T P\,v = min!$ also known as **adjustment with observation equations** (Gauss-Markov model):

- l is the known n-vector of (linearised) observations
- A is the known $n{\times}u$-design matrix of the adjustment model
- P is the known $n{\times}n$-weight matrix of the observations (here diagonal)
- x is the unknown u-vector of (linearised) parameters
- v is the unknown n-vector of residuals

Optionally you may specify m (linearised) **constraints** $B^T x = b$, which are fulfilled by the true parameters x :

- b is the known m-vector of (linearised) constraints,
- B is the known $u{\times}m$-matrix of (linearised) constraints.

An adjustment problem only arises if $n + m > u$ holds. At the moment only

- uncorrelated observations with P being a diagonal matrix and
- regular adjustment problems with A and B having full rank,

are supported. Overlined quantities symbolise estimates, i.e. either adjusted quantities $\bar{x}, \bar{l}$ or posterior standard deviations $\bar{\sigma}$.

The linear adjustment model

At least the vector of observations l and the design matrix A must be given. Their number of rows must be equal. Optionally, names can be assigned to the observations and to the parameters. They will appear in the result tables.

For all observations an a priori standard deviation σ or a weight p can be specified. In the first case IN DUBIO PRO GEO computes weights for the adjustment by $1/\sigma^2$. Standard deviations have the same unit as the observations and must not be negative. Weights must be positive. If only one value is given, it is assigned to all observations. If no value is given, unit weights are assigned to all observations.

As **main results** of the adjustment, you obtain

- adjusted parameters $\bar{x}$
- adjusted observations $\bar{l}$
- for both vectors the corresponding a posteriori standard deviations $\bar{\sigma}$
- for both vectors the corresponding a priori standard deviations σ , but only, if those have been given for the observations
- for both vectors the corresponding cofactor matrices $Q_{\bar{x}}, Q_{\bar{l}}$
- redundancy parts r of all observations
- redundancy matrix R

Linear constraints for parameters

Optionally, a constraints matrix B and a constraints vector b can be given. The number of rows and columns of B must match the dimensions of b and x . Alternatively, the transposed matrix B^T can be given.

In this case the adjusted parameters $\bar{x}$ are computed such that they also fulfill these linear constraints:

$$B^T \bar{x} = b$$

Linear functions of adjusted quantities

Oftentimes, you are not directly interested in the adjusted quantities $\bar{x}$ or $\bar{l}$, but they are mere auxiliary quantities on the way to the quantities f actually desired. Therefore, you compute these quantities as **functions** of adjusted quantities. If these functions are **linear** in the form $f = F\bar{x}$ or $f = F\bar{l}$, then you directly specify the matrix F and obtains f as a result of the computations. For f a posteriori standard deviations $\bar{\sigma}_f$ and also priori standard deviations σ_f are computed, but the latter only, if those have been given for the observations.

Optionally, names can be assigned to the functions. They will appear in the result table.

If an absolute term f_o is added, such that $f = f_o + F\bar{x}$ or $f = f_o + F\bar{l}$, then the standard deviations do not change. f_o may be added manually.

Statistical tests

Statistical tests are only computed, if a probability of type I decision error α has been given. The smaller this value, the more rarely the null hypothesis is rejected in the tests.

The **global test** checks if a priori and a posteriori standard deviations are in good agreement. It is therefore only applicable, if both values could be computed. This test detects some possible model misspecifications.

For the localisation of outliers the **τ test** of Pope or the **w test** of Baarda can be used, but the latter only if a priori standard deviations have been given for all observations. In this case it is preferable. The test statistic for one outlier is either $max|SV|$ or $max|NV|$. If this exceeds the corresponding critical value, an outlier is detected and localised in the observation, where the maximum is assumed.

Moreover, Information criteria (AIC, AICc, BIC) are computed for the adjustment model.

Linearisation

For **nonlinear** adjustment models $L+v=\varphi(X)$ with non-linear functions φ a linearisation is required, where starting from an approximate parameter vector X^0 we set

$$x := X - X^0, \quad l := L\text{-}\varphi(X^0)$$

and the matrix A consists of all partial derivatives $\partial\varphi/\partial X$ computed at the location $X=X^0$. In mathematics such a matrix is called Jacobian matrix. Finally, the adjusted quantities are computed as

$$\overline{X} = X^0 + \overline{x}, \quad \overline{L} = L + v$$

The final control condition $\overline{L} = \varphi(\overline{X})$ should be satisfied, otherwise the approximate parameters were too bad and the adjustment has to be repeated with $X^0 := \overline{X}$. The linearisation has to be carried out **manually** . The standard deviations of original and linearised parameters $\overline{X}, \overline{x}$ and of original and linearised observations $\overline{L}, \overline{l}$ are identical.

If **nonlinear** constraints $\beta(X) = 0$ are given, they must be linearised too:

$$\beta(X^0+x) = \beta(X^0) + B^T x = 0, \quad b = -\beta(X^0)$$

and the matrix B^T consists of partial derivatives $\partial\beta/\partial X$ computed at the location $X = X^0$.

If **nonlinear** functions $f = \psi(\overline{X})$ are given, they must be linearised too:

$$\psi(X^0 + \overline{x}) = \psi(X^0) + F\overline{x} = 0, \quad f_o = \psi(X^0)$$

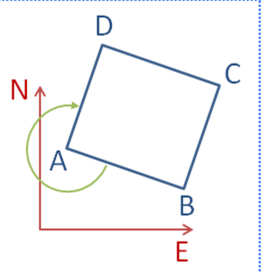
and the matrix F consists of partial derivatives $\partial\psi/\partial X$ computed at the location $X = X^0$. As before, the standard deviations can be obtained directly from the computation. The functional values must be computed manually by $\psi(\overline{X})$ or $\psi(X_o) + f$. These values must be identical apart from linearisation errors (final control condition). The same procedure applies to functions $f = \psi(\overline{L})$.

For more details on geodetic adjustment please seek advice from the Adjustment textbooks.

☆ Adjusting square through four vertices

The vertices of a planar square have been measured (E=East, N=North):

	E [m]	N [m]
A	17.11	14.02
B	39.37	8.26
C	45.13	30.53
D	22.80	36.30



Adjusting square through four vertices

Due to small measurement errors these points do not exactly form a square. These errors must be adjusted in the sense of least squares of residuals $v^T P v = min!$ for the observations. Moreover, the length of a side a and the area F of the adjusted square with accuracy estimates are required.

Observations L are the $n = 8$ measured coordinates in the chosen succession $E_A, N_A, E_B, \ldots, N_D$ All observation standard deviations σ_l are a priori assumed as `0.01 m` .

Correlations between these observations must be neglected because they are unknown.

Tip: See for yourself by the computation of ABCD in Planar polygons, that ABCD is approximately square:

Load observed polygon and "Compute"

A square is determined uniquely by $u = 4$ **parameters** . We choose the coordinates of the points A and B as parameters. To discriminate between observations and parameter, the latter are symbolised by small letters e_A, n_A, e_B, n_B and this succession is chosen. To complete the points A and B described by these parameters to an exact square we rotate the vector AB by `300 gon = 270°` (⇑ figure) and stake out points C and D from B and A by this vector. (A planar vector is rotated by `300 gon = 270°` by interchanging both components and changing the sign of the new east component.) Hence, the square with the parameters e_A, n_A, e_B, n_B has the following additional vertex coordinates:

$$e_C = e_B - (n_B - n_A) \qquad n_C = n_B + (e_B - e_A) \qquad e_D = e_A - (n_B - n_A) \qquad n_D = n_A + (e_B - e_A)$$

Therefore, we come up the following **functional adjustment model** $L + v = \varphi(X)$:

$$E_A + v_{EA} = e_A \qquad E_B + v_{EB} = e_B \qquad E_C + v_{EC} = e_B - (n_B - n_A) \qquad E_D + v_{ED} = e_A - (n_B - n_A)$$
$$N_A + v_{NA} = n_A \qquad N_B + v_{NB} = n_B \qquad N_C + v_{NC} = n_B + (e_B - e_A) \qquad N_D + v_{ND} = n_A + (e_B - e_A)$$

Since φ is a linear function, no **linearisation** is immediately required and we could identify the derived system of observation equations with $l + v = A\,x$. However, the side length a is a non-linear function of the parameters, and for the computation of the accuracy of a a linearisation is finally required nonetheless. We use the observed coordinates of points A and B as approximate parameters X^0 . The linearised observations $l = L - \varphi(X^0)$ read therefore:

```
       E [m]                                 N [m]
A      17.11- 17.11               = 0.00     14.02- 14.02                =0.00
B      39.37- 39.37               = 0.00      8.26-  8.26                =0.00
C      45.13-(39.37-8.26+14.02)= 0.00        30.53-( 8.26+39.37-17.11)=0.01
D      22.80-(17.11-8.26+14.02)=-0.07        36.30-(14.02+39.37-17.11)=0.02
```

Hence, the design matrix A and the vector of linearised observations l read:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & -1 \\ -1 & 0 & 1 & 1 \\ 1 & 1 & 0 & -1 \\ -1 & 1 & 1 & 0 \end{pmatrix} \qquad l = \begin{pmatrix} 0.00 \\ 0.00 \\ 0.00 \\ 0.00 \\ 0.00 \\ 0.01 \\ -0.07 \\ 0.02 \end{pmatrix}$$

Two **functions of adjusted parameters**

$$a=[(e_A-e_B)^2+(n_A-n_B)^2]^{1/2}, \; F= (e_A-e_B)^2+(n_A-n_B)^2$$

need to be computed. From these functions (in this succession) we derive the following derivatives, which constitute the Jacobian matrix

$$F = \begin{pmatrix} \frac{17.11-39.37}{22.993} & \frac{14.02-8.26}{22.993} & \frac{39.37-17.11}{22.993} & \frac{8.26-14.02}{22.993} \\ 2(17.11-39.37) & 2(14.02-8.26) & 2(39.37-17.11) & 2(8.26-14.02) \end{pmatrix} = \begin{pmatrix} -44.52 & 11.52 & 44.52 & -11.52 \\ -0.9681 & 0.2505 & 0.9681 & -0.2505 \end{pmatrix}$$

Together with the $n = 8$ prior standard deviations of the observations, which all amount to `0.01` , the input quantities are complete. As a probability of type I decision error we choose `0.01`.

Load example and "Compute"

First of all, we realise that the **global test** (overall model test) is rejected. This means that the prior and posterior accuracies are significantly different. All adjusted coordinates have the same standard deviation of $\sigma_{\bar{l}} = 7.1\ mm$ or rather $\overline{\sigma_{\bar{l}}} = 16.8\ mm$. (The accuracies of original and linearised quantities are always the same.) All redundancy parts amount to $r = 0.5 = 50\%$, which proves that all observations are well controllable. A normalised residual $NV = 4.6$ causes a rejection of the null hypothesis of the **w test**. The observation E_D could accordingly be grossly erroneous and the adjustment could be repeated without this observation. All studentised residuals SV are below their critical value. Therefore, it is also possible that the prior accuracy assumption was overly optimistic.

By means of the residuals $v = A\,\bar{x} - l$ we compute the adjusted observations $\bar{L} = L + v$ and thereby the **final coordinates of the vertices**:

	E [m]	N [m]
A	17.11-0.0225=17.0875	14.02-0.0125=14.0075
B	39.37+0.0025=39.3725	8.26+0.0025= 8.2625
C	45.13-0.0125=45.1175	30.53+0.0175=30.5475
D	22.80+0.0375=22.8325	36.30-0.0075=36.2925

We confirm by the computation of ABCD with final coordinates in Planar polygons , that ABCD is exactly square. This is immediately the final control condition.

Load final control condition and click "Compute"

The adjusted side length and the adjusted area of the square we find directly from the final control computation above $\bar{a} = 23.01361$ und $\bar{F} = 529.626.$ The same result we obtain by the size of the approximate square, which is defined by the approximate parameters,

$$F^0=(17.11-39.37)^2+(14.02-8.26)^2=528.685, \quad a^0 = \sqrt{528.685}=22.9932$$

plus linearised function values $f = F\,\bar{x}$, for which the adjustment gives the values 0.0204 und 0.940 .

The accuracies of $\bar{a}$ coincide with those of the adjusted coordinates because the cofactors are the same. The posterior standard deviation of $\bar{F}$ assumes the value $\overline{\sigma_{F}} = 0.77\ m^2$.

Exercise: Eliminate the observation E_D by erasing the 7th row of the matrix A , erasing the 7th linearised observation and the 7th observation standard deviation. Now repeat the adjustment. It turns out that all null hypotheses are now accepted. All posterior standard deviations assume considerably smaller values, e.g. for the adjusted area 0.23 m². However, this approach is disputable because is would be more consequential to fully erase the point D. (Why should only one coordinate of D be grossly erroneous?). But the statistical test for the decision of this alternative hypothesis you have to do manually.

☝ Loading adjustment models from other tools

Vertical networks and Sets of angles and distances can be re-adjusted with Adjustment with observation equations . This yields the following advantages:

- Weights can be changes. E.g. target points of low sighting accuracy can be downweighted.
- Outliers can be automatically detected by w- or τ-test.

- The accuracy may be tested vs. a theoretical value. For example, it may be tested statistically, if the accuracy specification of the manufacturer of the instrument is met.
- The redundancy parts, full cofactor matrices and other interesting values are displayed.
- For many values more digits are displayed, if desired.

Forthcoming: More tools will provide this option.

Did you know? *The library contains ~2500 scientific documents regarding all subjects of Geodesy.*

IN DUBIO PRO GEO Guide : Vertical networks

Page contents

All kinds of vertical networks are adjusted, both spirit levelling networks (height differences & lengths of lines) as well as trigonometric vertical networks (zenith angles & distances), both free as well as connected networks, even single levelling lines.

Known points

For a **connected vertical network** specify the list of points with known heights. The input of the two to four **columns** complies with the general rules of tabular input. The pointnames may be chosen as in the Coordinate lists . Coordinate lists may be transfered from other computation tools.

If heights should be subject to adjustment, please specify an accuracy measure, either a standard deviation or a weight. If a default value is given for the accuracy measure, it pads all missing values in this column. Otherwise all heights without accuracy measure are treated as error-free in the adjustment. This is equivalent to zero standard deviation. A list may consist of both adjustable points and fixed points.

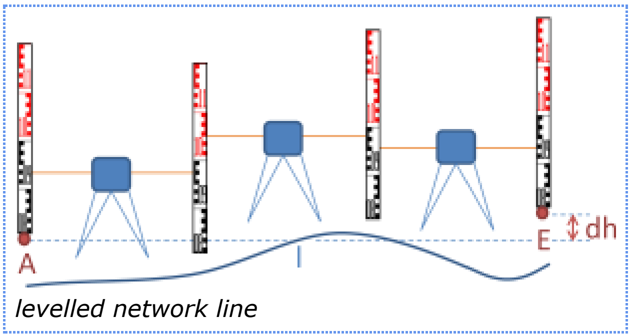
Points not present in the network are ignored. Points without height and network points missing in the list are treated as new points. A **free vertical network** consists exclusively of new points, e.g., if the list of known points is empty.

If you specify coordinates X and Y then the vertical network is displayed on the canvas. Points without X and Y are not displayed. These coordinates are not used for network adjustment. If new points should be displayed on the canvas, then you must you specify coordinates X and Y also for these points.

Measurements

Measurement values are given as Measurement lists . See also the rules of tabular input. Each measurement record starts with two pointnames, i.e. startpointname and endpointname of the network line. They are followed by up to five measurement values or accuracy measures in user-defined succession.

specific for spirit levelling networks

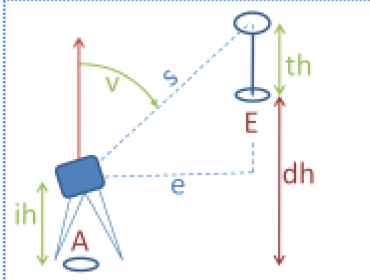


Height difference dh ***(required)***
: is the difference between endpoint and startpoint height of a line. If from startpoint to endpoint it goes uphill, height differences are specified positive, otherwise negative or zero.

Length of levelling line l ***(optional)***

specifies the approximate length of a levelling line and may be used to define weights. The values must not be negative.

specific for trigonometric vertical networks



trigonometric network line AE

***Zenith angle** v*
***Horizontal distance** e **or slope distance** s*
***Instrument height** ih*
***Target height** th*

have the same meaning as in the station and target rows.

All values are always required. Missing values ih, th are padded by the default values, if any.

for both kinds of vertical networks

Standard deviation (a priori)** σ_{dh} **or weight** p_{dh} **of the height difference

are used in the adjustment for the stochastic model. Only one of both values can be present. A default value pads all missing values in this column, or all values. Otherwise a missing accuracy measure is treated as zero standard deviation or infinite weight, which means that in the adjustment the related height difference acts as a constraint.
⚠ Also for trigonometric vertical networks these values refer to the adjustable height difference $dh = s \cdot \cot(v) + ih - th$, i.e. they include measurement errors in the instrument height and target height.

Code (optional)

is currently ignored.

☰ Multiple measured network lines

Some or all network lines may be measured **twice** , i.e.

1. by spirit levelling in forward direction AE and backward direction EA
2. by trigonometric measurement in A in two telescopic faces
3. by trigonometric measurement in sight AE and backsight EA

In this case you simply specify **two columns** for height difference dh or the zenith angle v . The other quantities in the measurement record must not appear twice here. For the third case the following is assumed: The values of the instrument height ih and the target height th are only specified for AE and in the backsight EA they must be exactly interchanged. If this condition is not met, please specify means or use the method described below.

It is not necessary to change the sign of the height difference in backward direction EA. For the height difference in forward direction AE the sign of the first value is assumed. Therefore, two forward directions could have been measured. Your case is recognised by the signs of the first pair of values. (Theoretically it may happen that the height differences of the first pair are so small, that the recognition fails. In the extreme case they may be zero. Then a different record of the measurement list should be shifted to the front.)

The network lines measured twice may also be specified in two separate measurement records. This practice allows greater flexibility. In the same way you may specify network lines measured more than twice: each measurement in a separate record.

Examples: The following inputs yield identical results.

geometric network	**trigonometric network** (Angle unit: Grad)
`// from to dh1 dh2 l pdh` `A E 16.10 16.11 2306 1`	`// from to v1 v2 e ih th pdh` `A E 92.1402 92.1418 88.2306 1.54 1.66 1`
`// from to dh1 dh2 l pdh` `A E 16.10 -16.11 2306 1`	`// from to v1 v2 e ih th pdh` `A E 92.1402 107.8582 88.2306 1.54 1.66 1`

`// from to dh l pdh`
`A E 16.10 2306 1`
`E A -16.11 2306 1`
	`// from to v e ih th pdh`
	`A E 92.1402 88.2306 1.54 1.66 1`
	`E A 107.8582 88.2306 1.66 1.54 1`

Multiple measured network lines enter the adjustment as **separate observations** , which is generally recommended. If you want to let them enter as mean values, please get them yourself. In the examples above the records could read:

geometric network	trigonometric network
`// from to dh l pdh` `A E (16.10+16.11)/2 2306 1`	`// from to v e ih th pdh` `A E (92.1402+92.1418)/2 88.2306 1.54 1.66 2`

Take care not to specify the multiple measured network lines as a constraint, otherwise there wil be a contradiction of the probably different values.

Stochastic model of the adjustment

Here observations of the adjustment are the given height differences dh or the height differences computed from zenith angles v and distances e or s as well as the heights of known adjustable points. Accuracy measures are needed for all of them.

⚠Also for trigonometric vertical networks these accuracy measures refer to the adjustable height difference dh , i.e. they include measurement errors in the instrument height and target height.

Correlated observations are not supported at the moment.

There are multiple opportunities, how to **define the stochastic model** via the selection of column format:

Weights of height differences "as given above or from standard deviations or all unity"

no column for standard deviation or weight selected

The default value for standard deviation or weight is treated as standard deviation σ for all observations. If the default value is missing, all heights of known points are treated as fixed and unit weight is assigned all height differences.

column for standard deviation selected

For all observations please specify a priori standard deviation σ_{dh} and/or σ_H . If a default value is given, it pads all missing values in this column. Otherwise, Points and height differences without standard deviation are treated as fixed, just as having standard deviation zero or weight ''INF''. Standard deviations have the same unit as heights and height differences and must not be negative. IN DUBIO PRO GEO computes weights for the adjustment by $1/\sigma^2$.

column for weight selected

For all observations please specify a weight p_{dh} and/or p_H . If a default value is given, it pads all missing values in this column. Otherwise, points and height differences without weight are treated as fixed, just as having infinite weight. Weights must not be negative.

combination of columns for standard deviation and weight selected

Example: Weights are specified for height differences, but standard deviation are specified for heights.
Here a variance component estimation is required. Unfortunately, this case is not yet implemented. Therefore, this selection is not yet recommended.

Weights of height differences derived from lengths of lines or distances

For the definition of weights for height differences you can indirectly use

- or the lengths of lines l (spirit levelling network), or
- the distances e or s (trigonometric vertical network).

Then, for each network line please specify such a value. For this purpose you select the respective column format. You can specify how the weights are derived from distances or lengths of lines. Each height difference without length of line is treated as fixed, just as having standard deviation zero or weight ''INF''. Lengths of lines must not be specified in the same unit as the heights and height differences. Distances must have the same unit as heights and height differences. All values must be positive.

If you selected such an option and in addition you specify explicite standard deviations or weights then you obtain an error message.

Weights of heights

For heights of known, movable points you specify a standard deviation in the same unit as the heights or a weight, as described above. Heights without accuracy measure or with standard deviation zero or weight "INF" are treated as error-free and fixed in the adjustment.

⚠ The weights of heights and height differences must agree. A variance component estimation is not yet implemented.

Height differences as constraints

If zero standard deviations or weight "INF" is assigned to a height difference then they act as constraints. It may happen that these constraints contradict. This case arises if these height differences

- have been measured multiple times and the numerical values are not identical, or
- have been measured between known fixed points, or
- form a closed loop with misclosure.

These cases cause an error. Use height differences as constraints with care.

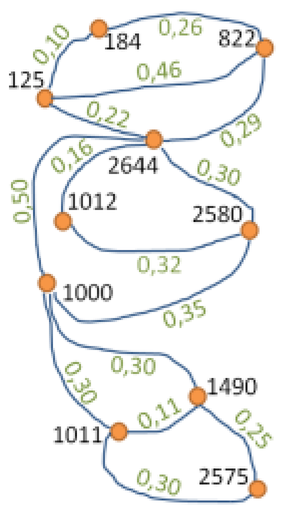
The computation is performed such that either the startpoint or the endpoint of each such line is eliminated from the list of adjustable points and is added after the adjustment using the constraint.

Results of the adjustment

The adjustment is performed according to the method of least squares. Free vertical networks are adjusted with the datum constraint that the sum of all adjusted heights is equal to zero.

The documented results contain the residuals, the adjusted values, the corresponding a posteriori standard deviations and the redundancy parts.

If a priori standard deviations are given for all heights and height differences (e.g. zero), then you compute the a priori standard deviations of the adjusted values by dividing the corresponding a posteriori values by $\overline{\sigma}_0$. You find such a case in the ☆Trigonometric levelling line.



HTW campus subnetwork - topology

☆ Campus subnetwork of HTW Dresden as a free spirit levelling network

In the third semester of the curriculum Geomatics of the HTW HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES parts of the campus vertical network are measured by spirit levelling. As a result the height differences dh in metres and lengths of the lines l in kilometres given below have been obtained. Each levelling line has been measured multiple times, here between two and six times. These observations should be adjusted as a free vertical network.

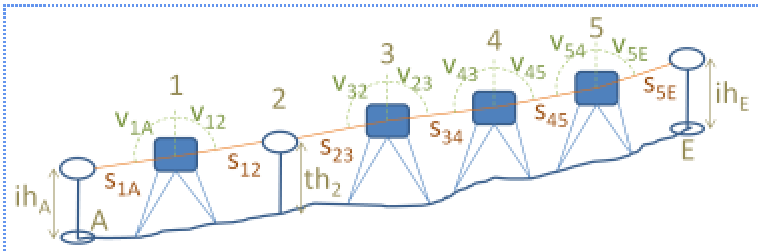
As a result for a single line of length 1 km (unit weight) we obtain a posteriori standard deviation of `0.47 mm` . All partial redundancies are above `0.7` , such that the network enjoys sufficient reliability.

All residuals amount to less than `0.5 mm` . For the point heights this yields a posteriori standard deviations up to a maximum of `0.15 mm`.

from	to	Δh [m]	l [km]	from	to	Δh [m]	l [km]	from	to	Δh [m]	l [km]
2580	2644	-0.05638	0.37	1011	2575	9.70155	0.29	822	2644	-2.91135	0.28
2580	2644	-0.05592	0.28	1000	1011	8.80803	0.29	822	2644	-2.91116	0.30
2580	2644	-0.05659	0.37	1000	1011	8.80839	0.30	184	822	3.11445	0.25
2580	2644	-0.05609	0.27	1000	1011	8.80838	0.33	184	822	3.11462	0.27
1490	2575	9.79288	0.25	1000	1011	8.80852	0.29	184	822	3.11540	0.25
1490	2575	9.79311	0.26	1000	1490	8.71692	0.28	184	822	3.11474	0.26
1012	2580	0.60757	0.32	1000	1490	8.71712	0.29	125	184	0.61625	0.11
1012	2580	0.60767	0.31	1000	1490	8.71700	0.28	125	184	0.61652	0.10
1012	2580	0.60785	0.32	1000	1490	8.71724	0.33	125	184	0.61656	0.10
1012	2580	0.60800	0.31	1000	2580	-0.60227	0.32	125	184	0.61632	0.11
1012	2644	0.55162	0.16	1000	2580	-0.60220	0.38	125	184	0.61656	0.10
1012	2644	0.55167	0.16	1000	2580	-0.60257	0.29	125	184	0.61669	0.10
1012	2644	0.55163	0.16	1000	2580	-0.60284	0.36	125	822	3.73164	0.46
1012	2644	0.55177	0.16	1000	2644	-0.65811	0.47	125	822	3.73182	0.46
1011	1490	-0.09163	0.10	1000	2644	-0.65810	0.51	125	2644	0.82011	0.21
1011	1490	-0.09118	0.12	1000	2644	-0.65860	0.49	125	2644	0.82042	0.22
1011	1490	-0.09169	0.10	822	2644	-2.91118	0.28	125	2644	0.82039	0.28
1011	1490	-0.09140	0.12	822	2644	-2.91111	0.30	125	2644	0.82062	0.22
1011	2575	9.70153	0.31								

Load example and "Compute"

Trigonometric levelling line



Trigonometric levelling line with a leap station at point 2

Let us consider the displayed levelling line with a leap station at point 2, i.e. no instrument has been erected at this point and consequently at point 1 no reflector was needed. At the points 3,4,5 there was in turn a station with instrument and a reflector position. Heights are given for the levelling bechmarks A by $H=116.10$ and E by $H=123.06$. The reflector at A, 2 and E always has a target height of $th=1.40$. The points 1,3,4,5 are not marked, such that we refer the heights to the tilt axis of the instrument: $ih=0.00$. However, the reflection point is always 0.005 above the tilt axis, such that on the points 3,4,5 there is a target height of $th=0.005$ to consider. The obtained measurement values are shown at the right.

from	to	height difference	slope distance	target height
1	A	105.545	55.454	1.400
1	2	95.112	71.689	1.400
3	2	103.230	49.528	1.400
3	4	92.751	65.666	0.005
4	3	107.239	65.666	0.005
4	5	96.345	78.300	0.005
5	4	103.645	78.300	0.005
5	E	99.300	45.650	1.400

Residuals are computed for the height differences with magnitudes up to `0.0017` . The resulting height differences of the new points 1,2,3,4,5, obtain a posteriori standard deviations up to `0.0012`.

Load example and "Compute"

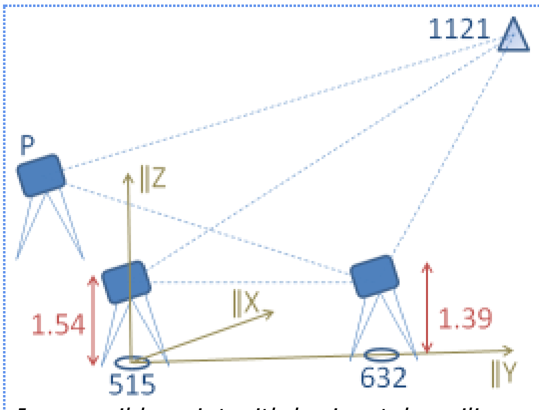
The same example is also used for the Universal computer and may be computed. However, the results are not fully identical because Universal computer computes a

robust adjustment.
The deviations in the final heights amount up to `0.0004`.

point	Vertical networks		Universal computer	
name	adj. height	stddev.	height (median)	range
1	122.3238	9.1e-4	122.3238	0.0046
2	126.4233	0.0012	126.4230	0.0046
3	130.3350	0.0012	130.3347	0.0046
4	137.7910	0.0011	137.7906	0.0033
5	142.2778	7.7e-4	142.2779	0.0046

☰ ☆ Inaccessible point with auxiliary triangles

This example has been computed with the ☆ Universal computer before. We now adjust the measurements as a trigonometric network. For this purpose we use the slope distances obtained in the universal computer. The known point height of `515` and `632` are set as fixed. The planar coordinates are used only for plotting the network on the canvas.



Inaccessible point with horizontal auxiliary triangles

During the computation a warning is issued that observations are found between fixed points, which are ignored. These are the two observations between the points `515` and `632` All observations are well reliable (redundancy parts ≥ 0.45). The adjusted heights of the inaccessible point `1121` is obtained as `201.1118` with a standard deviation of `0.0014`.

Load example and "Compute"

The same example is also used for ☆ Universal computer and may be computed. However, the results are not fully identical because Universal computer computes a robust adjustment and determines all three spatial coordinates.
The deviations in the final heights amount up to `0.07`.

☰ Loading adjustment models in "Adjustment with observation equations"

Vertical networks and Sets of angles and distances can be re-adjusted with Adjustment with observation equations . This yields the following advantages:

- Weights can be changes. E.g. target points of low sighting accuracy can be downweighted.
- Outliers can be automatically detected by w- or τ-test.
- The accuracy may be tested vs. a theoretical value. For example, it may be tested statistically, if the accuracy specification of the manufacturer of the instrument is met.
- The redundancy parts, full cofactor matrices and other interesting values are displayed.
- For many values more digits are displayed, if desired.

Forthcoming: More tools will provide this option.

The names of the observations are `A°E` and the names of the parameters are the names of the adjusted points. If you use ⇑ height differences as constraints then they do not show up as observations, but as constraints for parameters.

Exercise: Determine the standard deviation of the adjusted height difference of the points 125 and 2575 in the ☆ Campus subnetwork of HTW Dresden as a free spirit levelling network. Solution: 0.22 mm.

☝ Accuracy of adjusted, unmeasured height differences

In a Vertical network the accuracy of adjusted, unmeasured height differences should be computed. You could load the vertical network to Adjustment with observation equations like in the preceeding ☝trick and specify the height differences as functions of adjusted parameters (heights). It is simpler to specify the height differences directly in the height network as measurement with weight zero or standard deviation ''INF''. Then the values do not take part in the adjustment, but you obtain the adjusted values and also their accuracy.

Did you know? *On the start page you search IN DUBIO PRO GEO by the Google search engine.*

IN DUBIO PRO GEO Guide : Adjusting surfaces

Page contents

Through given data points in 3D-space an adjusting (i.e. best fitting) surface (plane, sphere, ellipsoid or general ›*quadric*) is computed. Also a plane through 3 points, a sphere through 4 points etc. can be computed. Additional points may be projected onto the surfaces.

Data points

First of all, data points must be given, through which the surface should pass, perhaps only approximately. All those points must have three coordinates. These coordinates are specified via Coordinate lists . In the right table the minimum number of data points is given for each type of surface. Some special configuations of data points may make the linear system of equations singular and are illegal.

Minimum number of data points	
plane	3
sphere	4
ellipsoid/ ellipt. hyperboloid	6
gen. ›*quadric*	9

Surfaces, surface parameters, surface equations

All adjusting surfaces are computed, which are computable from the given data points. As a first result for each surface you obtain the surface equation, from where you can derive the parameters of the surface.

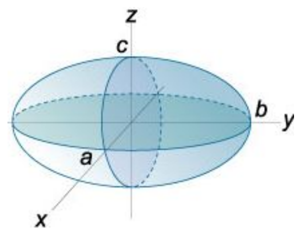
Plane: $(v_X, v_Y, v_Z)^T$ is a unit vector perpenticular to the plane (the so-called normal vector), which points away from the origin of the coordinate system. w is the distance of the origin from the plane.

$$v_X \cdot X + v_Y \cdot Y + v_Z \cdot Z = w$$

Sphere: X_o, Y_o, Z_o are the coordinates of the centre, R is the radius of the sphere.

$$(X_o - X)^2 + (Y_o - Y)^2 + (Z_o - Z)^2 = R^2$$

Triaxial ellipsoid / elliptic hyperboloid: X_o, Y_o, Z_o are the coordinates of the centre of the figure. $a,b,c > 0$ are the parameters of form of the surface. For the ellipsoid, they are the semiaxes. Also for the elliptic hyperboloid they are sometimes called semiaxes. The axes and planes of symmetry pass through the centre and are situated parallel to the axes and coordinate planes.

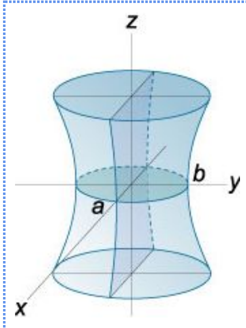


Triaxial ellipsoid

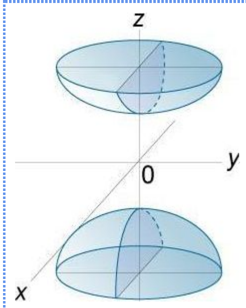
$$\pm\frac{(X_o - X)^2}{a^2} \pm \frac{(Y_o - Y)^2}{b^2} \pm \frac{(Z_o - Z)^2}{c^2} = 1$$

The type of surface depends on the signs of the fractions in this equation:

- 3× plus ⇒ ellipsoid,
- 2× plus, 1× minus ⇒ elliptic hyperboloid of one sheet,
- 1× plus, 2× minus ⇒ elliptic hyperboloid of two sheets,
- 3× minus is not possible.



Elliptic hyperboloid of one sheet



Elliptic hyperboloid of two sheets

General quadric: This type of surface is nothing but a triaxial ellipsoid or elliptic hyperboloid in oblique (non-axiparallel) position:

$$\begin{vmatrix} X_0 - X \\ Y_0 - Y \\ Z_0 - Z \end{vmatrix} \cdot \begin{vmatrix} u_{XX} & u_{XY} & u_{XZ} \\ u_{XY} & u_{YY} & u_{YZ} \\ u_{XZ} & u_{YZ} & u_{ZZ} \end{vmatrix} \cdot \begin{vmatrix} X_0 - X \\ Y_0 - Y \\ Z_0 - Z \end{vmatrix} = 1$$

X_0, Y_0, Z_0 are the coordinates of the centre of the figure. a^2, b^2, c^2 are the magnitudes of the eigenvalues of the matrix U. The type of surface depends on the signs of the eigenvalues of this matrix. The rule is the same as above. For the ellipsoid a,b,c are the semiaxes.

☹ At the moment it is not permitted that a curved adjusting surface passes through the ›*barycentre* of the data points. If data points are arranged that way, this would lead to a fatal error "singular matrix". Please avoid this case for the time being.

☰ Least squares adjustment

If the number of points to be adjusted exceeds the number of points required for the unique determination of the surface (›*redundancy*, ⇑table, then an adjustment is computed by the method of weighted least squares. In this case accuracy measures are required, either a standard deviation σ or a weight p . In the first case the weight is computed by $p=1/\sigma^2$.

Each accuracy measure applies to all points. An individual weighting of points is not directly supported. ☝Points with individual weights . Accuracy measures must not be negative, and weights must not be equal to zero. If an accuracy measure is missing or the standard deviation is equal to zero then the related coordinates are assumed error-free and held fixed during the adjustment.

Forthcoming: *Individual weights can be assigned.*

☰ Skipping surface computations

Normally, all surfaces are computed sequentially, starting from the plane and stopping at the general ›*quadric*. Ofentimes, every surface solution provides the initial guess for the next one.

Single surface computations can also be skipped. In this case the solution is not available as an initial guess in the next step. This could complicate or hinder further computations. However, this can be useful for three reasons:

- Using the normal initial guess no convergence is reached. A different initial guess is tried then.
- The computation time is occasionally shortened. The computation may be successful even for an extremely large number of data points.
- Single surface solutions are not computable due to singularities or near-singularities.

☰ Points to be projected

Optionally points can be given, which need to be projected onto each computed surface. The coordinates of these points are specified via ❓Coordinate lists . Type of system and column format must coincide with the settings for the data points. Two different modes are supported:

2D point (two coordinates given):
It is tried to find a third coordinate, such that the point lies on the surface (projection along third axis). If no such point exits, "not a number" is obtained. If two exist, the point closer to the data points is obtained.

3D point (three coordinates given):
The closest point on the surface is computed (orthogonal projection onto the surface). For each point the projection vector to the image on the surface is given, which allows to compute the distance of the point to the surface.

⚠ If the points to be projected are located far away from the data points, the accuracy of the projection is often poor due to the bad *›error propagation* of the extrapolation.

It is possible that data points and points to be projected have equal names. This case will inevitably occur, if the column format "coordinates" is chosen. In order to avoid confusion, this option is not recommended. Please note: In case of the 3D point the point to be projected and the projected point on the surface have the same name. If this is undesired, a suffix may be appended to the names of the projected points to distinguish them from the names of the points to be projected.

On the canvas all data points are displayed in horizontal projection, but neither points to be projected nor projected points.

Use of a *›grid system*

is not yet possible at the moment. Please use cartesian instead.

☆ Terrain model approximation and interpolation at Großer Garten Dresden

At Dresden Großer Garten the following terrain points are measured:

	East. [m]	North. [m]	Hght [m]
1	413736	5653962	122
2	414006	5654264	123
3	414145	5654626	117
4	413496	5654915	121
5	413035	5655364	114
6	412418	5654935	116
7	413134	5654353	121
8	413163	5654817	115
9	413678	5654471	121

Terrain model approximation and interpolation at Großer Garten Dresden

Although these coordinates refer to the UTM zone 33U, we work with a cartesian system, as long as *›grid systems* are not yet supported in Adjusting surfaces . Due to the low accuracy of the horizontal coordinates, the grid scale factor can be neglected already here.

The mean surface slope is desired by computing an adjusting plane through these 9 points. On this plane lattice points with a lattice spacing 100 m × 100 m should be computed. (Tip: Create lattice points)

Load example and "Compute"

The resulting plane is described by the equation

$$0.0057072740260338 \cdot X - 0.00092455626269208 \cdot Y + 0.99998328596978 \cdot Z = 32009.200056141$$

from which you extract the slope angle:

$$arccos(0.999983286) = 0.331° = 19.9'$$

Moreover, you also extract the azimuth of the dip (steepest descent):

$$arctan(-0.000925/0.005707) = 350.8° = 389.8\ gon$$

The dip is approximately in a northward direction.

Plane through three points, sphere through four points

Adjusting surfaces also computes a plane through three points or a sphere through four points. Weights do not matter, they may also be omitted. If you need the distances of further points from the surface or their projections onto the surface, specify them as points to be projected . The distances are computed by the lengths of the difference vectors.

Points with individual weights

In Adjusting surfaces weights are required for the data points or ›*control points*, except in the case of no ›*redundancy*. Individual weights for each point cannot yet be given at the moment, but only for each coordinate axis. However, there is a workaround: Specify the points with higher weight multiple times with different names in a coordinate list, or in the column format "coordinates" without names and identical coordinates. For example, a point specified twice acts like the same point specified once with double weight.

Did you know? *If javascript is off, you are still able to use almost all features. IN DUBIO PRO GEO is doing without cookies.*

IN DUBIO PRO GEO Guide : Library

Page contents

Here you have access to a collection of links to scientific documents in the World Wide Web regarding important subjects of Geodesy. At the moment there are `2824` documents with a total number of `90000` printing pages and `5` GByte available. Latest update with full link checking: `2018/09/26`.

Introduction

The documents are protected by copyright laws.

Exceptionally there is full bibliographic information available. Partly this information can be found in the documents themselves. Otherwise it is advisable to look for this information on the server, where the link is directed to, perhaps by the keyword publications.

Open or restricted access

Most documents are open and free available. Some documents require a registration at the server, where the original document is hosted. Sometimes such a registration is free of charge.

Documents with restricted access are usually included in this library only if they cover a subject important for IN DUBIO PRO GEO, i.e. geodetic computations and adjustment.

Students of geomatics at HTW HOCHSCHULE FÜR TECHNIK UND WIRTSCHAFT DRESDEN UNIVERSITY OF APPLIED SCIENCES connected to HTW network, may have access to a local copy of some documents. A different opportunity to obtain documents with restricted access is ⇛SCI-HUB, however, at your own responsibility.

open access
restricted access
only for HTW-users
no online access

Symbols and their meaning

Categories and document types

Each document may belong to multiple categories. They will be combined by AND-operation, i.e. all matches must be assigned to both categories.

Each document is attributed by one document type. You can search for multiple or all types at a time.

Search text

Sorry, we search only in bibliographic information (authors, title, reference). I.e., we do not search in the documents themselves. Therefore, note that there is no full text search! The search is case-insensitive.

You may search for one or multiple **substrings** . So you find with the search text `oto` both `Foto` and `Fotograph` , but also `Photo` and `Photograph` . Multiple substrings can be separated by space or comma. So you find with the search text `Höhe,height` both documents in German as well as in English regarding the subject `height`.

Did you know? *Place mousepointer over terms marked with › to get their meaning.*

IN DUBIO PRO GEO Guide : Tutorial

Page contents

Tutorials explain the functions of IN DUBIO PRO GEO by means of practical examples. The solutions of the problems are comprehensible on the basis of pre-filled forms. In the select areas not all options are available. The forms may be submitted by the button Compute and the results may be viewed. For the sake of convenience the results are given in extracts in the tutorial. The most important intermediate and final results are highlighted by golden boxes. The solutions are commented in further golden boxes.

Sorry, at the moment tutorials are only in German.

The tutorial problems either involve different computation tools or are more complex problems. Smaller problems, which use only one computation tool, you find in the guides of the various computation tools.

⚠ The solutions to the problems sketched in the tutorials and guides refer to the case that you work with the standard Settings . Otherwise, small variations may occur.

List of Tutorials

- Equilateral triangular lattice
- Area partitionment
- Circular arc stakeout
- Hansen´s problem
- Spatial line section
- Rectangle through five points
- Square through four points
- Cylinder through seven points
- Incompletely connected traverse
- Adjustment of a triangle

List of guiding examples

- ☆ Coordinate lists: GPS reference point of HTW Dresden
- ☆ Create lattice points: 2D lattice for Großer Garten Dresden
- ☆ Create lattice points: Loxodrome from Dresden (Saxony) to Dresden (Ontario)
- ☆ Planar polygons: Surrounding polygon for Großer Garten Dresden
- ☆ Matrix computations: Orthogonal matrix
- ☆ Arithmetic expressions in matrices
- ☆ Satellite orbits: Orbit computation from a GPS almanach
- ☆ Grid scale factors: Points on the 51° parallel
- ☆ Normal gravity formulae: gravity benchmark at the geodetic laboratory of HTW Dresden
- ☆ Transformation by parameters: Rotate cuboid about centre axis
- ☆ Transformation by control points: Cuboid through four vertices
- ☆ Sets of angles and distances: Pure processing of horizontal angles
- ☆ Sets of angles and distances: Processing of zenith angles with slope distances
- ☆ Sets of angles and distances: Joint processing of all measurements
- ☆ Station centring: Eccentric angular measurements to remote targets
- ☆ Atmospheric EDM correction: Leica TS30, correction of erroneous settings
- ☆ Traverses: Branched traverse with spatial intersection
- ☆ Universal computer: Polar values computed from cartesian coordinates
- ☆ Universal computer: Inaccessible point with auxiliary triangles
- ☆ Universal computer: Planar trilateration network
- ☆ Universal computer: Trigonometric levelling line
- ☆ Universal computer: Computation of the orthocentre of a triangle
- ☆ Repeated measurements: Repeated height determination of a point
- ☆ Repeated measurements: Normally distributed random numbers

- ☆ Repeated measurements: Not normally distributed random numbers
- ☆ Double measurements: Double determination of point heights
- ☆ Adjustment with observation equations: Adjusting square through four vertices
- ☆ Vertical networks: Campus network of HTW Dresden as a free spirit levelling network
- ☆ Vertical networks: Trigonometric levelling line
- ☆ Vertical networks: Inaccessible point with auxiliary triangles
- ☆ Adjusting surfaces: Terrain model approximation and interpolation
- ☆ Error propagation: Radius of circular arc
- ☆ Error propagation: Arcs intersection

Did you know? *The solutions to the problems sketched in the tutorials and guides refer to the case that you work with the standard* ⚙*Settings . Otherwise, small variations may occur. While executing some examples, settings will be reset to standard values.*

☰ IN DUBIO PRO GEO Guide : Bag of tricks

Page contents

IN DUBIO PRO GEO provides more than you probably expect. This bag of tricks contains a selection of valuable tricks, which make the work easier. They are sorted by level of user (beginner ⇒ expert).

☰ ☝ Arithmetic expressions in input fields and tabular data records

Instead of numerical values, like e.g.

```
16.1063     16,1063     161063e-4     1610.63%
```

you may always also give arithmetic expressions, like e.g.

```
8.1+8.0063       (3,3009-1)*7,0       8.1+80063e-4    pi*16.1063/pi
161063/10000     log(exp(16.1063))    2,3009*7,0      sqrt(16,1063*16.1063)
3,3009*7,0-7     asin(sin(0.161063))*100              (16.1063^(-0.5))^(-2)
```

All 10 arithmetic expressions yield the same numerical value. This also works in tabular data records like measurement lists or coordinate lists as well as matrices.
In these examples the chosen output decimal separator is not in effect.

The following mathematical functions are supported: `abs acos acosh asin asinh atan2 atan atanh cos cosh exp log10 log sin sinh sqrt tan tanh`
⚠ Arguments of trigonometric functions are always expected in the angle unit radian, no matter which unit has been used elsewhere.

☹ Arithmetic expressions in input areas are not workable in the ⇓angle unit `DegMinSec`.

See Input areas and ☆ Arithmetic expressions in matrices.

☰ ☝ More decimal digits and overlong pointnames in coordinate lists

If in the coordinate lists of the computed points not enough decimal digits are displayed for you, it is recommended to load this list in a new browser tab or to a new coordinate list. Then you see more decimal digits.

⚠ Beim Speichern einer Liste werden eventuell zuvor schon gespeicherte Listen überschrieben. Aber zum Ansehen der Liste in der Eingabemaske von XY Edit coordinate list 1 or XY Edit coordinate list 2 When saving a list, possibly a previously saved list is overwritten. But for display of a list, this is not necessary.

If you use overlong pointnames, they are truncated in tables mostly to 12, sometimes to 9 leading characters. A warning is issued. To see the pointnames in full length, the same trick can be used.

See Filter, load and save coordinate lists.

Circle through three points

Planar polygons and Spatial polygons also compute a circle through three points, this is the case if the closed planar or spatial polygon consists of exactly three points. Below "Special points" you find the centre of the circumscribed circle M_3 . The radius R you find if the polygon is loaded in Planar triangles and computed. By the way, the same applies to the centre of the inscribed circle M_4 and the corresponding radius r . See Planar polygons.

Plane through three points, sphere through four points

Adjusting surfaces also computes a plane through three points or a sphere through four points. Weights do not matter, they may also be omitted. If you need the distances of further points from the surface or their projections onto the surface, specify them as points to be projected . The distances are computed by the lengths of the difference vectors.

Measurement lists with distances etc. in grid scale

In Measurement lists the unit for metric quantities e, s, dh, l, ih, th must always be the natural length unit .

If for the grid system in a measurement list the grid scale is always applied nonetheless, please temporarily change the system type to cartesian (XYZ or YXZ) lefthanded and perform the computations. Now everything is computed in the grid scale. Afterwards reset the system type back to grid system (northing easting height or easting northing height).

By the way, the same trick works for translation parameters as well as for edge lengths and lattice spacings for Create lattice points.

Blind target points in the universal computer

Oftentimes the Universal computer computes polar values only between points related by measurements (station and target in a station setup). More results are sometimes obtained by adding blind target points without measurements. E.g. if you want to obtain the horizontal distance between two known or computed points then they may be additionally given as station and target without measurements. This value may be used in further computations, if useful. Find such a case in the ☆ Polar values computed from cartesian coordinates.

Error propagation with the Universal computer and with Trilateration

Although the Universal computer does not support the error propagation directly, it is possible to skillfully compute an error propagation also here. Let us assume, we have n inaccurate start quantities (coordinates and/or measurements). You run the computation $n+1$ times, one time with unchanged start values and n times with one start value changed by its standard deviation or maximum absolute deviation, one after the other. The differences Δ_i of the desired results with respect to the first run of the computation are totalize according to the law of error propagation:

for standard deviations $\sigma^2 = \Delta_1^2 + \ldots + \Delta_n^2$

for maximum absolute deviations: $\Delta = |\Delta_1| + \ldots + |\Delta_n|$

If you rename the points wíth changed coordinates and append them to the coordinate list and if you also assign different names to the $n+1$ results of the unknown points, then you can run the $n+1$ computations in the Universal computer **in a single flow** .

The same works just as well with the computation tool Trilateration.

See ☆ Arcs intersection.

Points with individual weights

In Adjusting surfaces weights are required for the data points or control points, except in the case of no redundancy. Individual weights for each point cannot yet be given at the moment, but only for each coordinate axis. However, there is a workaround: Specify the points with higher weight multiple times with different names in a coordinate list, or in the column format "coordinates" without names and identical coordinates. For example, a point specified twice acts like the same point specified once with double weight.

Loading adjustment models in "Adjustment with observation equations"

Vertical networks and Sets of angles and distances can be re-adjusted with Adjustment with observation equations . This yields the following advantages:

- Weights can be changes. E.g. target points of low sighting accuracy can be downweighted.
- Outliers can be automatically detected by w- or τ-test.
- The accuracy may be tested vs. a theoretical value. For example, it may be tested statistically, if the accuracy specification of the manufacturer of the instrument is met.
- The redundancy parts, full cofactor matrices and other interesting values are displayed.
- For many values more digits are displayed, if desired.

Forthcoming: More tools will provide this option.

Accuracy of adjusted, unmeasured height differences

In a Vertical network the accuracy of adjusted, unmeasured height differences should be computed. You could load the vertical network to Adjustment with observation equations like in the preceeding trick and specify the height differences as functions of adjusted parameters (heights). It is simpler to specify the height differences directly in the height network as measurement with weight zero or standard deviation "INF". Then the values do not take part in the adjustment, but you obtain the adjusted values and also their accuracy.

Satellite orbit in the starfixed system

Satellite orbits usually computes discrete orbit points in the earth fixed (rotating) righthanded cartesian coordinate system ECEF. However, if you want to obtain the orbit points in the star fixed (quasi inertial) system ECSF, simply specify for the angular velocity of Earth's rotation $\omega_E = 0$. Then you obtain a system with axes coincident with the earth fixed system at the beginning of the week and then kept star fixed.

Satellite orbit velocitiy

Satellite orbits are computed in the form of Coordinate lists of discrete orbit points on a time grid. If you want to obtain the orbit velocity, transfer the coordinate list to Spatial polygons and compute it as an open polygon. Then you obtain the spatial distances of consecutive orbit points as side lengths of the polygon. If you chose e.g. 1 second as a computation time increment Δt then the side lengths are immediately velocities in metres/second.

Usually you obtain the velocities in the earth-fixed (rotating) coordinate system. If you want to obtain them in the starfixed system, please apply the previous trick. If you do this for the ☆ Orbit computation from a GPS almanach (time increment 1h), then you obtain orbit velocities between `13600 km/h` and `14000 km/h`.

☰ Miscellaneous

- To show the current settings for the coordinate system, after you have transferred coordinates into a coordinate input area (i.e. not manually typed) without invoking Settings you simply click Clear all and afterwards you click Reset.
- You may hide undesired images and/or additional information, e.g. the globe logo or canvases. Please change the Settings.
- Missing measurements in Measurement lists inside a measurement record can be skipped by chosing semicolons as delimiters and by leaving the field blank: " ; ; " At the end of the row measurements can simply be omitted.
- The tools for ellipsoids of rotation also operate on the sphere. Specify the sphere for reference ellipsoid.

Did you know? *There are much more tricks, which will here be added soon.*

IN DUBIO PRO GEO Guide : Information criteria

Page contents

Information criteria are computed for your models. This enables you to select the optimal model. Some computation tools compute various models unasked and list the corresponding information criteria.

Introduction

An information criterion is a criterion for the selection of a model in statistics. If we have stochastic observations and a set of model candidates at our disposal, then we compute for all candidates the corresponding value of the information criterion. A small value indicates that a model is adequate. The model with the least value is most adequate for the observations and should be selected.

Definitions

Unfortunately, in statistics there are various definitions of information criteria. The most important are:

information criterion	formula	symbols
Akaike	$AIC = 2 \cdot k - 2 \cdot log(L(\bar{\theta}; l))$	k = number of model parameters θ
Akaike corrected	$AICc = AIC + 2k(k+1)/(n-k-1)$	n = number of observations l
Bayes	$BIC = log(n)k - 2 \cdot log(L(\bar{\theta}; l))$	L = likelihood function of the model
		$\bar{\theta}$ = maximum likelihood estimation of θ

All these criteria are split into a penalty term for the number of model parameters, which penalizes an overfitting, and a model fitting term $-2 \cdot log(L(\bar{\theta}; l))$. Note that to the k model parameters θ also belong the (co-)variance parameters and components. However, if d datum defects are present, e.g. for free network adjustment, then k is to be decreased by d. If the model additionally contains m independent constraints for parameters, then k is to be decreased by m.

In geodesy models with normal distributed observations are widespread. Two cases are important:

Normal distributed observations - prio case

In this case the covariance matrix of the observation Σ_l is fully known.
We have $\theta = x$ and $k = u - m - d$. The likelihood function L reads:

$$L(\bar{x}; l) = (2\pi \cdot det(\Sigma_l))^{-1/2} exp(-(A\bar{x} - l)^T \Sigma_l^{-1} (A\bar{x} - l)/2) = (2\pi\sigma^2)^{-n/2} det(P)^{1/2} exp(-\sigma^{-2} (A\bar{x} - l)^T P (A\bar{x} - l)/2)$$

Here the symbols of Adjustment with observation equations are used. From there it is deduced that

$$-2 \cdot log(L(\bar{x}; l)) = n \cdot log(2\pi\sigma^2) - log(det(P)) + \sigma^{-2} (A\bar{x} - l)^T P (A\bar{x} - l)$$

If the case may be that the weight matrix P is a diagonal matrix with the elements $p_1, p_2, \ldots, p_n$, we obtain

$$-2\cdot log(L(\bar{x}; l)) = n\cdot log(2\pi\sigma^2) - \Sigma\, log(p_i) + \sigma^{-2}(A\bar{x} - l)^T P(A\bar{x} - l)$$

The first two terms are identical for all information criteria and models and may be omitted when taking the minimum. For comparability of the values, IN DUBIO PRO GEO computes them nonetheless.

$$C_{prio} := n\cdot log(2\pi\sigma^2) - \Sigma\, log(p_i)$$

$$\Omega(\bar{x},l) := (A\bar{x} - l)^T P(A\bar{x} - l)$$

$$AIC_{prio} = 2k + \sigma^{-2}\Omega(\bar{x},l) + C_{prio}$$

$$AICc_{prio} = 2k + 2k(k+1)/(n-k-1) + \sigma^{-2}\Omega(\bar{x},l) + C_{prio}$$

$$BIC_{prio} = log(n)k + \sigma^{-2}\Omega(\bar{x},l) + C_{prio}$$

Normal distributed observations - post case

In this case the covariance matrix of the observation $\Sigma_l = \sigma^2 P^{-1}$ contains an unknown covariance factor σ^2. It must be estimated as well. In this way θ encloses a further quantity, i.e. $k = u - m - d + 1$. As an estimator for σ^2 the (biased) maximum likelihood estimator is used:

$$\bar{\sigma}^2 = \Omega(\bar{x},l)/n$$

The new likelihood function reads:

$$L(\bar{x},\bar{\sigma}^2; l) = (2\pi\bar{\sigma}^2)^{-n/2} det(P)^{1/2} exp(-\bar{\sigma}^{-2}(A\bar{x} - l)^T P(A\bar{x} - l)/2) = (2\pi\Omega(\bar{x},l)/n)^{-n/2} det(P)^{1/2} exp(-n/2)$$

$$-2\cdot log(L(\bar{x},\bar{\sigma}^2; l)) = n\cdot log(2\pi\Omega(\bar{x},l)/n) - \Sigma\, log(p_i) + n$$

Summarizing, we obtain in the post case:

$$C_{post} := n\cdot log(2\pi) - \Sigma\, log(p_i) + n$$

$$AIC_{post} = 2k + n\cdot log(\Omega(\bar{x},l)/n) + C_{post}$$

$$AICc_{post} = 2k + 2k(k+1)/(n-k-1) + n\cdot log(\Omega(\bar{x},l)/n) + C_{post}$$

$$BIC_{post} = log(n)k + n\cdot log(\Omega(\bar{x},l)/n) + C_{post}$$

***Did you know?** With Repeated measurements you can test measurement value for normality.*

IN DUBIO PRO GEO Guide : Earth curvature correction

Page contents

Introduction

Computations in the vertical triangle with long side lengths should take into account the curvature of the earth. The plumb lines of the start point and the end point of lines and edges of networks or sides of polygons can no longer be regarded as parallel. In the computation tool Vertical triangles this is always taken into account. In the computation tools Traverses, Universal computer, Vertical networks this is only an option. Be sure not to check this option if you have corrected your measurements already for the curvature of the earth.

In the computation tool Trilateration no curvature of the earth is corrected.

Radius of the earth

The radius of the earth to be used for the correction must be given in the same unit as your coordinates and distances.

The flattening of the earth is neglected. It is best to use the local mean radius of curvature of the ellipsoid at point site. It is computed by the computation tool Latitude dependent quantities. A value nearly suitable for the entire earth is `6371000 m`.

Effect of earth curvature for horizontal sighting

Using the universal computer it shall be investigated, how large the effect of the curvature of the earth may be for a horizontal sighting at different distances. For this purpose we measure from point `Station` to different targets at various distances.

The obtained height differences `dh` between `Station` and the target points as well as the shortenings of the horizontal distances `e` with respect to the slope distances `s` show the effect of the curvature of the earth.

We obtain the following results:

target	height diff.	horiz. dist.
0001	7.848062e-8	1.000000000
0003	7.063255e-7	3.000000000
0010	7.848255e-6	10.00000000
0030	7.063244e-5	30.00000000
0100	7.848060e-4	99.99999999
0300	0.007063256	299.9999998

Pointnames and measurements

```
Station
// target   zenith angle  slope distance
   0001       100           1
   0003       100           3
   0010       100           10
   0030       100           30
   0100       100           100
   0300       100           300
   1000       100           1000
   3000       100           3000
```

☑ Apply Earth curvature correction with Earth radius 6371000 x

Load example and “Compute” in Universal computer

1000	0.078480614	999.9999918
3000	0.706325498	2999.999778

For comparison we compute the last row with Vertical triangles . Here the corresponding results are given below **dh** and **e** .

Load example and “Compute” in Vertical triangles

Did you know? *Angles are also accepted like* **16°10'19.63"** *or* **970.32717"** *.*

IN DUBIO PRO GEO Guide : Error propagation

Page contents

Introduction

In geodesy we not only compute unknown quantities, but we also estimate, how accurate or inaccurate such quantities are obtained. Computed quantities inherit their deviations, the so-called "errors", from the quantities, from which they are computed. These quantities may have been measured and are affected by measurement errors or computed from other quantities and so have again inherited their deviations. It is said that errors propagate. Both amplification as well as dilution of errors is possible.

It is important to carry out error propagations

1. to specify the accuracy of computed quantities
2. to determine the required accuracy of measured values and
3. to optimize measurement arrangements.

These three goals are attained by the application of propagation laws.

Accuracy measures

Accuracies may be expressed by numbers in different ways. IN DUBIO PRO GEO works with the following three accuracy measures:

standard deviation	**Std**	formerly known as "mean error"
maximum absolute deviation	**Max**	worst case error
weight	**Wgt**	a relative accuracy measure

In geodesy the standard deviation is more popular because the maximum absolute deviation is difficult to estimate or yields values, which are theoretically possible, but only by a chain of worst cases.

- Accuracy measures must not be negative, and weights must not be zero.
- In case of missing accuracy measures the related quantities are treated as error-free.
- Deviation measures must not be arbitrary large, otherwise you may get incorrect results and in extreme cases even no result at all.

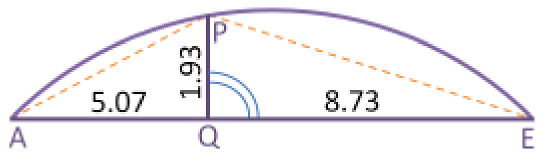
A simple test for deviations to be sufficiently small is to repeat the computation with all accuracy measures halved and to check that also all computed measures are approximately halved.

Application of propagation laws

For each accuracy measure there is a corresponding propagation law. Please find the theoretical basics explained here: Fehler- und Kovarianzfortpflanzung **(Sorry, only in German)**

If for any uncertain start quantity a related measure of uncertainty is available then for some computation tools ( ) these measures may be specified and the related accuracy measures for all computed quantities are obtained. This requires the start quantities to be statistically uncorrelated. Practically this means that these deviations ultimately result from measurement errors, but each of them has affected only one start quantity.

Radius of circular arc



Radius of circular arc by chord segment and chord offset

How accurate can we determine the radius of a circular arc from tape measurements of the chord segments AQ and QE and chord offset PQ? From experiences we obtain for all three measurement values a **standard deviation** of `0.03` The result is a radius of `12.57` with a standard deviation of `0.18`.

Load example and "Compute"

If we assume instead or additionally a **maximum absolute deviation** of the tape measurements of `0.1` each, then we obtain for the radius a maximum absolute deviation of `0.86`.

Load example and "Compute"

In both cases the deviations are significantly amplified. This effect is even stronger for shallow arcs, which occur practically e.g. in traffic infrastructure construction.

If we would not know for sure, which standard deviation should be assigned to the tape mesurements, but only that they have a certain ration, then we could work with weights. If these standard deviation are all equal, we choose the weight 1 for all measurements and compute the **weight propagation**. As a result we obtain for the radius the relatively small weight `0.028` which indicates a bad error propagation.

Load example and "Compute"

If we would try to determine the radius from measurements of the long chord AE and the two short chords AP and PE (⇑ figure) with the same accuracy, i.e., with the same weight 1, the we would obtain by weight propagation for the radius the even smaller weight `0.0034` This measurement setup is therefore even less recommendable.

Load example and "Compute"

Error propagation with the Universal computer and with Trilateration

Although the Universal computer does not support the error propagation directly, it is possible to skillfully compute an error propagation also here. Let us assume, we have n inaccurate start quantities (coordinates and/or measurements). You run the computation $n+1$ times, one time with unchanged start values and n times with one start value changed by its standard deviation or maximum absolute deviation, one after the other. The differences Δ_i of the desired results with respect to the first run of the computation are totalize according to the law of error propagation:

for standard deviations $\sigma^2 = \Delta_1^2 + \ldots + \Delta_n^2$

for maximum absolute deviations: $\Delta = |\Delta_1| + \ldots + |\Delta_n|$

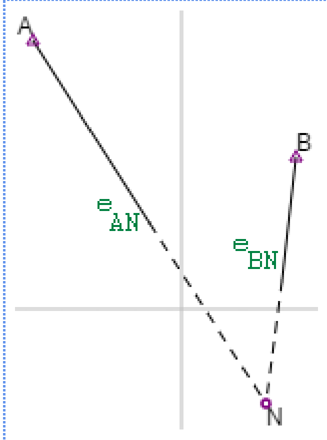
If you rename the points wíth changed coordinates and append them to the coordinate list and if you also assign different names to the $n+1$ results of the unknown points, then you can run the $n+1$ computations in the Universal computer **in a single flow** .

The same works just as well with the computation tool Trilateration.

Arcs intersection

Let us consider an arcs intersection with two known points A and B and an unknown point N as well as two measured horizontal distances $e_{AN} = 11.436$; $e_{BN} = 6.576$. The four position coordinates have standard deviations of `0.03` and both distances have standard deviations of `0.01`. Consequently, we have 6 inaccurate start quantities $x_A, y_A, x_B, y_B, e_{AN}, e_{BN}$. Thus, the list of known points may read:

```
     Y     X
A   16.10 17.11 // unchanged point
B   23.06 14.02 // unchanged point
AY  16.13 17.11 // Y-changed point A
BY  23.09 14.02 // Y-changed point B
AX  16.10 17.14 // X-changed point A
BX  23.06 14.05 // X-changed point B
```

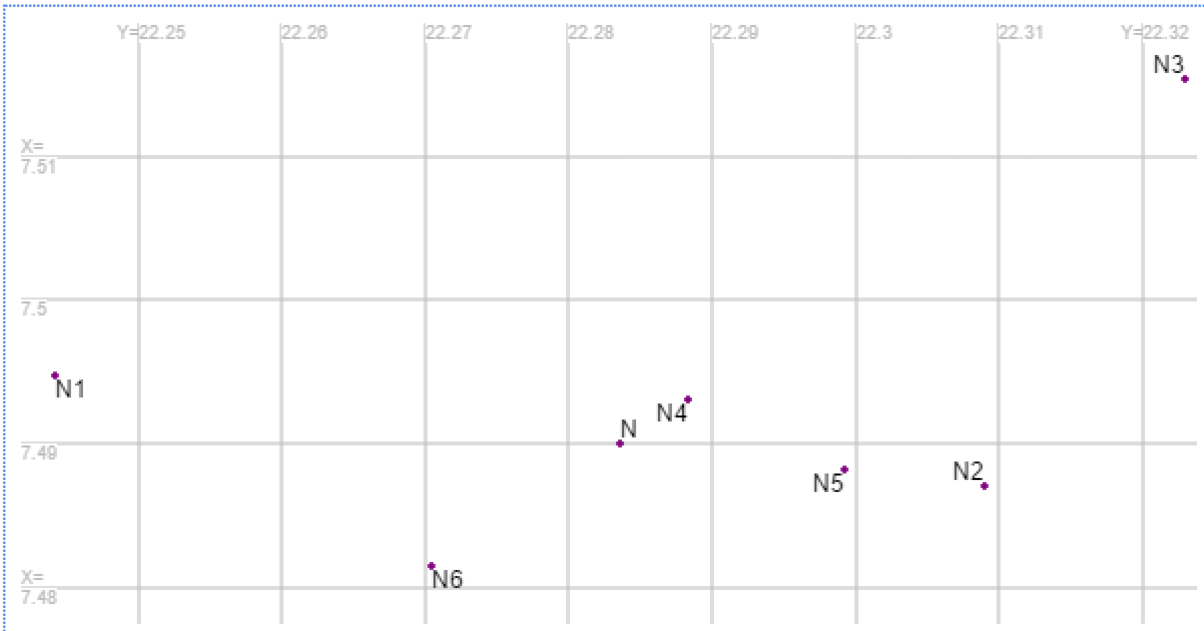


Arcs intersection

Now we compute the 7 arcs intersections **in a single workflow** by the Universal computer and to the 7 results for the unknown point we assign the pointnames `N,N1,N2,…,N6` . Every arcs intersection has two solutions. They combine to a total of $2^7=128$ solutions. But it is sufficient to consider only the first one. All other solutions yield the same accuracies.

Load example and "Compute" with Universal computer
Load example and "Compute" with Trilateration

It is easy to display the spatial distribution of the 7 computed points by storing them in a coordinate list and looking at the canvas. The largest impact is made by the X coordinates of the known points `A (→N1), B (→N3)`:



Display of the computed points on the canvas of the coordinate list

Now we must compute the differences Δ_i . This can be supported by Transf. by parameters and computation of a translation, such that N is moved to the origin of the coordinate system. The transformed coordinates of the other 6 points are immediately the desired differences. Finally, we obtain for the unknown point N by application of the law of error propagation for standard deviations:

$$\sigma_Y=(0.039^2+0.025^2+0.039^2+0.005^2+0.016^2+0.013^2)^{1/2}=0.065$$

$$\sigma_X=(0.005^2+0.003^2+0.025^2+0.003^2+0.002^2+0.008^2)^{1/2}=0.028$$

If the given deviations would have been maximum absolute deviations then by application of the law of related error propagation we would obtain:

$$\Delta_Y = 0.138;\ \Delta_X = 0.047$$

In both cases the deviations amplify mainly in the Y coordinate.

Did you know? *Place mousepointer over terms marked with › to get their meaning.*