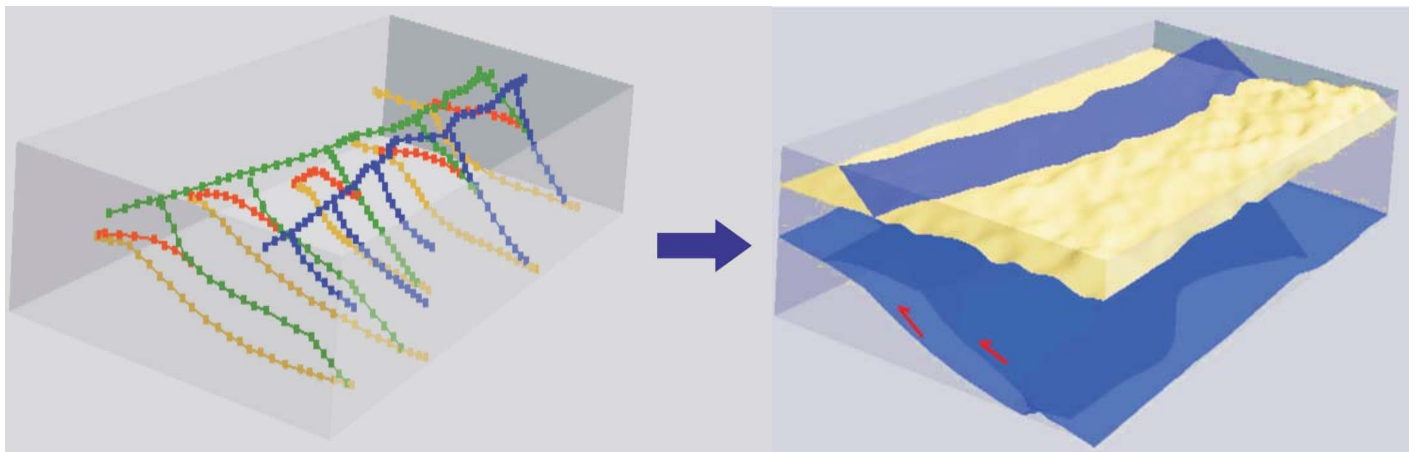




Introduction to 3D Geomodelling with gOcad

lecture materials

Chair of Geoscience Mathematics and Informatics
Department of Geophysics and Geoinformatics
TU Bergakademie Freiberg



FUNDAMENTAL CONCEPTS

1. 1. The problem

Geometry and properties of geological objects are only accurately known at certain sampling locations and have to be interpolated, estimated or simulated for the remaining object. Following Jean-Laurent Mallet, "Geomodelling consists of the set of mathematical and computer science methods allowing to model the topology, the geometry and the properties of geological objects in an unified way while taking into account any type of data related to these objects". This implies that a modelling geologist has to

1. subdivide the volume of interest into a set of geoobjects according to the project motivation, like a set of fault blocks, horizons, reservoirs, ore bodies
2. compile all data related to these geoobjects in order to constrain the topology, geometry and properties in an optimum way. Such data can be of heterogeneous type, certainty and spatial distribution.
3. use appropriate methods to create self-consistent 3D geomodels composed of geoobjects. These need to honor available data. Examples are interpolation, estimation and simulation methods.

1. 2. Discrete Model

In order to store spatial data and geoobjects in a computer, a discrete data model is required (fig. a). Discrete geoobject representation is based on regular or irregular tessellations to model the spatial extent of geoobjects using surface partitions or volume partitions.

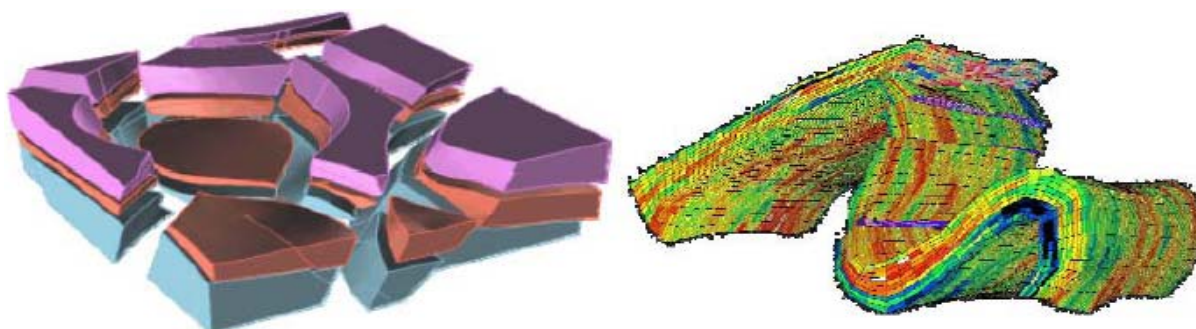


Fig. 1. a. Discretization of the Model Space into Regions Corresponding to Geoobjects (Model3d, left), and Discretization of a Geoobject into 3d Cells (SGrid, right).

Discrete topological model. The Boundary Representation (BRep) approach models the spatial extent of geoobjects by a discretization of their boundary. Using this technique, the geological space can be partitioned into regions. The radial edge Weiler representation implemented in gOcad Model3d objects maintains a hierarchy of topological elements. Each level of the hierarchy corresponds to various stages of the discretization process as one moves from a geomodel down to a mesh with its associated numerical properties. The abstraction hierarchy is comprised by the following models:

1. discrete model (geometry and property model).
2. discrete topological model.

It is composed of

1. volume region decomposition
2. face decomposition
3. border decomposition
4. triangulated mesh

Discrete model. A discrete model stores the topological information and the geometry, constraints and properties. In gOcad, this information is stored at the nodes of Model3d, Surfaces, Solids, Curves, or PointsSet objects. By definition, a discrete model $M^n(\Omega, N, \varphi, C)$ consists of a graph $G(\Omega, N_i)$ defining the topological neighborhood N_i of the set of nodes Ω , vectorial functions φ^v defining the geometry and properties of each node α , a set of linear constraints C_i , which can be chosen and weighted in order to let φ^v honor data and geological concepts.

1.3. Data model

The most primitive model object available is the PointsSet, which is composed of a set of points possessing properties such as geometry (XYZ coordinates) and arbitrary vectorial properties. Curves are nodes which are connected by linear segments (fig. b). Each node has two or one (in the case of an extremity) neighbour nodes. Curve objects can be composed of a set of isolated curve parts.

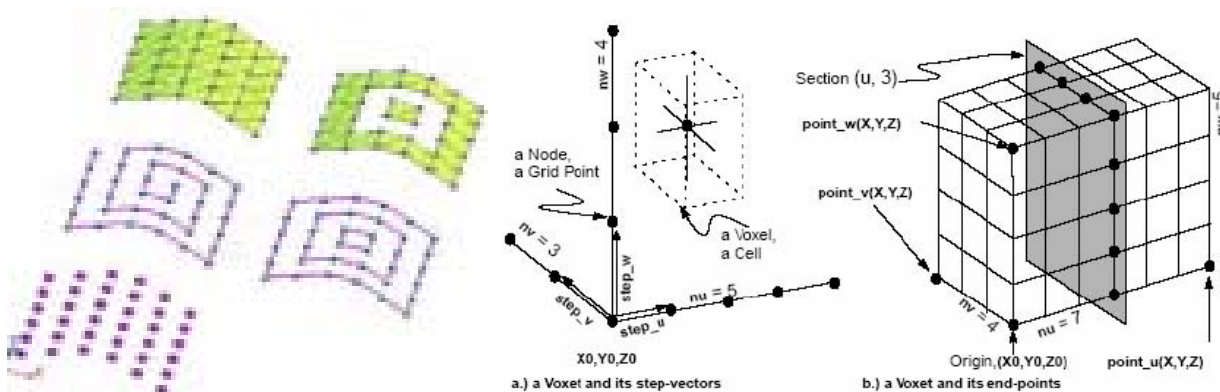


Fig. 1. b. Different algorithms to connect points to curves or surfaces (left); Parameters describing a voxel (right)

Triangulated surfaces are most important for modelling geological boundary surfaces. Planar triangles are interpolated between a set of connected nodes. A surface object is composed of a set of isolated parts (faces, fig. b). Parts possess topological features like border (boundary) and borderstones. A sealed Model3d (BRep model) or an SGrid that is composed of voxels (fig. b) can be built from a set of intersecting surface parts. The geometrical and property information of surfaces is stored at the points.

Solids are composed of connected tetrahedra. A volume region which represents a geobject can be discretized using solids, which allows to model properties in 3d volumes. Again, the properties are stored at the nodes and interpolated between them. SGrids and Voxets are regular grids. Numerical properties can be carried by the cell nodes or the cell center. SGrids can be deformed and cut in order to fill the volume region of a geobject. Such curvilinear SGrids are commonly used for property modeling in geobjects, like hydrocarbon reservoirs. Parallelepipedic Voxets are similar to SGrids and often used for property modeling in gross volumes, like seismic cubes.

GENERAL FEATURES

2. 1. Starting gOcad

Strat gOcad with start/all programs/paradigm/gocad2011.2

When you start gOcad, you first create a new project or select an existing one in which you store your work. Each project consists of a project file and a project folder, which are both needed for working with the project. After you have specified the project, you choose the modules you want to work with. Tick the „change modules“ button for an existing project and tick the modules in the Module Selector. If you create a new project, the Module Selector is opened automatically. Only those modules, for which you have licences, are available for you and are displayed in black in the Module Selector.

When you create a project, you have to define all units of measure and orientation in the „Initialize Project Units“.

Gocad works with a cartesian and metric coordinate system. If you want to reference your coordinate system to a global projected coordinate system, you can tick „Specify Global Cartographic Reference System“ and choose the system and the datum. Note that this information is only used for reference, but not for data conversion.

Take care for choosing a proper z-axis orientation. In geological projects, z-axis is normally upward. If you choose a downward axis, some gOcad tools do not work, f.e. the tool „apply range thickness constraints“. In this excersice, we will always work with „z-axis upward“.

2. 2. The gOcad User Interface

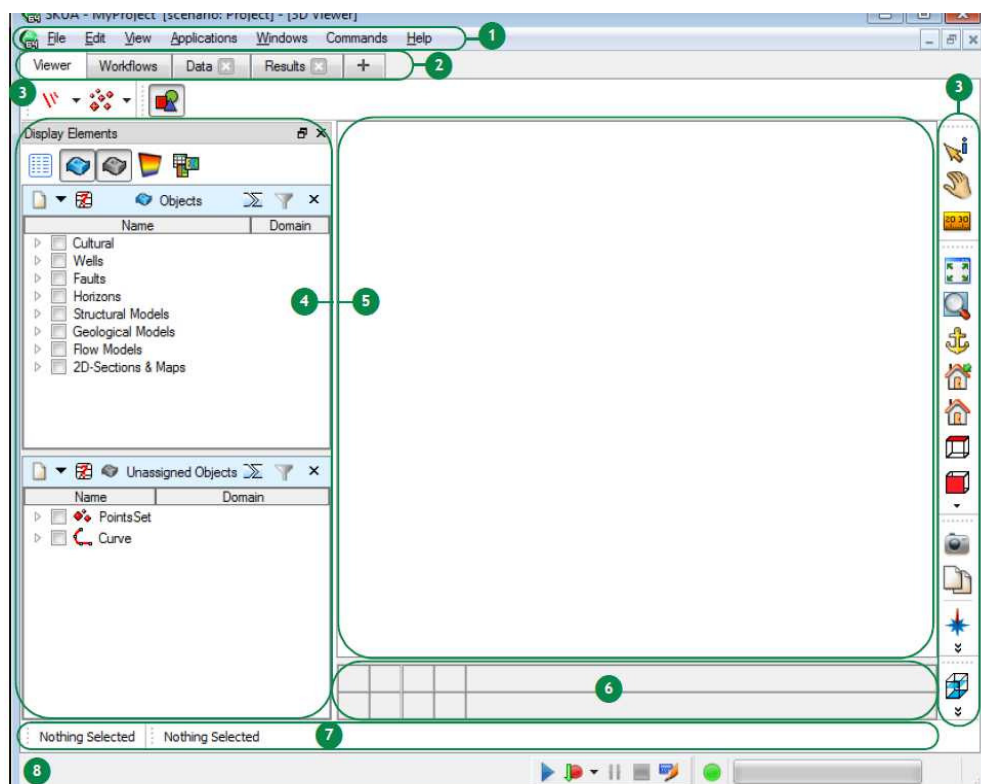


Fig. 2. a. gOcad User Interface, find the names of the toolbars in the text below.

The User interface (fig. a) is subdivided into the following areas:

1. Main Menu Bar - data load and save, general settings, general data management, data analysis applications, help access.

1a. Commands - used for object creation; topology, geometry and property editing, interpolation and computations. Upon start-up, the bar appears in General Mode, which can be changed to particular object modes (Surface Mode etc.) with respective functionality.

2. Tabs - Predefined tabs like object browsers, data tables, view windows. Arrange your data in the manner that best suits your needs.

3. Camera settings toolbar- Here you can setup the camera position, AutoSetup will direct the camera towards all selected objects (fig. b). The dialog at the very right (default value: "1") allows to scale the Z axis for vertical exaggeration.

3a. Digitization Selection Bar

4. Object panel - contains the objects and data browsers from which you can access all your modelling objects, many commands for visualisation, organizing your objects and property analysis.

5. Viewer workspace/ 3D camera - shows the objects selected in the object panel in a cartesian coordinate system; left mouse button for spin, right mouse button for zoom. If the Camera Control Panel is activated using the Panel Selection Panel (5), camera settings can be changed like the increment of spin/pan/zoom actions carried out using the arrow keys (pan), numerical keys 2,4,6,8 (orbit) or numerical keys 2,4,6,8 and „+“ (zoom) of your keyboard. Selected objects can also be filtered according to many criteria.

6. Coordinates and information - Tracks the pointer in the view and displays the tracked coordinates and properties.

7. Quick attribute toolbar - depending on the currently highlighted objects in the Object Panel (6), the bar dynamically changes (fig. c) and allows to set the graphical object attributes. General hint: Many command actions correspond to standards: drag & drop, multiple selection of objects in the camera or dialogs: control or shift key, undo: control-z, save: control-s, repeat a command: space key.

8. Status toolbar - contains information about carrying out commands.

9. Terminal (not shown) - The terminal is initially hidden and can be opened with view - Terminal. It displays the section history, all commands and error messages related to the commands.

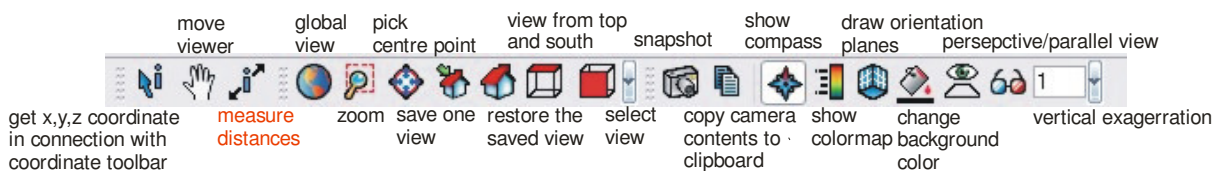


Fig. 2. b. Camera Settings Bar

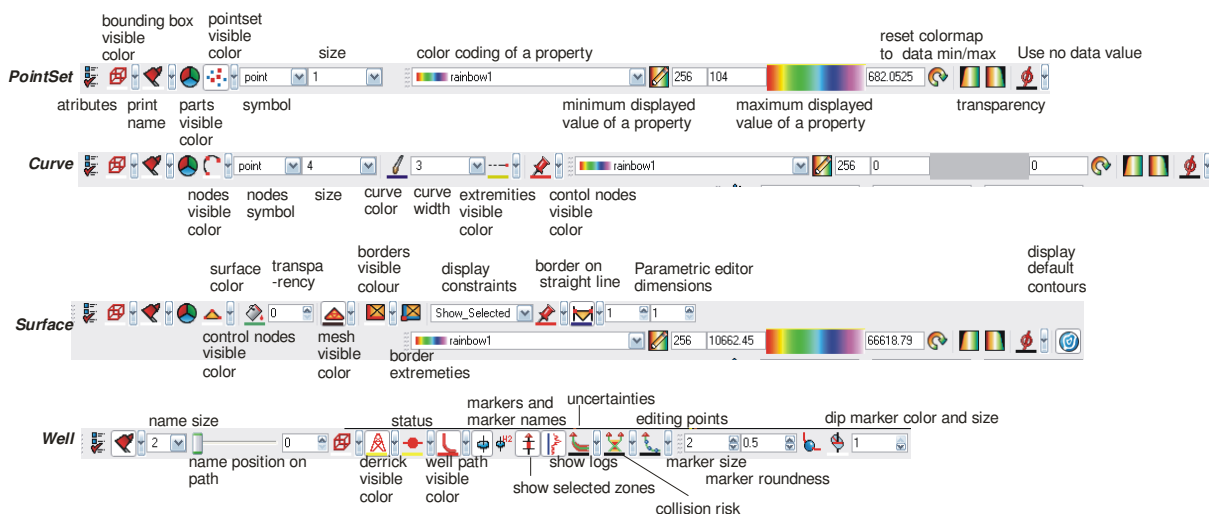


Fig. 2. c. Some Attribute Quick Edit Toolbars

2. 3. General Terms and Features

Attribute: Attributes include all graphical properties of objects, like color.

Coordinate System: gOcad uses a cartesian and metric coordinate system like UTM. This means, all coordinates describing the geometry of an object are interpreted as meters (or feet). The units and the direction of the Z axis (vertical up or down) can be specified in the start-up project specification dialog (fig. d). Objects can be represented in time or depth. Attention!! The definition cannot be changed later!

Command History: The log of all gOcad commands executed in your session appears, when you click on the small triangle symbol left above the Attribute Quick Edit Bar (fig. e). The history is also stored in the file "history.gsc" in the current project directory.

Objects: represent geological objects and input data, like PointsSets, Curves, Surfaces, SGrids, and Wells. Objects can possess numerical and geological properties and subregions. Each object has an unique name (without whitespaces). The property at a point can be picked using the "XYZ?" icon in the Camera Toolbar (8). The result appears in the status bar.

Operating system and Plugin development: gOcad is programmed in ANSI C++ and allows to develop custom plugins. Qt is used as a widget library for user interfaces (www.trolltech.com). Thus, gOcad is available for different OS: MS Windows, Linux (PC), Irix, Solaris.

OK-Apply-Cancel Buttons: Note that OK executes a command and closes a dialog, while Apply keeps it. Sometimes you have to be careful and not press Apply + OK instead Apply + Cancel, as the command would be executed twice. Particularly, cut operations may cause troubles.

Node, Point, Atom: Point object that are used in a different context. Points are nodes, if we deal with topological objects and as Points if we deal with geometry. Atom store topology, geometry and constraints.

Control point: Data point controlling the geometry of the interpolated surfaces, can be weighted by priority, is stored with the gOcad object.

Control node: must be considered by the interpolation. It is always part of the surface and stored with the gOcad object.

Border: All triangle edges tha are not shared by two triangles

Border extremities: are border stones subdividing a border.

Part: Some object types can be composed of isolated topological parts, for example, a surface object can be composed of several disconnected triangulated surface parts.

Property: Numerical properties, including the geometry, are generally stored as 32bit-float numbers at the nodes or cells of objects. Properties can be scalar or vectorial. For example, orientation data can be stored as vectors.

Region: A region is a named subset of nodes of a discrete model object. This can be used for selective object editing or data filtering.

Storage: gOcad objects can be stored either as ASCII files : *File →Save Objects and File →Save Objects as . . . or as binary gOcad Projects File →Save as Project . . .*

Wizards: Based on the command history of a gOcad session, it is possible to create text files containing gOcad commands in a workflow-like order. gOcad creates neat dialogs in a workflow-like style at runtime. Wizards can be accessed via *Applications →Wizards*.

Workflows: Plugins which guide the user through common tasks, for example Structural Modelling or Property Modelling.

DATA PREPARATION AND IMPORT

3. 1. Introduction

This excersices deals with the preparation of primary data of different type and coordinate system into gOcad compatible file formats and with the import of these files in gOcad. The modelling area is the Matterhorn summit in the Swiss Alps. The data set comprises an elevation raster dataset in WGS 1984 coordinates, a topographic map in CH1903 LV03 coordinates and a geological map with unkown spatial reference. Since the CH1903LV03 coordinate system is metric and cartesian, we can directly use it in gOcad and want to transform all the other data to this coordinate system. We want transform the elevation model into a Point Feature and then in a txt file storing the posistion and elevation of each data point. This file can be imported to gOcad and the ground surface can be modelled from the elevation data. Additionally, we want to digitize the main thrust and the border between para- and orthogneisses with the objective to importvthese two geological boundaries to gOcad.

3. 2. Digital elevation model

3.2.1. Preparation

If you want to model the topography the Earth's surface in gOcad, you have to import a digital elevation model either in a point vector or in a text format. Our DEM is provided as a raster image in WGS1984, which cannot be used in gOcad. Therefore, we have to perform a data and a coordinate transformation in ArcMap.

If the resolution of the DEM is very high, so that you would get too many data points for gOcad, this can be reduced by resampling of the image. During resampling, the size of the pixel cells is changed, but the extend of the image remains the same.

Data Managment Tools → Raster → Resample

The DEM can be clipped on the size of the geological map. Since this is given in CH1903 LV03 coordinates, we reproject our dataframe in Swiss Coordinates.

Data frame properties → Coordinate system → predefined → Projected Coordinate Systems → National Grids → CH1903 LV03

Then, we can clip the raster image to the extend of the geological map:

ArcToolbox → Data management Tools → Raster → Clip

Input Raster: dem_resample

Rectangle → Enter Y Max, Y Min, X Min, X Max

Output Raster dem_resample_Clip.tif (save as Tiff-file!)

Finally, we transform the raster to a point feature.

ArcToolbox → 3D Analyst → Conversion Tools → From Raster → Raster to Point → Input Raster: dem_wgs

If this tool has not been used, you have to activate it

Tools → Extensions → tick 3D Analyst.

Before exporting the DEM shapefile to gOcad, it has to be saved in CH1903 LV03 coordinates.

Click on dem → Data → export data → use the same coordinate system as the data frame → dem_chs.shp

Then, you can save the point feature as txt-file:

*ArcToolbox → Spatial Statistics Tools → Utilities → Export Feature Attribute to ASCII → value field: GRID-CODE
→ Delimiter: space → Output ASCII file: dem_chs.txt*

If you look at the txt-file, you see that the point coordinates are saved in CH1903 LV03. If necessary, replace comma by point.

3.2.2. Data import

Import the dem_chs.txt file with

File → Import Objects → Horizon Interpretations → PointsSets → XYZ

If you want to see the imported objects, you have to open the object panel and the pop-up menu PointsSets. Then tick dem_chs. If you still don't see the points, you have to set the 3D camera to show the newly imported objects. Use the Global View auto setup icon in the Camera Settings Bar (4) or click in the camera and press the "End"-key. Now, you can change the view by moving the mouse. Finally, have a look at the front view by clicking in the camera settings bar. Save the gOcad PointsSet with

right click in the object panel → save.

3.3. Georeferencing of maps

Open ArcCatalog and go to the folder which contains the Matterhorn data set. The folder contains an aerial photography of the Matterhorn, „matterhorn_topo.tif“ with isohypse lines and coordinates. The raster properties are displayed with a right click on the raster image. The properties include information about the data source, extent, spatial reference, number of columns, rows and bands, the cell size, format and pixeltype. Of course, the map does not have a spatial reference and the cell size and extent is not correct. The image comes with a graticule and the projection information. The projection is Swiss Grid or CH1903 LV03. We want to georeference the map in this coordinate system. The map can be imported to a GIS project with the „add“ (+) button in the menu bar. Make sure that your dataframe is projected in CH1903 LV03 prior to adding the map. A click to the little globe icon („full extend“) zooms to the imported object.

First, we have to define the georefernce system of the map.

*Tools → Data Management Tools → Projections and Transformations → Define Projection → matterhorn_el.tif
Coordinate System → Select → Projected Coordinate systems → National Grids → CH1903 LV03*

Now, you can define control points with

Georeferencing toolbar → add control points

*Add control points → left mouse click on a coordinate cross, right mouse click → window: enter coordinates → enter
x and y coordinates → ok.*

Georeferncing is performed with *Georeferncing → Transformation → affine*

And the rotation of the map to the horizontal plane with *Georeferencing → Rectify*

After georeferencing, you may save the control points with *View Link Table → Save*

The georeferencing of the map is saved with *Georeferencing → update Georeferencing.*

In the next step, we have to georeference the geological map. Since the coordinates on the map are unreadable, the map has to be georeferenced on the topographic map using prominent points that are visible in both maps.

*Tools → Data Management Tools → Projections and Transformations → Define Projection → matterhorn_el.tif
Coordinate System → Select → Projected Coordinate systems → National Grids → CH1903 LV03*

*Georeferencing toolbar → add control points → left mouse click on a prominent point in the geological map left
mouseclick on the same point in the topographic map*

Georeferencing → Transformation → affine

Georeferencing → Rectify

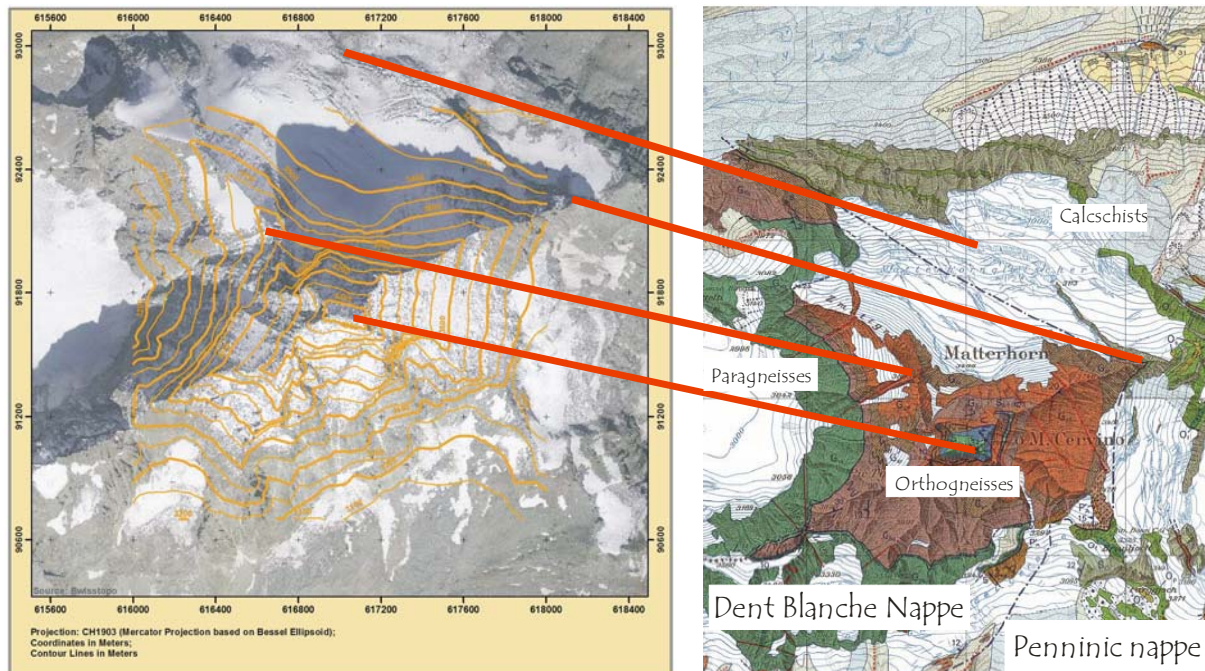


Fig. 3. a. topographic and geological map of the Matterhorn and a few prominent points that are suitable for georeferencing the geological map on the topographic map

3. 4. Create a shapefile from the map

3.4.1. Preparation

Next, we want to digitize the fault separating the Dent Blanche and the Penninic nappe and the border between ortho- and paragneisses (green and the brown units west and south of the Matterhorn summit). Open ArcCatalog, create a shapefile „fault“, with line features and define CH1903 as coordinate system. Add the shapefile to the matterhorn project and digitize the geological objects.

Arc Catalog → right click on Matterhorn → New Shapefile → name: fault → feature type: polyline → spatial reference → edit → Select → Projected coordinate system → National Grids → CH1903 LV03 → add → ok → ok

ArcMap → Add fault.shp

Editor Toolbar → Start Editing → Sketch Tool → Digitize the fault with left mouse clicks → finish with double left click → Editor Toolbar → Stop Editing → Save

3.4.2. Data import

Now, we want to import the shapefiles in gOcad.

Main menu → file → import objects → cultural data → ArcView shape

If you want to see the imported objects, you have to open the object panel and the pop-up menu curves. Then tick the objects.

Save the objects with a right click on the object name in the object panel → save

3. 5. Georeferenced pixel images

3.5.1. Preparation

gOcad allows the import of georeferenced pixel images in the worldfile and geotiff formats. Export the geological map in worldfile format and the geotiff formats!

File → *export map* → *tick write world file*

File → *export map* → *format* → *write geotiff tags*

The worldfile format contains 6 lines that can be read with a click on the filename and the F3 button. The numbers exhibit the cell size, the rotation vector and the coordinates of the left upper corner. If lines 2 and 3 are not zero, the image is rotated. The rotation has to be corrected before importing the file to gOcad. Use *Georeferencing* → *Rectify*.

3.5.2. Data import

File → *Import Objects* → *Images* → *Georeferenced images* → *WorldFile Image oder GeoTIFF Image*

3. 6. Seismic sections and geological cross sections

Images of many different fomats can be loaded - but we still need to georeference them as the origin is by default xyz = (0, 0, 0) and the step vectors have unit length. We want to import a cross sections in TIF format and locate it along the profile line profil.pl.

File → *Import Objects* → *Images* → *As 2d Voxet "profil.tif"*

File → *Load object* → *profil.pl*

Now, we can read the coordinates of the cross section corners with the tool „Get XYZ coordinates“ and assign these coordinates to the image. First, we have to determine the positions of the origin, point u, point v and point x, which is possible if we know the vertical scale of the image. Next, the image has to be fixed in 3D space by point w which is coincident with the origin. The coordinates are given in table a.

voxet → *tools* → *resize with points*

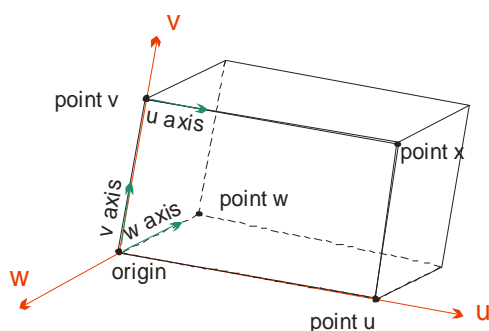


Fig. 3. b. Illustration of points and axis used for resizing an image in Gocad.

	X	Y	Z
Origin	615433	93031	3000
point u	618057	91169	3000
point v	615433	93031	4400
point w	615433	93031	3000

Table 3.1. Coordinates for resizing the cross section.

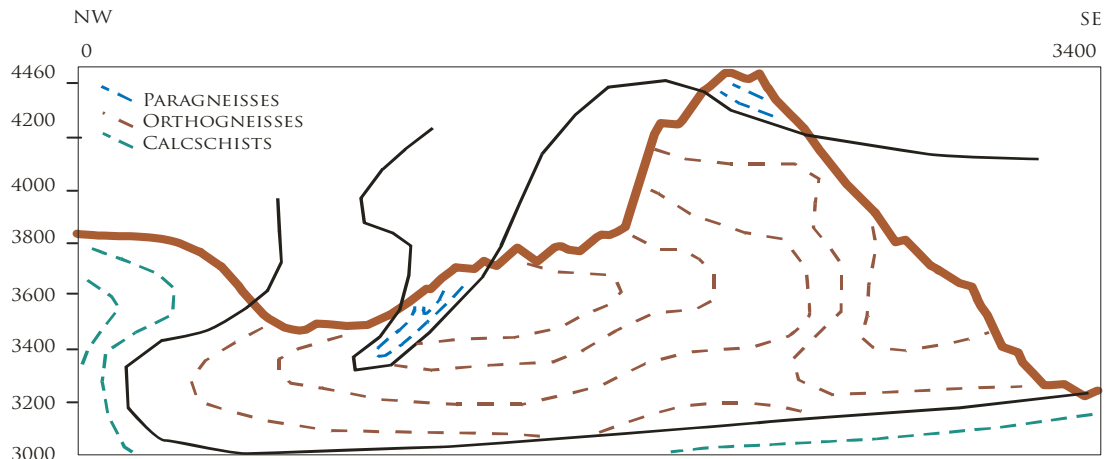


Fig. 3. c. Cross section through the matterhorn showing the main rock units after Marthaler (2008)

The imported cross section is located at the correct position, but it is plane sheet, while the profile is curved. This means, that we have to project the section onto the profile. We can perform this projection by creating a vertical surface through the profile line and by draping the voxel on this plane.

3. 7. Well data

3.7.1. Preparation

Wells are commonly stored in relational data bases with different formats. Some generic ASCII formats for well import including well path, markers and logs can be imported to Gocad, and also standard formats like WITSML and OpenSpirit.

In the Gocad Suite, the depth information of a well is stored in the properties : Z and Zm: Z is the True Vertical Depth Sub Sea (TVDSS). TVDSS is defined as the vertical distance from a point in the well to normal height null. Z can be positive or negative depending on the orientation of the project (upward: negative, or downward: positive). Zm is the Measured Depth (MD). The Measured depth is the length measured along the wellbore. Whatever the orientation of the project, Zm is always positive.

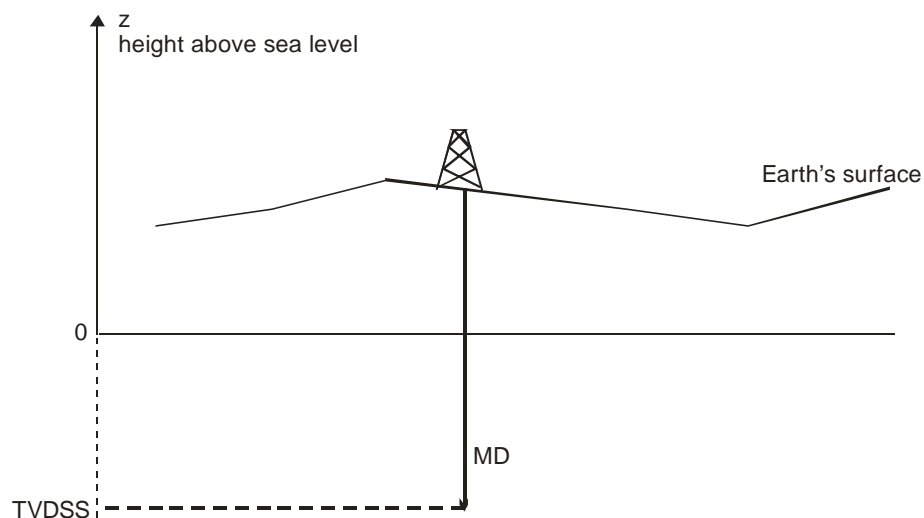


Fig. 3. d. Difference between the true vertical depth (TVDSS) and the measured depth (MD).

The well marker file has to be formatted with the XYZ coordinates for each well marker. Save the wellmarkers as ASCII file and don't forget to replace commas by points!

Well	X	Y	Z	Marker
1	616451.5156	91913.87207	3655	orthogneiss
1	616451.5156	91913.87207	3410	calcschist
1	616451.5156	91913.87207	3211	end
2	617268.2563	91336.05493	3622	orthogneiss
2	617268.2563	91336.05493	3303	calcschist
2	617268.2563	91336.05493	3195	end
3	617686.6226	91742.36584	3632	orthogneiss
3	617686.6226	91742.36584	3114	calcschist
3	617686.6226	91742.36584	3033	end
4	616721.1748	92277.81812	3605	orthogneiss
4	616721.1748	92277.81812	3124	calcschist
4	616721.1748	92277.81812	3082	end

Table 3.2. Layout of a wellmarker file.

3.7.1. Data import

Well paths and markers are imported separately from ASCII-files using

File → *Import objects* → *Well Data* → *Paths...Column-based file* → *current file* → *choose the file name* → *Delimited* → *Next* → *Press on preload all if not 100% of the file are preloaded* → *Next* → *What information do you have for the path* → *tick X-Y-TVDSS* → *To name the objects* → *use a column* → *Next* → *Start line 2* → *end at line* → *type the last line* → *Click through all entries in the property list and type the column number for each* → *ok*

File → *Import objects* → *Well Data* → *Markers...Column-based file* → *current file* → *choose the file name* → *Delimited* → *Next* → *Press on preload all if not 100% of the file are preloaded* → *Next* → *Do you have a MD column* → *yes* → *next* → *Do you have Dip and Azimuth information?* → *No* → *Next* → *If a marker already exists* → *add it as a new one* → *Next* → *Start line: 2* → *End line* → *number of the last line* → *property list and parameters* → *Click through all entries in the property list and type the column number for each* → *ok*

The paths are imported with the wellname, X, Y and TVDSS values. The name of the objects should be used from the column containing the wellname. This is necessary to link path and marker. Tick all well marker in the *object* → *panel well* to make them visible.

3.7.2. Group Well objects

As you can see, you have to tick every single well marker to make it visible. Usually, you will work with many wells simultaneously. To make this more comfortable and less time consuming, you can group the wells and manipulate the visualization of all of them as if they were a single Gocad object.

Edit → *New Group* → *name: wells, object type: well; Edit* → *add to group* → *select: 1 2 3 4*

3. 8. Options for the data import in gOcad

3.8.1. Import of data files

Data files can be imported with the menu *file* → *import data*. Pop-up menus offer many options for the import of various data types in various file formats. After import, the data set is transformed to a gOcad object and can be saved.

3.8.2. Load gOcad objects

This tool provides the possibility to restore gOcad objects that have been saved by the command *Object Panel* → *gOcad object* → *save*

3.8.3. Import objects from project

This command allows loading gOcad objects with all linked data from another gOcad project. Some objects like surfaces with control points, wellmarker-wellpath objects or textures have to be imported with this tool and cannot be restored with the *save* → *load object* tool.

POINTSSETS, CURVES AND T-SURFACES

4. 1. PointsSets

4.1.1.Characteristics of PointsSets

A PointsSet is the most primitive gOcad object. Numerical properties and regions are defined for the whole PointsSet object and stored with each point. For example, each point stores its XYZ coordinates and an arbitrary number of additional scalar or vectorial properties. Each point may contain flags indicating whether it belongs to subset regions of the PointsSet. Commonly, PointsSets are used as input data, like sampling points with orientation, geochemical or petrophysical measurements, or simply location data for terrain models. PointsSet gOcad objects are characterized by a *.vs file extension.

We will explore the characteristics and the manipulation of PointsSets using two data sets. The data set „WellPoints.vs“ consists of 861 data points comprising spatial and property values at each sampling point. By observing the data set in the 3D camera, you get a first impression of its spatial distribution: About 12 data points are located at an identical XY position that is coincident with one vertical well. The whole data set reproduces the geometry of an anticline with a curved hinge line. The data set describes a structural oil trap. Each data point comprises the properties core porosity, log porosity and permeability. Create a gOcad project and import the PointsSets as column-based file.

File →Import Object →Horizon Interpretation →PointsSets → Column-based file → current file:wellpoints.dat →delimited →next →next →Do you have X,Y,Z column? →yes,yes,yes →to name the object →use file name →next →Preload all →Start at line: 2 →End at line: tick the arrow to the right of the field and the last line → Property list and parameters: →X →column 1 →Y →column 2 →Z →column 3 →add →porosity column 4 →logporosity →column 5 →permeability →column 6 →advanced →domain →depth →ok

If you tick the object name wellpoints in the object panel, the Attribute Quick Edit Bar below the object panel changes dynamically and you can set the symbol type, color and size.

4.1.2. Display properties of a PointsSet

If you want to visualize properties, just choose one and highlight its name. Again, the Attribute Quick Edit Bar changes and you can assign appropriate graphical property attributes, like color maps. Another way to manipulate graphical attributes is to click on the object name with the right mouse button and to enter the attribute window.

Color coding of properties

Object panel →PointsSets → pop-up menu wellpoints → open pop-up menu properties →tick the property you want to get displayed.

You can change the layout of the points and the color bar for the properties in the Quick Edit Bar appearing when you click on the selected property. You can also read the minimum and maximum values of the property. The colored arrow resets the color map to the minimum and maximum values of the PointsSet.

Let's have a look at the areal distribution of the data set:

Cross Plot →object: WellPoints →region: entire object →X-Axis property: X →Y-Axis property: Y →use equal minimum and maximum values for both axes and a linear scale

Simple statistics functions

If you want to get a better understanding about the distribution of your data, you may employ simple statistic functions with

compute → statistics → histogram or cross plot 2D

Let's check all three properties with the histogram tool and let's answer the following questions:

How are the properties distributed (uni- or multimodal, symmetric or asymmetric)? Which characteristic numbers do they have? Do you find outliers?

Histogram → object: WellPoints → region: everywhere → property: coreporosity → read the characteristic numbers in the first box on the left side → always play with the number of bins change to 100 decrease to 20 to see whether new peaks appear → interpret number of modes, skewness, geometry of the distribution

The porosity has a bimodal distribution with two peaks. This phenomenon is either an indication that the data set includes multiple geological units like clay lenses in a sandstone or that rock formation was affected by secondary processes like cementation of pore space.

Instead, the permeability that is expected to be a function of the porosity, is skewed to the right and looks like a log-normal distribution. Therefore, let's change the scale to log10 and have a look at the data. Now, we observe again the bimodal distribution found in the porosity histograms. Another thing is important in the permeability set: We notice outliers with extremely high permeabilities. This may indicate the existence of a thief zone in the reservoir. A thief zone is a very permeable formation encountered during drilling into which circulating fluids can be lost. Thief zones, which include crevices, caverns and porous formations, must be sealed off with a liner or plugged with special cements or fibrous clogging agents before drilling can resume. Therefore, it is important to discover and to locate the thief zone. Where is the thief zone located?

You can check it quickly using the color map clip value and transparency settings.

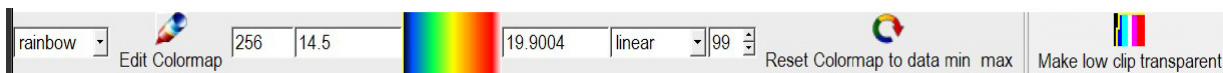


Fig. 4.a. Part of the Attribute Quick Edit Bar allowing to make transparent a range of values

The bivariate data analysis deals with the relation between two properties and can be performed with the cross plot tool that can create and scale cross plots. Additionally, Pearson's correlation coefficient and the linear regression coefficient are displayed. For better data analysis, the data have to be exported to statistics software like S-Plus or R.

How are the properties related to one another? Plot the cross plots and scale them, change the scale from linear to logarithmic, if necessary.

You see that the core porosity and the log porosity are highly correlated, but are not equal. The core porosity is systematically higher than the log porosity, which may be caused by expansion of the core after its exhumation. Since the log porosity was measured in situ, we assume it to be more realistic and will work with it. If you print a cross plot between log porosity and permeability, you find that there may be a functional relation between both properties, but that this relation is not linear. This is evidenced by a difference in the correlation and rank correlation coefficient (fig. d). If you logarithmize the permeability, the correlation coefficient is increasing considerably, however, the regression line is still somewhat asymmetric, which is caused by the outliers pulling the regression line upward. Create a region „without_outliers“ and create a new regression model. Now, you see that it is more symmetric and the correlation coefficient as well as the coefficient of determination are better.

gOcad offers a tool to view 3D cross plots, but a real multivariate data analysis has to be performed in an external statistics software.

1.1.3. Define Regions

If you are interested in values of certain range or want to query point properties, you may select them and assign them to a region. Create and initialize a region containing the permeability values that are bigger than 600!

Region →Create →object: WellPoints →region: thiefzone

Region →Initialize →from property range →object: WellPoints →region: thiefzone →property: perm →use min: 600 →ok

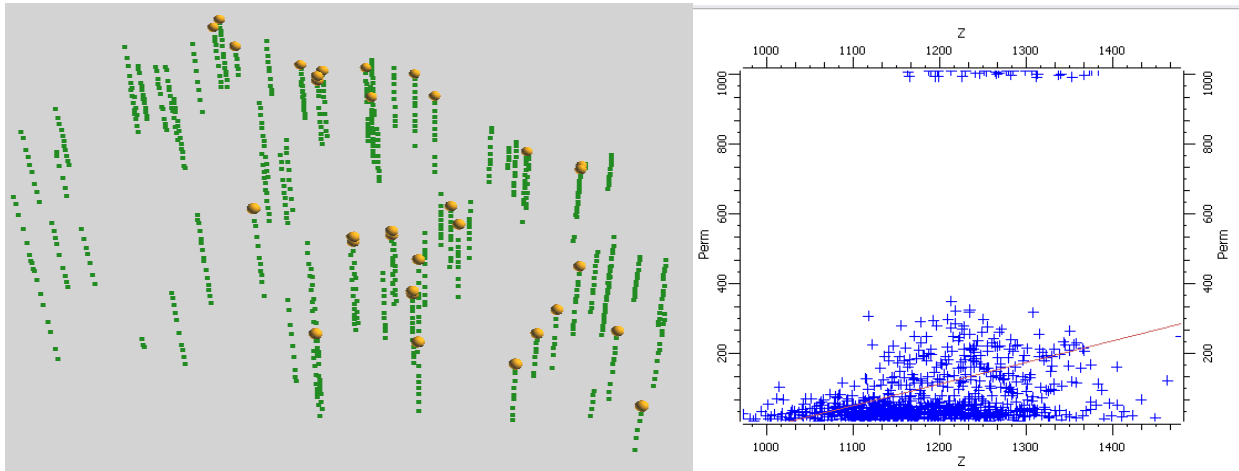


Fig. 4.b. Data points (green) and points with extremely high permeability (red). All points with high permeability are located at the top of the data set. However, this is not visible in a cross plot, because of the anticlinal structure of the sampled formation.

1.1.4. Apply scripts on gOcad objects

Now we load a different type of data, a simple ASCII file „DEM.txt“ containing points representing terrain elevation data (X and Y in m, Z in dm).

File →Import Objects →Horizon Interpretations →PointsSets →XYZ →DEM.txt

You don't see the data in your 3D camera although the PointsSet is ticked in the object panel? The 3D camera is still focused at the wellPoints, and we need to set it up to show the new object which is located somewhere else. Deselect wellPoints in the object panel and click on the global view icon in the Camera Settings Bar (4). In order to represent the object in proper geometry, we need to convert the Z values to meters.

Right click at DEM in the object panel →copy →dem

point set mode →compute →apply script →on Object

Select dem as object and enter the main script: “z=z/10;”. →Check →ok

Each script ends with “;”. You can check the syntax of the script by clicking “Check”, and watching the gOcad status bar, and then click either OK or Apply and Cancel. Why? OK executes the command and closes the dialog, Apply executes the command and keeps the dialog. Pressing Apply and then OK would result in twice executing the command, which is not wanted in our example.

Property scripts are very useful tools for various tasks and can be applied for all gOcad objects likewise. A rich set of mathematical functions is available. For example, scripts may be used to compute saturation values in SGrids:

$$Sh = 1 - \sqrt{30 / (\text{resistivity} * \text{pow}(\text{porosity}, 2))};$$

1.1.5. Digitize new points and apply properties

Now, we want to digitize additional points of the same topographic height.

First we want to interpolate our first surface with

Surface →New →From PointsSets →PointsSet →name: relief_dt →atom set: dem →ok

Open the pop-up menus relief →properties →click on Z

Now you can see the Quick Editing Bar.

Click on Create default isovalues on the right side of the bar. Turn the camera to “top view”

Start digitizing one isoline:

Digitizing Toolbar → Digitize Points

Digitize with left mouse clicks. Finish digitizing with a right mouse click and enter a name for the new point set.

Create a property isovalue and assign a constant value!

Property → Create → object: digitized pointsset → property: isovalue

PointsSet → Property → edit → set constant

4. 2. Curve objects

4.2.1. Characteristics of curve objects

Curve objects are points connected by line segments. Files containing gOcad curve objects have the extension *.pl. In contrast to PointsSets, curves can be projected on surfaces.

File → Load Objects → "curve_topology.pl" → Choose view from top

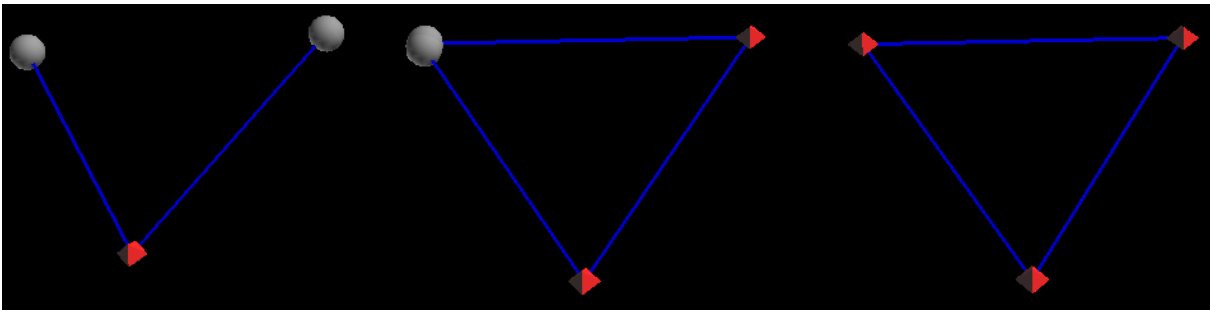


Fig. 4.c. Curve Topology: sphere: border extremity, octaeder: curve node

The fig. a shows the curve object *curve_topology*, which you can load into gOcad: It is composed of three isolated parts. The left and the right part consist of 3 nodes, while the middle part consists of four nodes. The difference is topological: The left part has two extremities (grey spheres): two nodes have only one neighbour. In the middle part, two extremity nodes have the same geometry, but are not connected. If you are curious, you can explore how the topology is stored by having a look at the file in an editor.

4.2.2. Import of curve objects

Now, import the shapefile *elevation.shp*!

Main menu → file → import objects → cultural data → ArcView shape

The shapefile contains several properties: XYZ, Id and Elevation. You see, that the Z values of all line objects are constant and that you look at isolines in a map. The height of each isoline is saved in the property Elevation, and if we want to build a 3D model, we have to assign the elevations to the Z values.

PointsSet → compute → apply script → on object → main script Z=Elevation;

4.2.3. Editing of curve objects

You can edit the curve object with the tools offered in the *Curve → Tools* menu.

You can drag one node with the mouse with

Curves → tools → node → drag with mouse (you can repeat this function by pressing the space button)

Curve → tools → segment → split allows to add a new node

Curve → tools → node → disconnect allows to divide the curve

Curve → tools → node → bridge allows to connect to parts of one curve

Curve → tools → Node link → links extremities → reconstruct curve from links allows to close a curve.

4.3. Triangulated surfaces

4.3.1. Surface topology

Load the gOcad object `surface_topology_parts.ts!`

A surface object comprises several constituents: triangles, nodes, borders, border extremities and constraints (fig. c). You can control the display these elements in the Quick Attribute Toolbar, when you tick one surface object in the object panel and leave the object highlighted. A surface object can be composed of several isolated parts, each having one border. The cut creates two topologically isolated parts. The borders can be subdivided by border extremities. If parts are merged, the border between the parts disappears and the border extremities can be deleted.

Surface → Tools → Parts → Merge All

Surface → Tools → border extremity → delete → pick extremities in the 3D camera

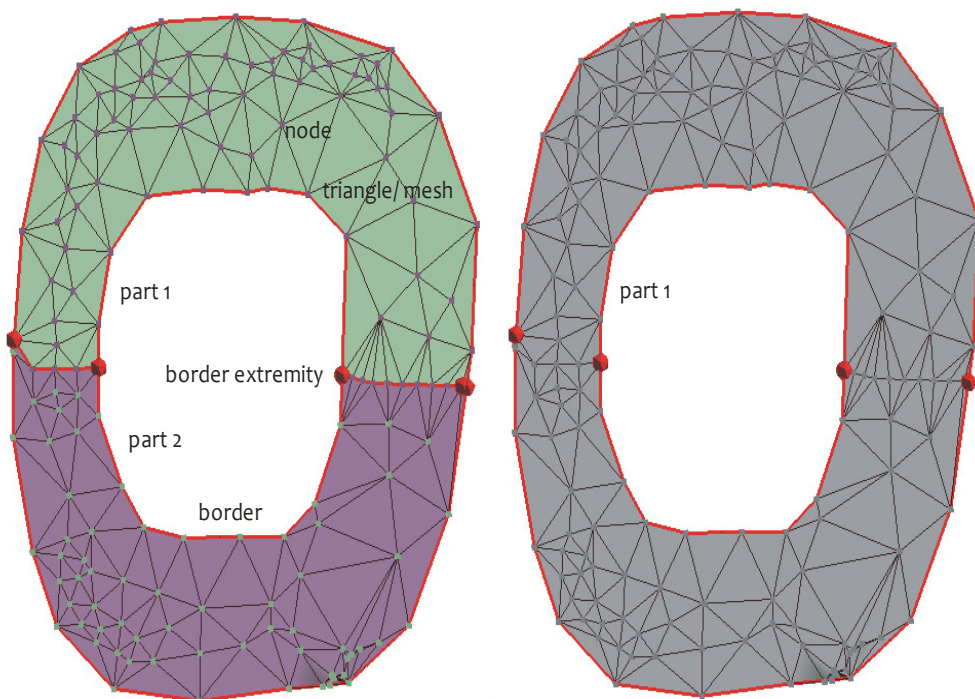


Fig. 4.d. Topological elements and parts of surfaces. After a cut, nodes along the cut line are duplicated and belong to a border. The nodes of the borders have the same location but are not connected. Each border has at least one border extremity. If the surface parts are merged, the triangles on both sides of the borders are connected and the border disappears.

4.3.2. Methods of Surface Interpolation

gOcad provides many tools to interpolate the surfaces of geological objects and to honor different data with different weight or certainty. In general, two interpolation algorithms are available: Direct Triangulation (DT) and Discrete Smooth Interpolation (DSI).

The Direct Triangulation is a triangulation, such that no point is inside of any triangle. The interpolation algorithm maximizes the minimum angle of the triangles to avoid skinny triangles. In gOcad, the data points are automatically set as control nodes, if a surface is created by Direct Triangulation.

In order to create geomodels which honor available data, a unique interpolation method is available in gOcad named Discrete Smooth Interpolation. This method allows to interpolate the functions of the discrete model, like geometry or properties, while honoring a set of constraints C_i . Several kinds of constraints can be distinguished. So-called hard equality constraints are control nodes, which are not allowed to move. Soft equality constraints are control points, these are nodes of another object which should be honored in a least squares sense. Other possible

constraints are for example vectorial links between nodes, range constraints, gradient constraints. The DSI algorithm converges towards a solution where R is the local roughness at node α , ρ is a constraint defined for node α , μ is a stiffness coefficient, and ω and ω^* are weight coefficients. This generic method allows the usage of heterogeneous data as constraints for the interpolation. It is always reproducible, on what primary data an interpolated object is based on. In gOcad, a comprehensive set of constraints is implemented. Simultaneously, DSI guarantees that the interpolation result is as smooth as possible.

$$R^*(\varphi) = \sum_{\alpha \in \Omega} \mu(\alpha) \cdot R(\varphi|\alpha) + (\phi \cdot \varpi) \cdot \sum_{c \in C^\approx} \varpi_c \cdot \rho(\varphi|c) \tag{1}$$

4.3.3. Comparison of Direct Triangulation (DT) and Discrete Smooth Interpolation (DSI)

Direct Triangulation yields ragged surfaces, DSI yields surfaces that look like a table cloth hanging over the object (fig. b). Table a summarizes differences between both interpolation methods.

	Direct Triangulation	Discrete Smooth Interpolation
constituents	nodes, mesh, border, border extremities	nodes, mesh, control points, control nodes, border, border extremities
AtomSet coincides with	nodes	control points usually no part of the surface
Extend of the surface	bordered by data set	bordered by min and max coordinates Or clip with bounding box
Nodes	all points of the AtomSet + additional nodes	no relation with AtomSet
Control points	no	points of the AtomSets
Control nodes	Data points or user-defined	userdefined, either nodes or atoms or borders

Table 4.1: Differences between Direct Triangulation and Discrete Smooth Interpolation

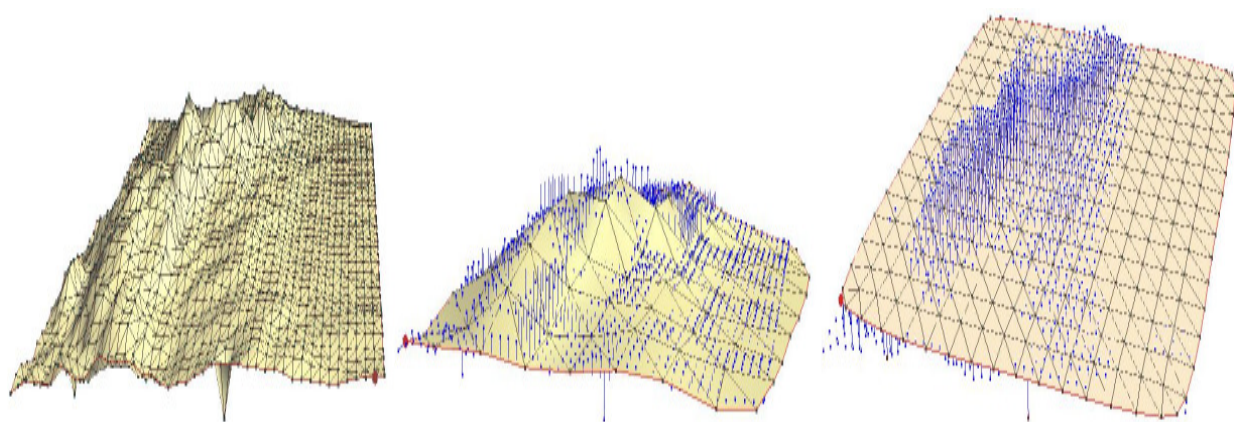


Fig. 4.e. Interpolation of surfaces using the same AtomSet, but three different interpolation algorithms: left – Direct Triangulation, centre – DSI with high fitting factor, right – DSI smooth. Blue lines: Control points for DSI.

4.3.4. Craete a surface from a PointsSet

Direct Triangulation

Surface → New → From PointSets → PointSet → relief_dt → AtomsSet: dem → ok

Now, you can edit the layout of the the surface and display or hide its different elements like nodes, mesh and the border. Click on the name of the surface in the object panel, this activates the Quick Edit Bar.

Display the nodes, the border and the the border extremities, hide the mesh.

Display a colorcode for the Z values of the surface!

Discrete Smooth Interpolation

Surface → New → from PointSets → PointSets Medium Plane → name: relief_dsi → AtomsSet: dem → Clip with XYZ → ok

This tool creates a trend surface with a minimum distance to the points of the AtomsSet. In order to interpolate a surface that follows the shape of the PointsSet, we have to split the surface in several triangles. Then, we set the points of the AtomsSet as control points and interpolate. You can play with the fitting factor in the “Advanced” menu that can be set to 0 for creating smooth surfaces and to a maximum value of 2 in order to honor the data. The fitting factor corresponds to the coefficient μ in the global DSI equation. DSI is an iterative algorithm, you can run more iterations by clicking „Apply“ several times.

Surface → Tools → Split → All → relief-dsi → Aplly+Apply+Apply

Surface → Constraints → Control Points → Set Control Points—AtomsSet: dem

Surface → Interpolation → Geometry → On Entire Surface → dem_dsi → tick smooth → ok

The DSI algorithm tries to minimize the roughness of the surface, and one sideeffect is that it minimizes the area of the surface: If you press several times apply, the surface will shrink (fig. c). In order to avoid this, we have to set constraints on the boundaries of the surface.

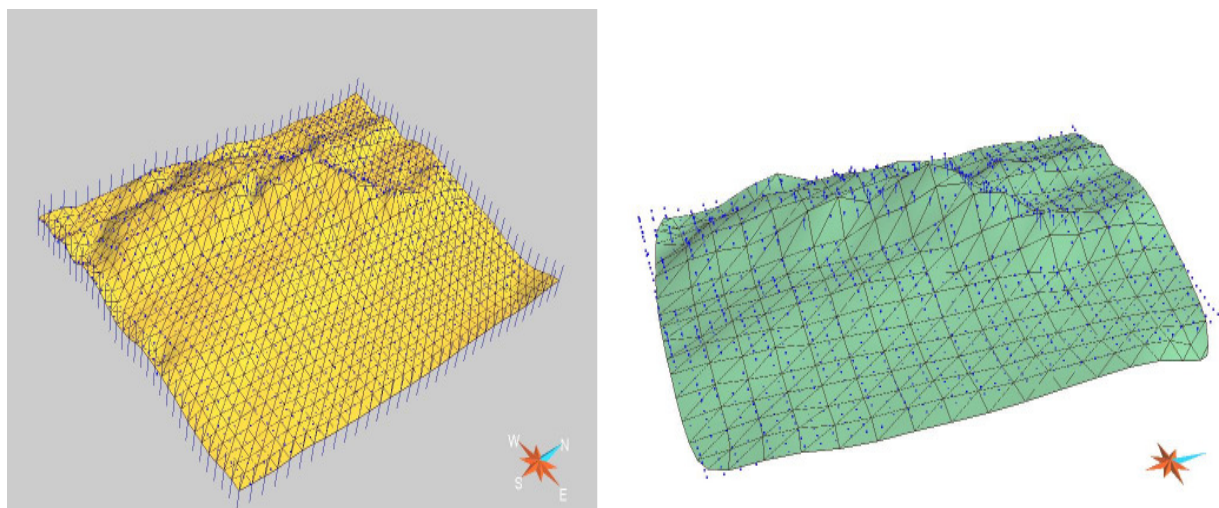


Fig. 4.f. Interpolation with and without constraints on borders. If no constraints on border are set, the corners of the surface are rounded.

Create a bounding box and set boundary constraints

You can crate to types of bounding boxes: one that is surrounding the data set and one that is parallel to the XYZ axes.

Voxet → New → from objects box (fits the geometry of the objects)

Voxet → New → from objects cage (is parallel to the axis)

Now, create a new PointsSet Medium Plane with the name relief_dsi, clip it with the voxet, split it and set control points! Add border constraints to avoid the shrinking of the surface!

Surface → Constraints → Constraints on Borders → All Borders → ok

Interpolate a surface with high fitting factor!

4.3.5. Create a surface from several curves

Direct Triangulation

Display the object „Elevation“ and create a surface from the dataset.

Surface → New → From Several Curves → elevation_dt → press strg-key → select the curves by clicking on them while keeping the string-key pressed.

“Nb of levels” of levels determines how many triangle strips will be inserted between the curves.

The “beautify”-tools allow to improve the triangulation by removing redundant nodes and switching triangles according to the Delaunay criterion. The Delaunay criterion states that the minimum triangle angle is maximized. The various “beautify”-commands are important to obtain a numerically stable surface that can ,f.e., be cutted.

Surface → Tools → Beautify → Beautify Triangles for equilaterlity

Apply this tool several times until you are content with the shape of the triangles!

Discrete Smooth Interpolation

Voxet → New → from objects cage → name: box → scale w dimentsion by 1.2

Surface → New → from PointSets → PointSets Medium Plane → name: fault2_dsi → AtomSet: fault2 fault2_r114 fault2_r115 → Clip with voxet → ok

Surface → Tools → Split → All → muf-dsi → Aplly+Apply+Apply

Surface → Constraints → Control Points → Set Control Points → AtomsSet: fault muf1 muf muf3

→ Advanced → optimize shooting direction

Surface → Constraints → Constraints on Borders → All Borders → ok

Surface → Interpolation → Geometry → On Entire Surface → muf_dsi → advanced → tick conjugate → ok

Surface → Tools → Beautify → Beautify Triangles for equilaterlity

4.3.6. Manipulation of surfaces

You can manually edit a surface object using the commands in the tools menue.

Tools → Node → Move to → click on the blue arrow on the right side of the window and pick the point in the seismic section where you want the node to move to. → apply → pick the node you want to move.

Or enter the X,Y,Z coordinates → apply → pick the node you want to move

Tools → Node → Drag with mouse → pick the node → keep the mouse botton → move the mouse

Tools → Node → Translate → enter the distance in m in each coordinate direction → apply → pick the node

Tools → Triangle → delete → pick the triangle you want to delete

Tools → Triangle → brigde → pick two triangles that border the hole to be closed

SANDSTONE HORIZON

5. 1. Data import

The objective of this exercise is to model a sandstone horizon and the according ground surface from different types of input data. The DEM is given as txt-file, while the geological data describing the sandstone are stored in a well-marker file in txt-format, a seismic section in tif-format and a gOcad curve object in pl-format. Create a new gOcad project and import the well markers and the seismic section. Digitize the base of the third red reflector and load additional data points in the file "sandstone-reflectors.pl". Import the dem.txt-file with

File → Import Objects → Horizon Interpretations → PointsSets → XYZ

If you want to see the imported objects you have to open the object panel and the pop-up menu point sets. Then tick dem. If you still don't see the points, you have to set the 3D camera to show the newly imported objects.

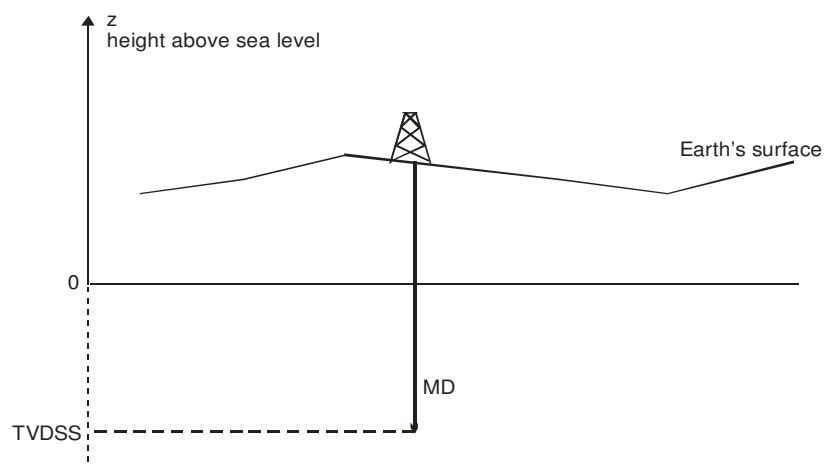


Fig.5.a. Difference between the true vertical depth (TVDSS) and the measured depth (MD).

Well	X	Y	Z	Marker
1	2545.80396	1989.56763	1753.457	quaternary
1	2545.80396	1989.56763	1479.793	sandstone
1	2545.80396	1989.56763	1163.45667	end
2	6272.86035	2085.09595	1602.45667	quaternary
2	6272.86035	2085.09595	1270.678	sandstone
2	6272.86035	2085.09595	1157.333	end
3	4015.66284	4173.15527	1602.45667	quaternary
3	4015.66284	4173.15527	1384.034	sandstone
3	4015.66284	4173.15527	1119.1234	end

Table 5.1: Wellmarker file in txt-format

The wellmarker file (Table a) has to be formatted with the XYZ coordinates for the starting point of the well and for each well marker. The wellmarkers are imported with the columns containing XYZ coordinates, wellname and markername. The depth information of wells is stored in the properties (fig. a): Z and Zm: Z is the True Vertical Depth Sub Sea (TVDSS). TVDSS is defined as the vertical distance from a point in the well to the average sea level. Z can be positive or negative depending on the orientation of the project (upward: negative, or downward: positive). Zm is the measured depth (MD). The measured depth is the length measured along the wellbore. Whatever the orientation of the project, Zm is always positive. Well paths and markers are imported separately from ASCII-files using

File → *Import objects* → *Well Data* → *Paths...Column-based file* → *current file* → *choose the file name:* „well.txt“ → *Delimited* → *Next* → *Press on preload all if not 100% of the file are preloaded* → *Next* → *What information do you have for the path* → *tick X-Y-TVDSS* → *To name the objects* → *use a column* → *Next* → *Start line 2* → *end at line* → *type the last line* → *Click through all entries in the property list and type the column number for each* → *ok*

File → *Import objects* → *Well Data* → *Markers...Column-based file* → *current file* → *choose the file name* → *Delimited* → *Next* → *Press on preload all if not 100% of the file are preloaded* → *Next* → *Do you have a MD column* → *no* → *Do you have X,Y and TVDSS columns?* → *yes* → *next* → *Do you have Dip and Azimuth information?* → *No* → *Next* → *If a marker already exists* → *add it as a new one* → *Next* →

Start line: 2 → *End line* → *number of the last line* → *property list and parameters* → *Click through all entries in the property list and type the column number for each* → *ok*

Tick the wellmarker in the *object panel* → *well* to make them visible. As you can see, you have to tick every single wellmarker. Usually, you will work with many wells simultaneously. To make this more comfortable and less time consuming, you can group the wells and manipulate the visualisation of all of them as if they were a single gOcad object.

Edit → *New Group* → *name: wells, object type: well*

Edit → *add to group* → *select: 1-3*

The paths are imported with the wellname, X, Y and TVDSS values. The name of the objects should be used from the column containing the wellname. This is necessary to link path and marker.

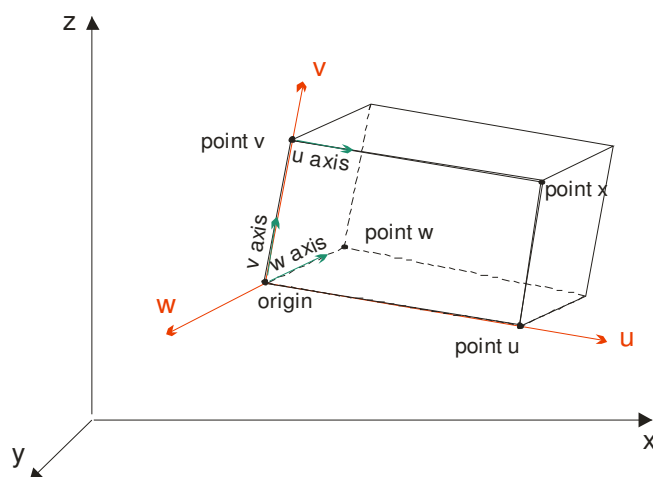


Fig. 5.b. Illustration of points and axis used for resizing an image in gOcad.

Images of many different formats can be loaded, but we still need to georeference them as the origin is by default $xyz = (0, 0, 0)$ and the step vectors have unit length. In order to place the image properly, we have to determine the XYZ coordinates of the upper corners and the depth of the section. So, we can calculate the position of all four corners (point v, point x, origin and point u). In order to place the image properly in 3 dimensions, we have to calculate the position of point w as cross product of the u and v_axis (Table a, fig. b):

$U_axis = point_x - point_v$
 $V_axis = point_v - origin$
 $W_axis = u_axis \times v_axis$
 $Point_w = origin + w_axis$

File → Import Objects → Images → As 2d Voxet → „Seismic.tif“

voxet → tools → resize with points → insert the coordinates from the table

	origin	point u	point v	point w
x	500	9500	500	500
y	1000	1000	1000	1001
z	1600	1600	900	1600

Table 5.2: Coordinates for resizing the image of the seismic section

Now, we will create a new curve by digitizing a seismic reflector of the seismic section. Use the third red line from top. Set the view settings on view from south.

Object Panel → Objects → Curve → From Digitized Polyline

Digitize with left mouse clicks. Finish digitizing with a right mouse click. Enter the name „sandstone_seismic“.

If you are not satisfied, you can edit your Curve geometry or topology.

Curve mode → Tools → Node/Segment.

For example, we see one seismic line digitized with two parts in SeisCurves. We can connect the extremities by clicking on them, and the two Parts are merged:

Curve → Tools → Node → Bridge

Merge the curve objects SeisCurves and sandstone_seismic to one single gOcad object containing all three curve parts (fig. c):

Object Panel → New → From Curve Parts → name: AllCurves → ok → hold str-key → tick the three curve parts with the mouse

5. 2. Modelling the sandstone horizon

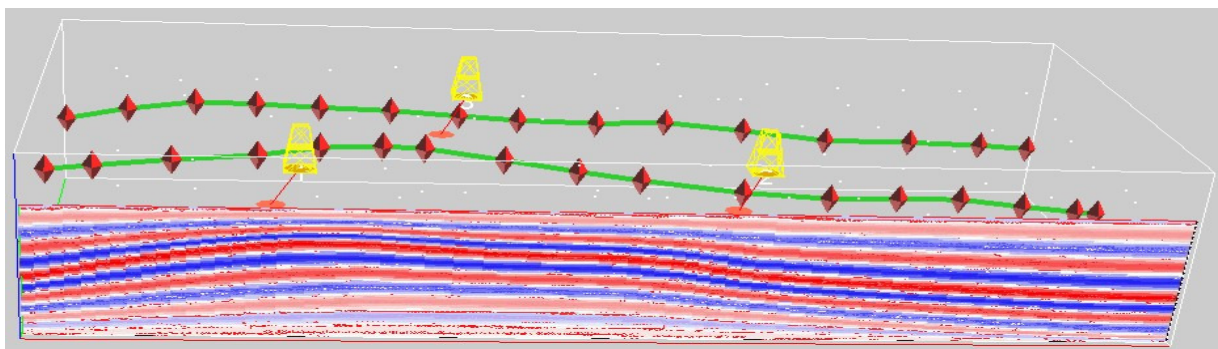


Fig. 5.c. The data set

5.2.1. Direct Triangulation

The Direct Triangulation is a triangulation that connects all data points such that no point is inside of any triangle. In gOcad, the data points are automatically set as control nodes if a surface is created by direct triangulation.

Using the curve AllCurves, we will create a new surface. Pick the three curve parts in the camera while keeping the Strg-key pressed.

Surface → New → From Several Curves Name: DirectSurf, nb of levels: 4

The argument nb of levels determines how many triangle strips will be inserted between the copied nodes of the curve parts, a low value would result in very narrow triangles and thus a rough surface. We can improve the mesh quality by removing redundant nodes and switching triangles according to the Delaunay criterion. The Delaunay criterion states that the minimum triangle angle is maximized, which results in a smoother surface.

The various beautify commands are generally important to obtain smooth and numerically stable surfaces.

Surface → Tools → Beautify → Beautify Triangles

Now, you can explore the graphical capabilities associated with surfaces using the Attribute Manager. Colorize DirectSurf by its Z property values (fig. d).

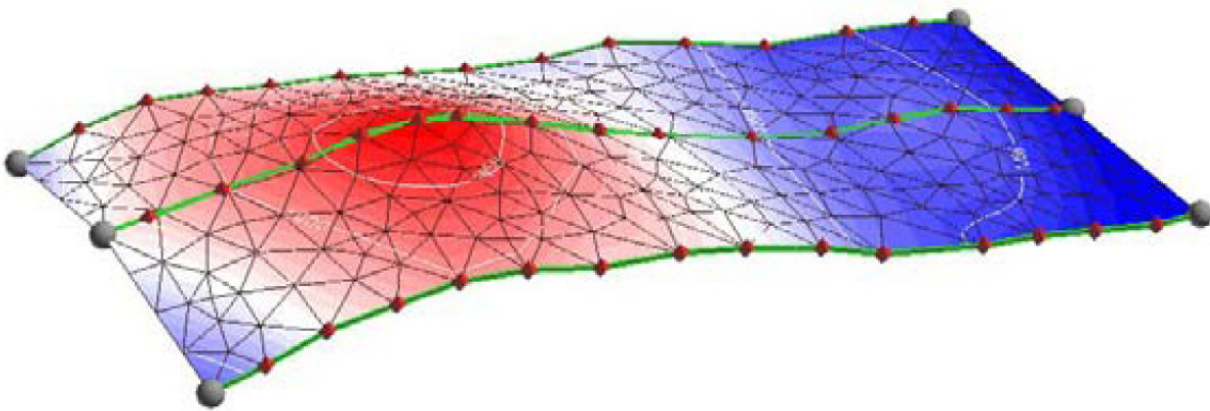


Fig. 5. d. Interpolated sandstone horizon with color-coded property Z

5.2.2. Discrete Smooth Interpolation

The DSI method allows to smoothly interpolate properties while taking into account constraints. We have to create a trend surface with a minimum distance to the curve nodes, and then improve this surface. First, we want to create a bounding box that allows to scale the size of the interpolated surfaces. You can either create an object box with boundaries parallel to the XYZ coordinates or an object cage with boundaries parallel to the boundaries of the data set.

Create a bounding box

Voxet → New → From Objects Box → Name: box → Objects: choose all → Scale Z dimension by 1.1 → ok

Surface → New → From PointsSet Medium Plane Name: DsiSurf, AtomSet: AllCurves → Clip with Voxet: box

The new plane surface consists of only two triangles; we need to discretize appropriately in order to fit it to our data:

Surface → Tools → Split All Apply+Apply+Apply+OK, or OK + 3x space key

Then we set the nodes of the AllCurves as control points:

Surface → Constraints → Control Points → Set Control Points

The DSI method has the side effect that it minimizes the Surface area, thus we set a constraint on the border which fixes the border nodes in the XY plane:

Surface → Constraints → Constraints on Borders → Set On Straight Lines → All Borders

Now, we can interpolate the surface

Surface → Interpolation → Geometry → On Entire Surface DsiSurf → Apply

5. 3. Honor the well data as control nodes

5.3.1. Discrete Smooth Interpolation

gOcad allows to incorporate additional data, in our case the well data, to update the model. However, the well markers cannot directly be applied as control nodes, but have to be converted into PointsSets and have first to be applied as control nodes.

Object Panel → *PointsSet* → *New* → *From Well Markers* → *name: sandstone* → *Marker name: sandstone* → *ok*

Constraints → *Control Points* → *surface: sandstone* → *AtomsSet: sandstone* → *ok*

Constraints → *Control Nodes* → *From Control Points* → *surface: sandstone* → *tol: 1000*

Interpolation → *On entire surface* → *sandstone* → *ok*

The tolerance is the maximum distance of data points from the surface that shall be considered for defining control nodes. In this way, points far from the surface can be excluded. If the control nodes are set properly, the triangulation of the surface changes such that the control nodes become nodes of the surface.

5.3.2. Direct Triangulation

gOcad provides the facility to model surfaces from well data by Direct Triangulation.

The well markers are automatically set as control nodes and the curve objects describing the position of the sandstone horizon can subsequently be set as control points. Then the directly triangulated surface has to be interpolated a second time using DSI to adapt it to the control points derived from the curve objects.

Surface → *New* → *From Others* → *Well Markers* → *name: sandstone_well* → *marker name: sandstone* → *in cage* → *voxet: box*

Constraints → *Control Points* → *Set Control Points* → *surface: sandstone_well* → *AtomsSet: AllCurves*

Constraints → *Constraints On Borders* → *All Borders* → *surface: sandstone_well*

Interpolation → *On entire surface* → *surface: sandstone_well*

Split the surface, if you are not content with the interpolation.

5. 4. Modelling of the ground surface

The ground surface can, as well as the sandstone horizon, be modelled with a Direct Triangulation or a Discrete Smooth Interpolation. The second method yields a much smoother surface.

5.4.1. Direct Triangulation

Surface → *New* → *PointsSets name: direct_ground* → *AtomsSet: dem* → *ok*

5.4.2. Discrete Smothth Interpolation

Surface → *New* → *PointsSets Medium Plane* → *name: dsi_ground* → *AtomsSet: dem* → *Clip with voxet* → *Voxet: box* → *ok*

Surface → *Tools* → *Split all*

Constraints → *Control Points* → *Set Control Points* → *surface: dsi_ground* → *AtomsSet: dem* → *ok*

Constraints on border → *all borders* → *surface: dsi_ground* → *ok*

Interpolate → *On entire surface* → *surface: dsi_ground* → *ok*

5. 5. Calculate thicknesses

gOcad provides a tool to calculate the thickness of geological bodies that are bordered by two triangulated surfaces (fig. e). We want to calculate the thickness of the Quarternary above the sandstone horizon. First, we have to create a property, where the thickness values can be stored.

Surface → Property → Create → object: sandstone → property name: thickness → ok

Surface → Compute → Property → Compute Vertical Thickness → Surface: dsi_ground, thickness: thickness, bottom: sandstone, is bottom: on, is signed: on

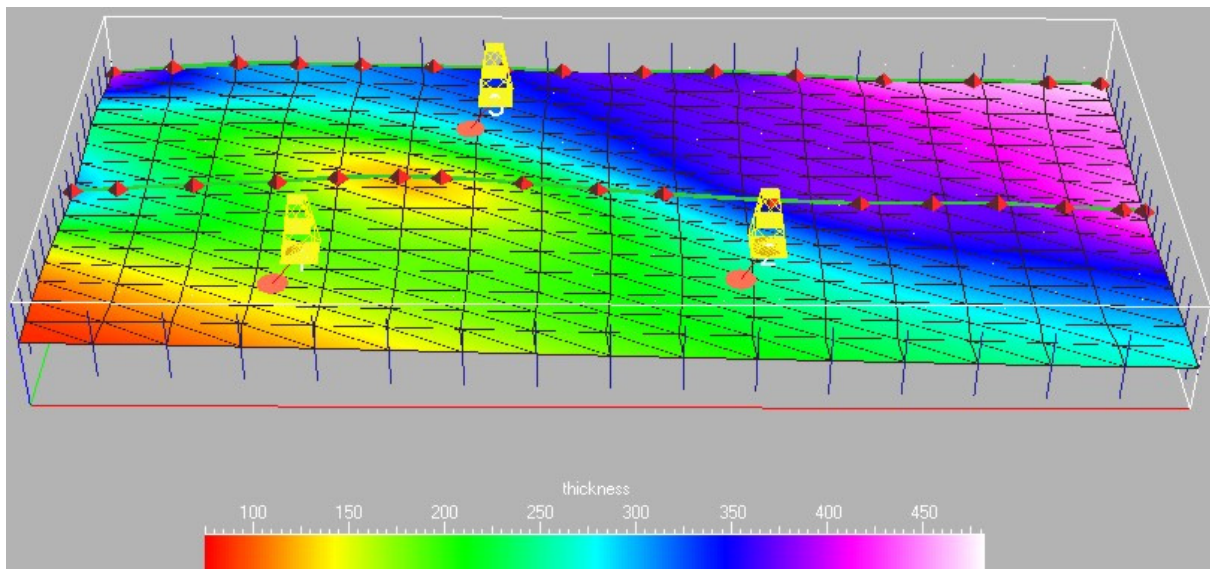


Fig. 5. e. Thickness of the hanging rock units of the sandstone horizon

5 6. Draping images on surfaces

Images like arial photographs or topographic maps can be imported to gOcad as 2D Voxets or georeferenced worldfiles and they can be draped on triangulated surfaces (fig. f). In this exercise, we want to drape an arial photograph on the ground surface of our model.

Load objects → photograph.vo → open

Object panel → surfaces → ground_surface → attributes → texture → draping: visible → voxel: photograph



Fig. 5. f. Draping of a photograph on a surface

OLIGOCENE LENSES

Start the gOcad project oligocene_start from the folder oligocene!

This project contains data from a geological map and from drilling holes describing the occurrence of two stratigraphic units, the Eocene in the underlying unit and the Oligocene on its top. Additionally, the ground surface is given in a digital elevation model. Aim of the exercise is to model the base of the Oligocene Series which is exposed in lenses.

The curves digitized from the map, have a Z value of zero, which is not realistic, since the outcrops were mapped on the ground surface. Therefore, we will first project the geological boundaries on the ground surface.

Project a curve on a surface

Curve → Tools → Project → On Surface → curve: oligoceneOutline and eoceneOutline → surface target: dem_0 → ok

If you have closer look at the result of the projection, you can see that the curves were not correctly projected on the DEM. The reason for this is a too low density of data points in comparison to the curvature of the ground. We can correct this with the tool "Densify".

Curve → Tools → Densify → Maximum Length: 10 m → XY unit → ok

Divide a curve objects in parts

Both stratigraphic units are given in a gOcad curve object consisting of two curves. In order to model the lenses separately, we have to split the curve objects into its parts. This can be done with

Curve → New → From curves → Per Curve Part → choose oligoceneOutline

Direct Triangulation

The DSI interpolation offers the opportunity to clip a new surface with a bounding box or with the minimum and maximum coordinates of the data set. Instead, DT allows creating a surface that is bordered by the data set and thus to display the shape of the lenses. For this reason, we will model the geology of the Oligocene base with Direct Triangulation.

Surface → New → From PointSet and Curve → Name: Oligocene1 → Curve: oligoceneOutline_part1 → AtomsSet Points: oligocenePoints → ok

Create a second surface for the second lense!

Now, let's have a critical look on our modelling result. Make the DEM visible! In large parts of the model, the Oligocene base is higher than the ground surface, which is not feasible. That's why we have to combine the Direct Triangulation with DSI, which allows to set constraints of various types.

Apply different boundary conditions to border parts

The borders of the interpolated surfaces consist of two parts:

- 1.) The outcrop of a geological border. This is a well constrained border, we know that the Oligocene base is cutting with the ground surface.
- 2.) The border of the map. In these locations, the Oligocene base is **not** coincident with the ground surface. This border is an artefact.

That's why we will apply different boundary conditions to these parts of the border. First we have to split the border in parts.

Surface → Tools → Border Extremities → Add → Click at the end of the map edge

Surface → Tools → Border Extremities → Move- → Click first at the extremity you want to move, then on the place where you want to move it to

Set control nodes along the geological border to force the surface through it and control points for the well markers and border constraints allowing a motion in Z direction for the map edge:

Constraints → Control Nodes → on one border → click at the border

Constraints → Control points → Set control points → oligocenePoints

Constraints → On border → Set on straight line → one border → click at the border

Interpolate many times until a the Oligocene base forms a depression

Now, we have obtained a much better modelling result, however, the “nose” is still not feasible. You can refine the mesh in this region for a better modelling result.

Refine the mesh locally

Surface → Tools → Split Rings → number of rings: 10 → pick a triangle → repeat

Interpolation → On Entire surface

This tool splits the triangle and the number of triangle rings around it.

Set thickness constraints to avoid crossing surfaces

Since this operation didn't bring the desired result, we can choose an alternative method to get the Oligocene base underground: We can set a thickness constraint, telling how deeply one surface has to be located beneath another.

Surface → Constraints → Set Range Thickness Constraint → surface: Oligocene1 → Controller: dem_0 → Constant Minimum thickness value: 3 → Tick off constant maximum thickness → Maximum thickness property: Z → Length unit: XY unit → ok

Cut a hole into a surface

Finally, we have to honor the Eocene outcrop in the Oligocene lense (fig. a):

Surface → New → From Curves → Border and Faults → name: oligocene_loch → curve border: oligoceneOutline_part1 → curve faults: oligoceneOutline_part2 → AtonsSet Points: oligocenePoints

This tool creates a new surface, but does not copy the boundary conditions of the original Oligocene surface. Set the curves of the Oligocene base (including that along the hole) as control nodes, the border constraints along the map edge, the well data as control points and the thickness constraints to obtain a geologically feasible model!

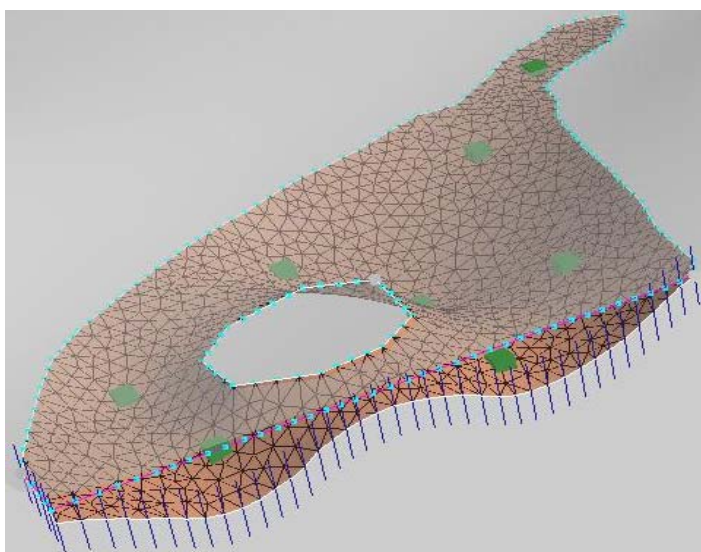


Fig. 6.a. Modelling result: Oligocene lense with hole

FAULTED HORIZON

7. 1. Introduction

This project shows how fault displaced stratigraphic horizons can be modelled in gOcad. A special kind of constraint, a vectorial link, is used to define the amount of displacement along a fault. The project contains the following data: a PointsSet containing the topography, a set of wells with the wellmarker horizon, a digitized interpretation horizonCurve and digitized interpretation faultCurve. The aim of this exercise is to model the fault and the displacement of the horizon along this fault.

7. 2. Create the horizon, fault and ground surfaces

Load the project faultmodel! Create a bounding box around the data set, scale it with 1.05! Model the Earth's surface, the fault and the horizon (fig. a)! Take care that all surfaces are clipped with the bounding box and that the horizon lies below the ground surface!

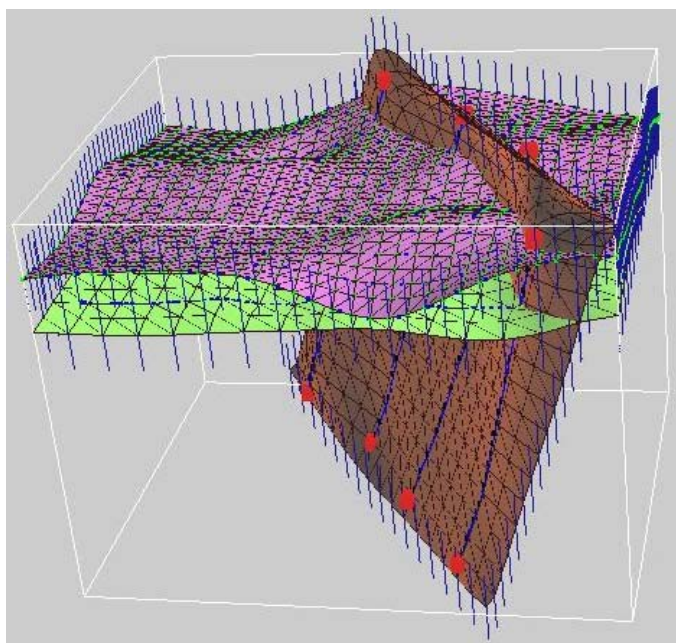


Fig. 7. a. Surfaces with constraints

Cut surfaces by surfaces

gOcad offers the opportunity to cut one set of surfaces A by another set of surfaces B. As a result, the surfaces A have new borders, where A and B intersect. New nodes are inserted at the intersection line and the triangulation is updated. Note: Never execute a cut operation twice on the same objects, precision & rounding errors may result in tiny slivers and thus inconsistencies! So if you use the command „Cut by Surfaces“, never click Apply + OK, but Apply + Cancel or Ok. We recommend to save your project before cut operations. The beautify tools are very important to attain good modelling results with cut surfaces. Beautify the triangles for equilaterality and simplify the borders!

In this project, we have to cut twice in order to clip the fault to the ground surface and to be enabled to model the fault displacement of the horizon.

Surface → Tools → Cut → by Surface → Surface: fault → surface by: dem → Beautify intersetion → ok
 Surface → Tools → Part → Delete selection → click on the part to delete
 Surface → Tools → Beautify → Simplify All Borders of Surfaces
 Surface → Tools → Beautify → Beautify Triangles for Equilaterality
 Surface → Tools → Cut → by Surface → Surface: horizon → surface by: fault → Beautify intersetion → ok
 Surface → Tools → Beautify → Simplify All Borders of Surfaces

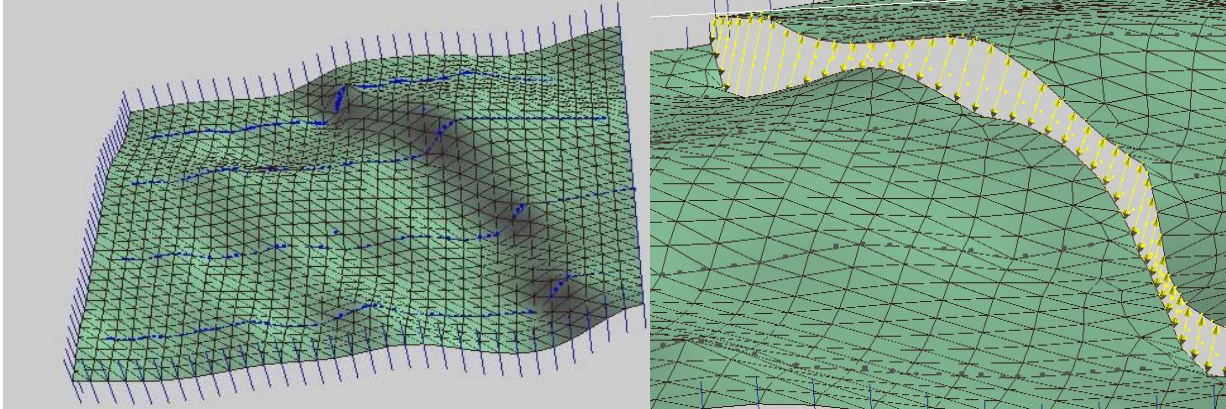


Fig. 7. b. The interpolated surface displays the fault displacement as a flexure (left); cut and fault displaced horizon, the yellow arrows indicate the vectorial links set between both surface parts.

7. 3. Modelling of the fault displacement

Now, we want to re-interpolate the horizon. During the cut operation the constraint *Surface* → Constraints → Constraints On Border → Set on Surface has been set automatically. It keeps the border of the horizon in contact with the fault.

Look at the modelling result! The horizon is only displaced in the area between the curve objects. The curves do not constrain the fault displacement, but rather a flexure of the horizon. The horizon was originally modelled as a continuous object with a steep gradient in its Z value near the fault. The data points in the vicinity of faults are erroneous seismic picks and should be deleted. Therefore, we create a PointsSet part from points nearer than 10 meters to the fault plane and delete it.

Click on the undo button in the Main Menu Bar

Constraints → Delete all of selected type → AtomsSet: horizon → Constraints type: Control Points → ok

Object Panel → PointsSet → New → From PointsSet, Curve or Surface → name: horizonpoints → AtomsSet: horizonCurve → ok

PointsSet → Segregate with surfaces → PointsSet: horizonpoints → surface: fault → maximum distance: 10 → Length unit: XY unit → ok

PointsSet → Tools → Part → Delete Selection → pick the part with the mouse

Constraints → Control Points → Set Control Points → surface: horizon → AtomsSet: horizonpoints

Now, we want to define a constant displacement vector between the two parts of the horizon (fig. b). This is achieved by setting a vectorial link with constant displacement. Measure the distance between the hanging and the foot wall with the „Get distances“ tool in the Camera Tool Bar. After setting all constraints, we reinterpolate the horizon surface.

Surface → Constraints → Constraints on Border → Set VecLink → ok → pick first the top and then the bottom border

Surface → Constraints → control Vectorial Link → Set displacement → Constant: 15 → ok → pick a vectorial link arrow

Surface → Interpolation → Geometry → On Entire Surface

GEO-OBJECTS WITH NON-UNIQUE Z VALUES

8. 1. Modelling folds

8.1.1. Direct Triangulation

For interpolation of surfaces, the points of an AtomsSet should be mappable on a plane. Otherwise, the surface topology may be inconsistent. The mapping direction is defined by a normal vector on the mapping plane. The default vector is the Z axis (001).

The PointsSet "foldpoints.vs" describes the position of a folded horizon. The Z values of this structure are ambiguous, this means that up to three fold limbs may have the same XY positions. During interpolation of a surface, the points with neighbouring X and Y values are connected. Modelling of the folds with gOcad's default settings will fail (fig. a). The problem can be eliminated by changing the mapping direction and choosing a mapping vector that is parallel to the c-axis of the fold.

Surface → New → PointSet → Advanced → use normal → X=1, Y=0, Z=1

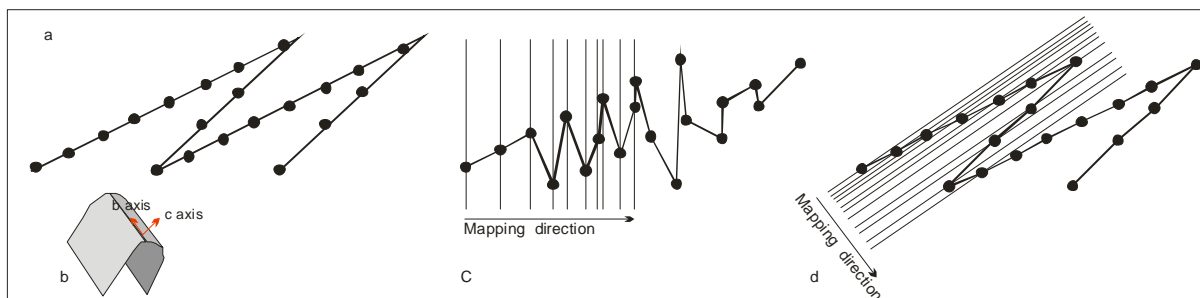


Fig. 8. a. How to model a fold; a) sequence how the data points should be connected, b) orientation of fold b- and c-axis, c) modelling result, if a vertical mapping vector is used, d) modelling result, if a mapping vector parallel to the c-axis is used.

8.1.1. Discrete Smooth Interpolation

Modelling the folds with DSI, we will generate similar problems as with DT, if we use a vertical shooting direction. The problem can be solved by choosing a shooting direction parallel to the c-axis of the fold.

Surface → New → PointSets Medium Plane → Tools → Split

Constraints → Control Points → Set Control Points → shooting direction → X=1, Y=0, Z=1

8. 2. Modelling diapirs

Diapirs are described by almost closed surfaces. This means, that it is almost impossible to find a mapping direction allowing unambiguous modelling of a surface. That's why the interaction with the user is needed to model diapirs and similar structures. The user has to pre-define the geometry of the surface by dragging one node of the surface into the interior of the diapir (fig. b).

Surface → New → PointSets Medium Plane → Tools → Fit → to PointsSet

Next, we have to move one node through the centre of the structure to the top of the structure. Split the medium plane as seldom as possible to get nodes along the edge of the diapir and one node at the centre. Pull the node in the centre by mouse drag and split the surface again. Then, set the PointsSet diapir as control points and optimize the shooting direction!

Surface → Tools → Split, space, space, space

Surface → Tools → Node → Drag with mouse

Surface → Tools → Split, space, space, space

Surface → Constraints → Control Points → Set Control Points → AtomSet:salt → Advanced → optimize shooting direction

Surface → Constraints → Constraints on Borders → Set on straight lines → All Borders

Surface → Interpolate → On entire surface → Conjugate → fitting factor:2

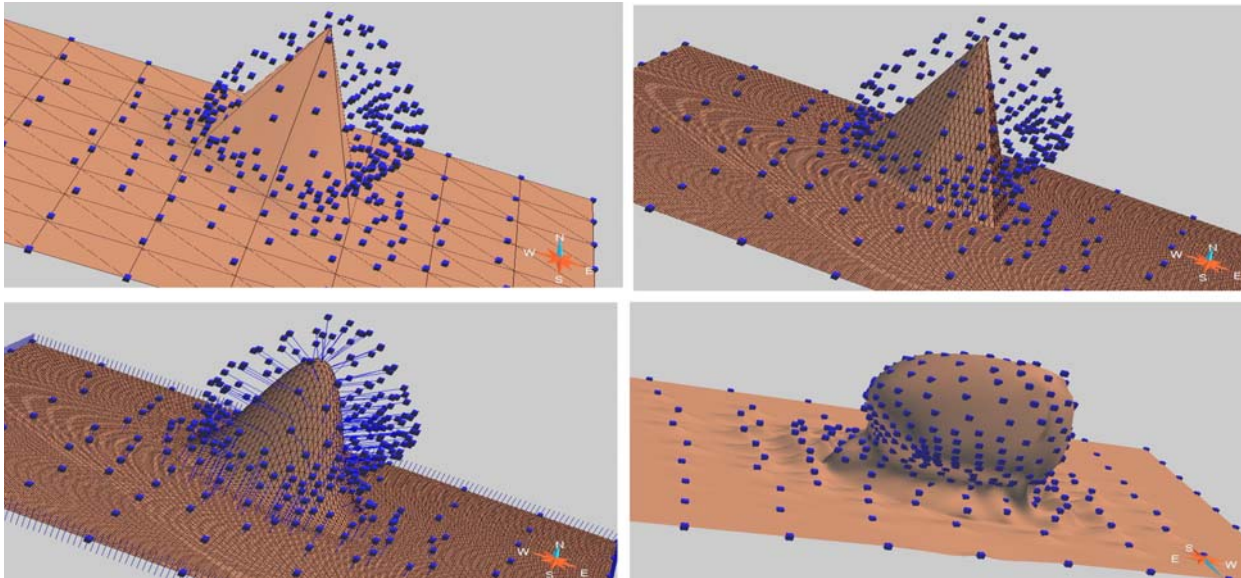


Fig. 8. b. Modelling of a diapir: drag a node of the medium plane from the centre of the diapir to the top, split the surface, set control points with optimized shooting direction, interpolate.

8. 3. Modelling of closed surfaces

Closed bodies like clay lenses can be modelled with a special tool (fig. c):

Surface → New → Build in forms → ellipsoid

Surface → Constraints → Control points → Set control points → optimize shooting direction

Surface → interpolate → on entire surface

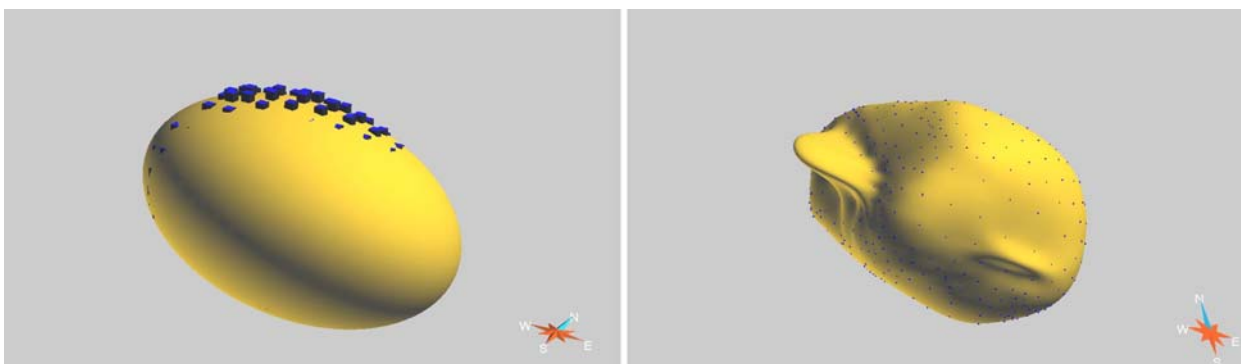


Fig. 8. c. Initial ellipsoid (left) and clay lens fitted to the data set (right).

Y-FAULT

This exercise emphasizes data import and data preparation, image georeferencing and data integration with DSI. We will create two conjugate normal faults (fig. a) "masterFault" and "northFault" from a set of wells, outcrop points and a 2D seismic section.

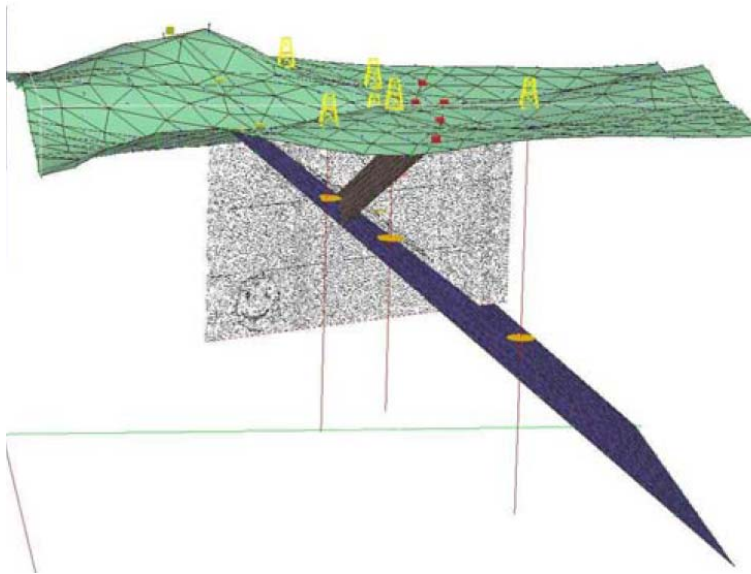


Fig. 9. a. Model of the y-fault

Data import

Create a new Project and load the digital elevation model and the positions of the faults in outcrops that are saved in the ASCII table files with XYZ points. Symbolize the data differently using the attribute Quick Edit Bar.

File → Import Objects → Horizon Interpretations → XYZ "dem.txt", "masterOutcrop.txt", northOutcrop.txt"

The well markers are imported with the columns containing MD, wellname and markername. Import the wells from wells.xls containing the well markers northFault and masterFault. Before the file can be imported it must be converted to an ASCII file (*.txt) and the delimiter must be changed to "point".

Additionally, we have to import a depth migrated vertical seismic section in tif-format:

File → Import Objects → Images → As 2d Voxet "seismic.tif"

Voxet → Tools → Resize with Points origin 10322.6, 11590, 392, point_u 10102, 11890,369, point_v 10322.6, 11590, 565, point_w 10323.4, 11590.6., 392

Model building

Digitize the reflectors masterCurve and northCurve as new curves on the seismic voxet. Create new PointsSets masterMarkers and northMarkers from the wellmarkers. To avoid mistakes, display only the objects you want to work with in the camera, and set graphical attributes consistently for the masterFault and northFault input data.

Build a new bounding box voxet, and scale any new surfaces by factor 1.05. This will avoid cut problems later. Model the surfaces using DSI. By the way, modelling can be more efficient if you work simultaneously for all three surfaces.

Save your project, before you cut the Surfaces !

SGrids

10. 1. Introduction

SGrids are regular 3D grids which can be deformed and cut to fit a structural surface model.

This sort of discretized volume is necessary, if we want to model the distribution of a property in a geological body. The grid axes are by default parallel to the XYZ axes of the model, but can be defined in a different way by the user.

In this exercise, we will model a part of a gold bearing quartz vein by creating a simple surface model from curve objects describing the vein boundaries, fitting an SGrid to the model and interpolating the gold grades in the SGrid using DSI. Resources with more than 0.3 ppm are mined profitably. A fault is truncating the reservoir separating a low-Au part in the north from a high-Au part in the south. We want to visualize the minable volume and to estimate the minable gold weight considering interpolation errors along the fault.

10. 2. Import of geochemical data as column-based file

Properties can be imported as PointsSets. The best way to do this, is to import a column-based file and to add the properties to the column list. After having imported the files, you can check whether you have correctly imported the properties by ticking them in the object panel and displaying them in the 3D camera.

File → Import Object → Horizon Interpretation → PointsSets → Column-based file → current file: au_grades.txt → delimited → next → next → Do you have X,Y,Z column? → yes,yes,yes → to name the object → use file name → next → Preload all → Start at line: 2 → End at line: 69 → Property list and parameters: → X → column 2 → Y → column 3 → Z → column 4 → add → Au → column 5 → advanced → domain → depth → ok

10. 3. Modelling of an SGrid between two geological boundaries

First, we model the two ore vein boundary surfaces using DSI

Voxel → New → From Objects Box → name: box → objects: au_grades bottomwall hangingwall → ok

Surface → New → From Pointssets Medium Plane → name: hanging → AtomsSet: hangingwall → Clip with voxel → apply → name: bottom → AtomsSet: bottomwall → Clip with voxel → ok

Tools → Split All → Surface: hanging bottom → ok

Constraints → Control Points → Set Control Points → Surface: hanging → AtomsSet: hangingwall → apply → Surface: bottom → AtomsSet: bottomwall → ok

Constraints → On Borders → All Borders → Surface: hanging bottom → ok

Interpolation → On entire surface → surface: hanging bottom → ok

Next, we create an SGrid filling the box. We have to define the size of the grid in relation to the box and the number of voxel in XYZ coordinate direction (Nu, Nv, Nw).

SGrid → New → From Objects Box → name: agrid → objects: box → Nu: 48 Nv: 48 Nw: 10 → property cell centred → ok

If you want to visualize the SGrid, you can use the object panel:

attribute manager → volume → tick full volume → ok

The grid has to be fitted to the modelled surfaces by

SGrid → *Tools* → *proportional between top and bottom* → *final horizon: hanging* → *reference horizon: bottom* → *ok*

Copy the grid to an object named southgrid (we will use this grid later)

Interpolate the Au grades on the SGrid with DSI:

SGrid → *Constraints* → *Set Property Control Points* → *Gridobject: agrid* → *AtomsSet: au_grades*

SGrid → *Interpolation* → *Initialize Property (Multi-Grid)* → *Gridobject: agrid* → *property: Au* → *region: everywhere* → *Advanced* → *tick: use all values*

You can visualize the property distribution (fig. a) by ticking

Object → *Panel* → *SGrid* → *agrid* → *properties* → *Au*

You can display cross sections perpendicular to the coordinate axes (fig.a).

Object Panel → *SGrid* → *properties* → *right click on porosity* → *attributes* → *property sections* → *choose axis* and *choose position of the cross section either by typing the number of the grid row or by pushing the bar right to the axis-selection bottom*

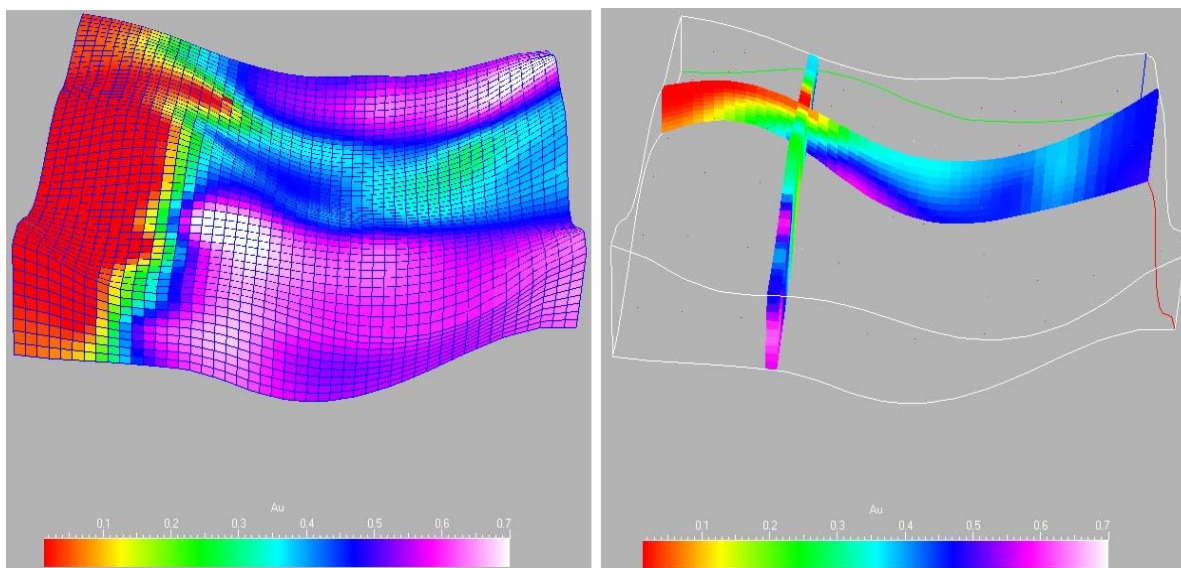


Fig. 10.a. SGrid with interpolated Au grades (left) and sections through the SGrid (right).

10. 4. Visualization of profitable and non-profitable parts of the ore body

Profitable parts of the deposit can either be visualized by iso-surfaces separating profitable and non-profitable volumes (fig. b) or by SGrid regions including all profitable or all non-profitable cells (fig. c).

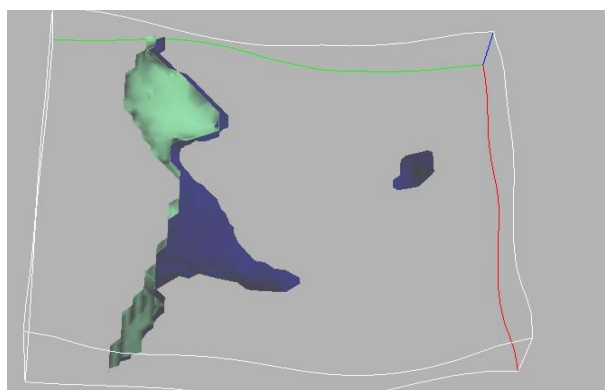


Fig. 10. b. Iso-surface separating mineable from non-mineable parts (right).

Iso-surfaces

Surface →New →From Grid →Grid Iso-Property →name: au03 →property: Au →isovalue. 0.3

Regions

Regions can be created in a similar manner as regions in PointsSets.

SGrid →Region →Create →object: agrid →region: mineable

SGid →Region →Initialize From Property Range →object: agrid →region: mineable →prpoerty: Au →Use Min Min: 0.3 →ok

The region has to be copied in order to calculate a complement region „non-mineable“:

Object Panel →SGrid →agrid →regions →mineable →right click →copy object: agrid →from: mineable →to: non-mineable →ok

Region →Edit →Complement →object: agrid →region: non-mineable

Query cell numbers and region volumes

You can check the number of voxels in the entire SGrid and in the regions and you can calculate the volume of the region.

Cells of the entire SGrid: *Object Panel →agrid →attributes →info*

Cells of a SGrid region: *agrid →regions →mineable →right click →Compute →Cell number in region*

Volume of a region: *agrid →regions →mineable →right click →Compute →Region volume*

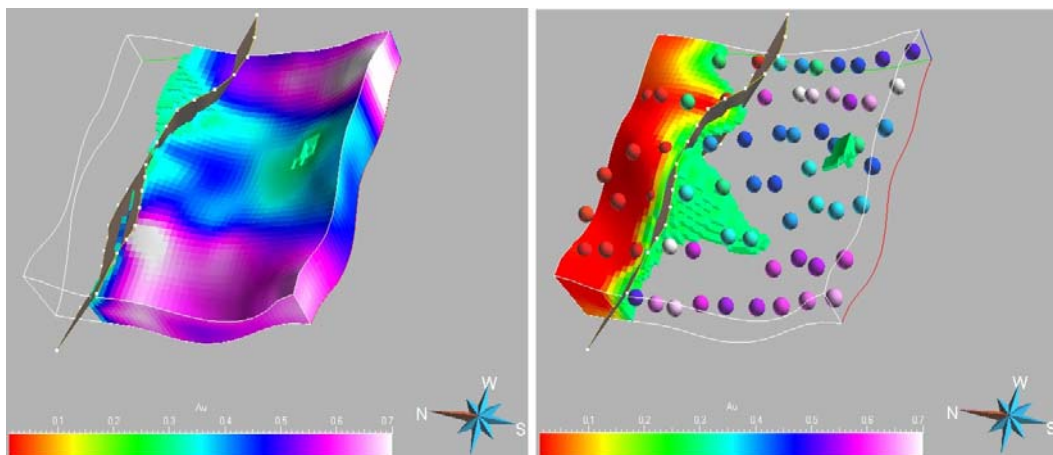


Fig. 10. c. Regions of the SGrid containing mineable (left) and non-mineable (right) Au grades. The gray surface is the fault truncating the Au deposit. Realize the misfit between the interpolation result and the primary data (spheres).

10. 5. Property interpolation considering the fault

The results of our data analysis show two parts with non-profitable Au grades: one big part comprising the north of the modelling area and a dead „hole“ in the centre of the southern modelling area. The dead rock in the north is separated from the deposit by a fault. However, the SGrid interpolation yields a low-Au volume to the south of the fault, which is not compatible to the primary data (fig. c). Therefore, it is expedient to exclude the area in the north of the fault from the SGrid interpolation in order to guarantee a feasible modelling result. Since gOcad allows to create regions by their position relative to gOcad objects, we want to model the fault, create a region in the south of the fault and to interpolate the Au grades only in this region.

Surface →New →From Curves Several Curves →name: fault →number of levels: 10 →ok →hold Strg-Key →tick the curves

SGrid →Region →Create →object: agrid →region: south

SGrid → *Region* → *Initialize* → *From surfaces* → *stratigraphic grid: agrid* → *region: south* → *surface: fault_0* → *including intersected cells* → *ok*

Visualize the region and compute the complement, if necessary!

Copy the *PointsSet* *au_grades* to a new *PointsSet* *south_grades* and delete all property data points north of the fault! Interpolate the properties in the south of the fault using the *AtomsSet* *south_grades* as property control points and interpolate in the region south of the fault! Use the *SGrid* object *southgrid* to avoid a hang-up of *gOcad*.

PointsSet → *Segregate with surfaces* → *PointsSet: south_grades* → *surface: fault* → *maximum distance: 5000* → *Length unit: XY unit* → *ok*

SGrid → *Constraints* → *Set Property Control Points* → *Gridobject: southgrid* → *AtomsSet: south_grades*

SGrid → *Interpolation* → *Initialize Property (Multi-Grid)* → *Gridobject: southgrid* → *property: Au* → *region: south* → *Advanced* → *tick: use all values*

The new interpolation result is better in the sense that the erroneous rim of low-Au cells near the fault does not appear (fig. d).

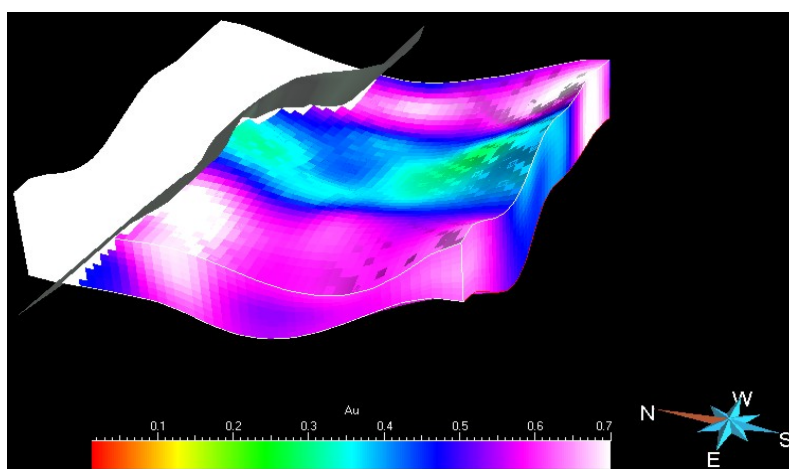


Fig. 10. d. Au grades after interpolation in the region south of the fault.

10. 6. Estimation of the mineable Au resources

Now, we want to estimate the mineable resources of the gold deposit in the *SGrid* south of the fault.

First, we will set all cells with non-profitable Au grades to Zero, such that these cells will not contribute to the estimated weight. Then we will create a new property „Au_weight“ and compute the Au weight of each *SGrid* voxel assuming a rock density of 2700 kg/m³. Finally, we will sum up everything.

SGrid → *Compute* → *Apply Script on object* → *object: southgrid* → *region: south* → *Main Script: if(Au<0.3){Au=0;}* → *ok*

Object panel → *properties* → *right click* → *create* → *property name: Au_weight*

SGrid → *Compute* → *Property* → *Compute Cell Volumes*

SGrid → *Compute* → *Apply Script on Object* → *object: southgrid* → *region: south* → *Main Script: Au_weight=2700*CELL_VOLUME*Au/1000000 ;* → *check* → *ok*

The result of the script yields the Au weight of each voxel. If we want to sum up the weights, we have to export the *Au_weight* data to EXCEL:

object panel → *southgrid* → *right click* → *export to* → *EXCEL* → *choose Au_weight*

MODEL3D AND TETRAHEDRAL MESH

11. 1. Introduction

The faulted blocks can be visualized in a 3D model of surfaces confining a rock volume with the object type „model3d“. The model3d shares each face between neighboring 3D volume regions. A region in this context is a closed space bounded by surfaces. When creating a model3d, we have to guarantee that we work with intersecting input surfaces. Therefore, it is necessary that the geological surfaces are larger than the bounding box. For creating a model3d, we will use the y-fault project which offers four open surfaces (fig. a). First, we have to create a bounding box smaller than the surfaces (scaling factor 0.9) and to create boundary surfaces from the box. These surfaces have to be „glued“ to one 3D body. Then, we can create the 3D model from all surfaces. The modelling algorithm honors geological relationships between surfaces like the priority of the Earth's surface or of an intrusive body cutting through horizons. Therefore, we have to assign the geological meaning of each surface included in the 3D model by setting geological information. This can be done by defining geologic features. A feature is a classification that is used to identify characteristics of geological objects like geological events, the stratigraphy or contact relations. After defining the geological feature, we can create the model3d.

11. 2. Create a model3d

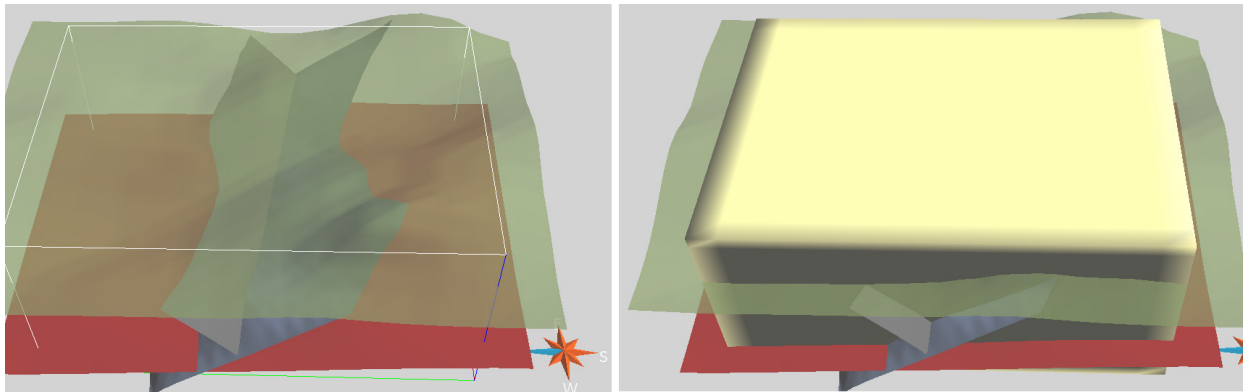


Fig. 11.a. Triangulated surfaces and object box for the creation of a model3d; surfaces and boxsurface

Voxet →New →From Objects Box →box Objects →choose: all surfaces →scale U;V;W dimension by 0.9

Surface →New from Voxet Cage →name: boxsurface →voxet: modelbox

Surface →Tools →Part →Merge all →boxsurface

Top of the object panel →question mark icon →assign data to geologic feature →choose horizon →assign horizon feature →choose all faults →assign fault feature →choose groundsurface →assign topography feature →choose boxsurface →assign model boundary feature

Model3d → From Surfaces → name: blocks → surfaces: choose all surfaces

Visualize the 3D model in the object panel (fig. b):

Model3d →blocks →regions

Delete the free extremities with

Model3d →more →remove free extremities

You can combine several blocks as layers. Now we want to create two layers above and below the stratigraphic horizon:

Model3d →Layers →Create one model: blocks →layer name: bottom →region: Region_1 →ok

Object panel →model3d →blocks →regions right click on Region_3 →add to layer →Layer: bottom →ok

Create an exploded view using the Quick Attribute Tool Bar tick on the „exploded view“ icon choose a number between 0 and 1 (fig. b).

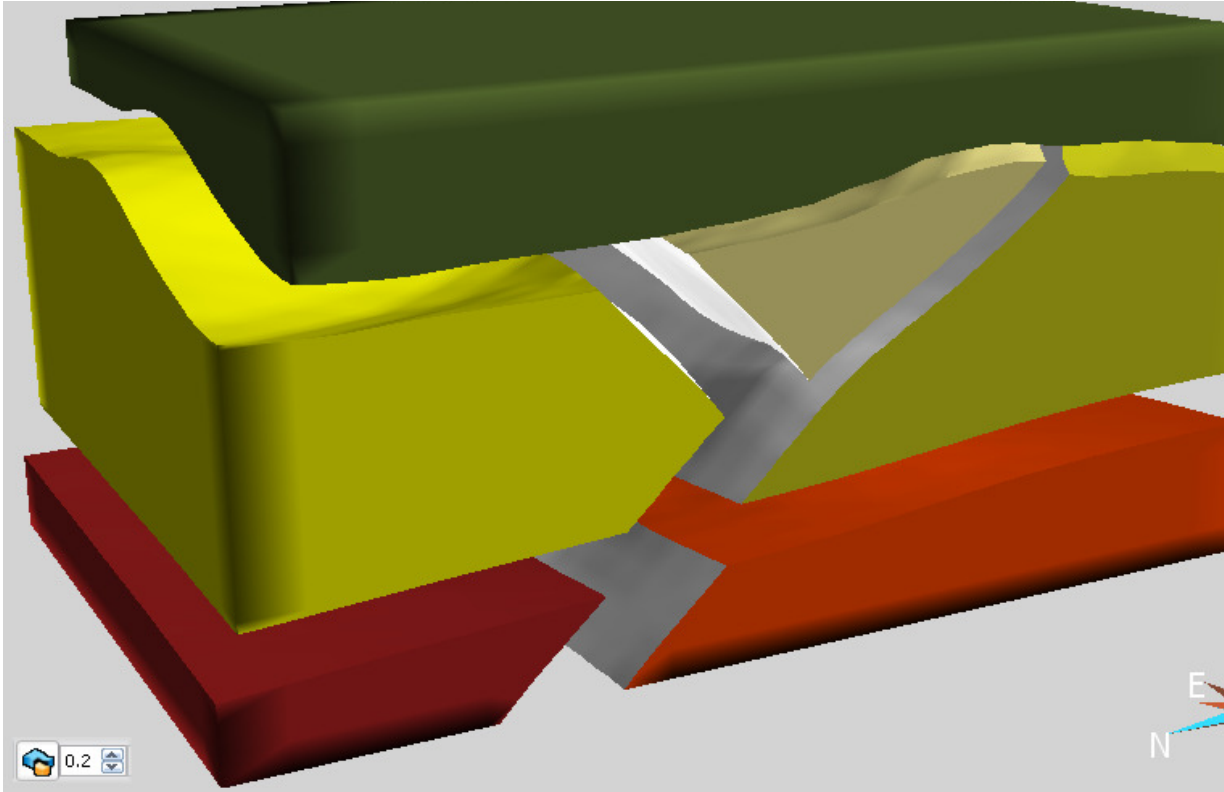


Fig. 11.b. Model3d in exploded view and the „exploded view“ icon of the Quick Attribute Toolbar

11. 3. How to refine the Y-fault surfaces prior to creating a model3d

Create the surfs in correct horizontal extension allowing proper cutting: ground surface > masterfault > northfault and horizon

Constraints on border for the faults: Choose advanced and set 0 1 0

Interpolate Conjugate, fitting factor=2

Split again all surfaces

Cut and simplify all borders

create amodelbox with 0.8, 0.8 1 scaling factors

move the boxsurfaces such, that the ground surface does not intersect the top boxsurface and such that the master fault intersects the bottom boxsurface ($Z=Z+10$;))

11. 4. Create a tetrahedral mesh (Solid object) and consistently triangulated surfaces

If geometrical models generated in gocat shall be used in other software, f.e. simulation software, a 3D mesh has to be generated and the triangles of neighbouring surfaces have to be triangulated in a consistent and water-tight way, this means that a node at the border of one surface needs a corresponding node at the neighbouring surfaces. No gaps and holes must exist. Normal gocat triangulation and cutting tools usually cause problems like shown below: the cut horizon crosses the fault, acute triangles occur along the cut boundary, the nodes are not corresponding on the fault surface and the horizon surface (fig. c). This can be fixed with the Tessellation Workflow, when the module „Finite element Constructor“ was selected.

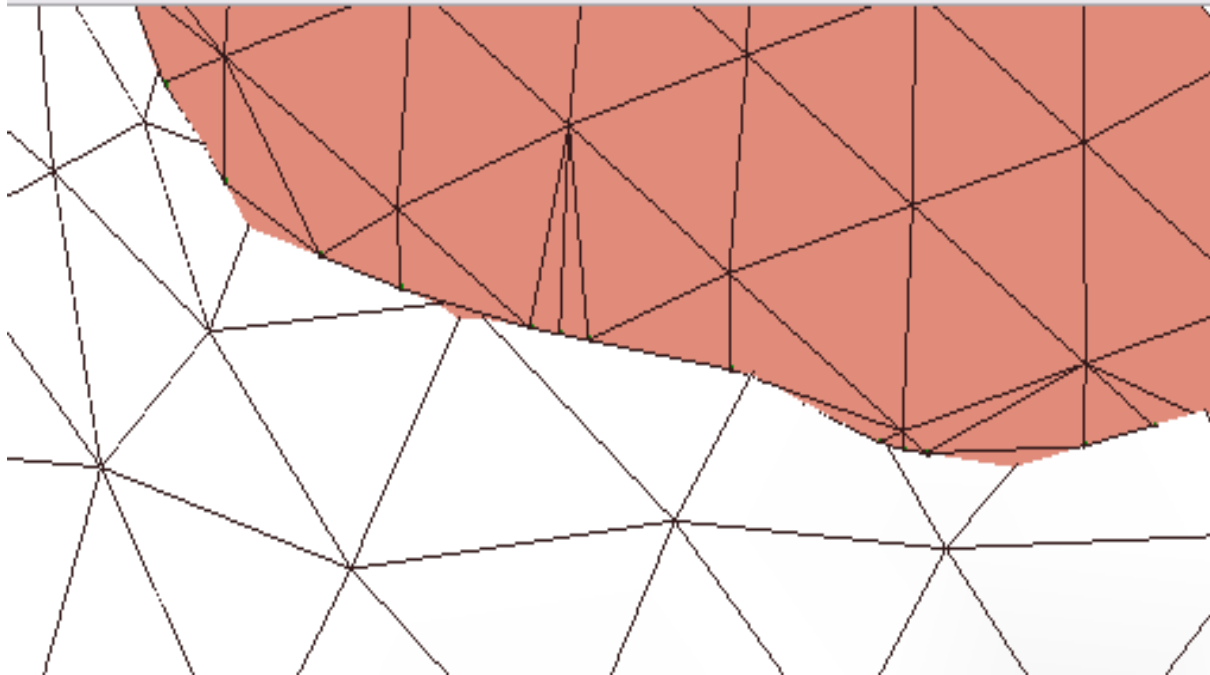


Fig. 11.c. Problems with gocat triangulation and cutting tools. After cutting the horizon surface along a fault acute triangles with very low angles occur and the triangles on the horizon boundary do not have corresponding nodes at the fault boundary (upper image). The cut horizon may even cross the fault (lower image).

The Tessellation workflow creates a gocat object named solid. A solid is an object consisting of a set of points connected by tetrahedra. It is used for ray-tracing and velocity-modelling, volume-modelling and restoration. Solids can be generated from a model3d with the Tessellation Workflow. However, the workflow does only work, when the model boundary box is created by six separate overlapping surfaces each with an own name (fig. d). Sometimes the box-surfaces have to be splitted and triangles have to be beautified for equilaterality in order to receive a successful meshing.

The model3d can be created as empty box, then all geological objects that are needed can be added and free extremities can be removed. Take care that the surfaces are bigger than the box, such that a completely closed volume is created (fig. e). The model3d is good enough for creating a tetrahedral mesh, when no red lines are left after removing the extremities. Red lines show inconsistencies in the model3d.

Surface →Model3d →Edit →Add Surface

Surface →Model3d →Edit →More →Remove Free Extremities →Region: Universe

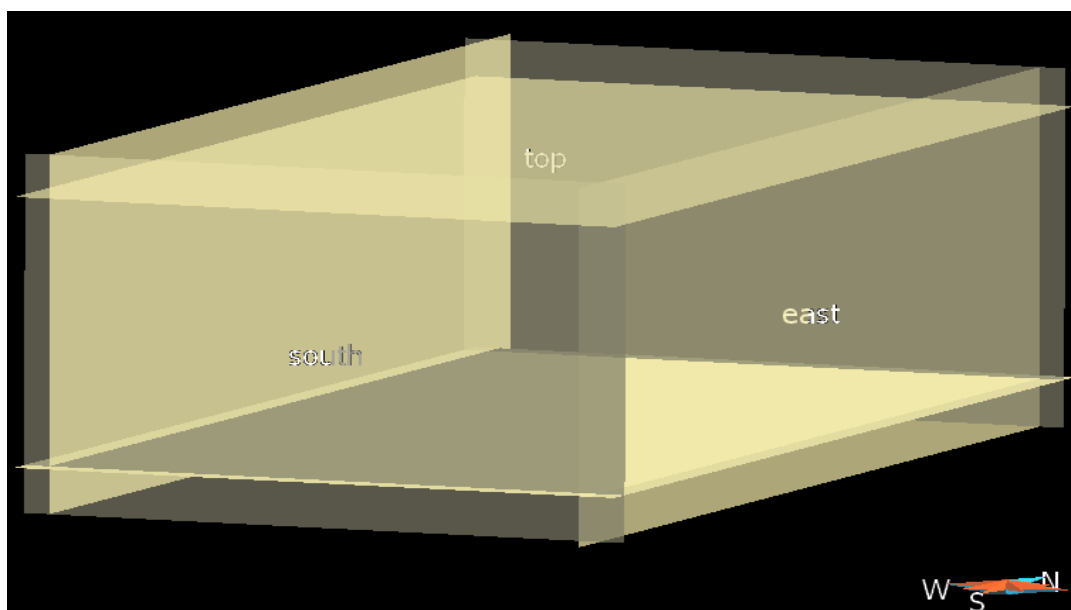


Fig. 11.d. For creating a tetrahedral mesh, a box of six crossing surfaces is needed.

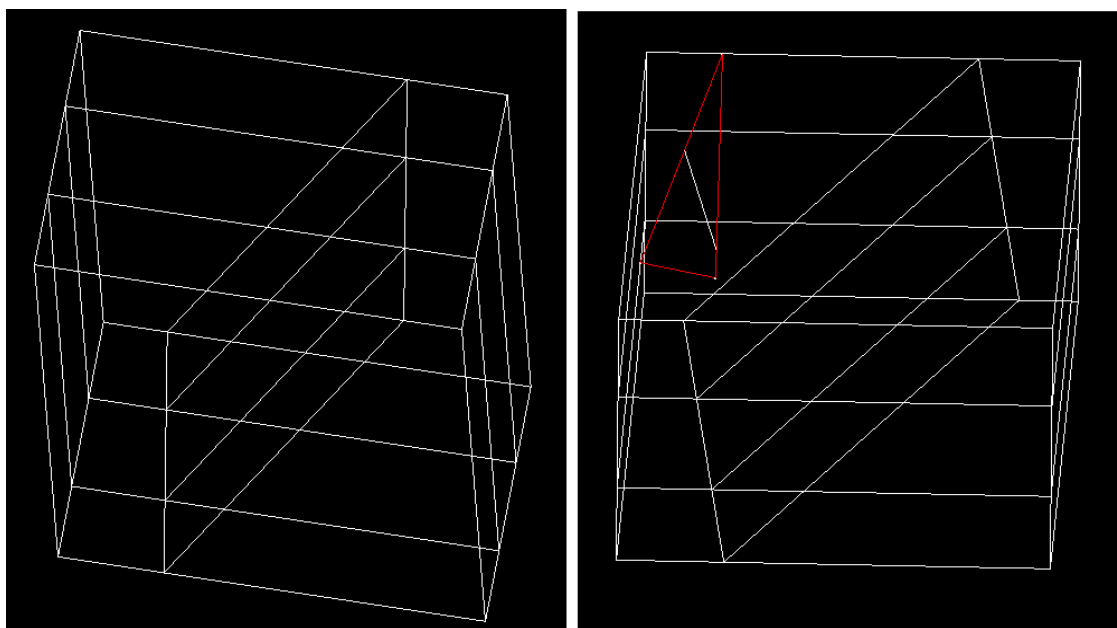


Fig. 11.e. After creating a model3d from the boxesurfaces, you can add fault and horizon surfaces to the model. If these surfaces surround a closed volume, the model contour is displayed in white color (left). If one set of surfaces does not completely surround a volume, red lines indicate the problem (right).

For creating the tetrahedral mesh, start the Tessellation workflow:

Task pane → Workflows → Tessellation Right click → New tessellation_study → ok

Object selection → An existing model → 3d model: model → Next

Regular tetrahedric mesh → next

name: tetramodel

If you change the default size, the number of tetrahedrons to be created is shown in the window below. If this number is too high, your computer will hang up, if this number is too low, narrow parts of the model cannot be meshed properly. Choose a size that results in a tetrahedron number $2000 < n < 20000$. Create the tetrahedric mesh and move to the Object

panel. The tetrahedron model can be displayed in the Object panel as Solid object with regions (fig.f).

Additionally, gocad creates new surfaces with the name of the solid model tetramodel1 to tetramodel9. These surfaces are consistently meshed, which means that every triangle node at one surface has a corresponding node at the neighbouring surface (fig.g). Such, the triangulated surfaces can be used as input for simulation softwares.

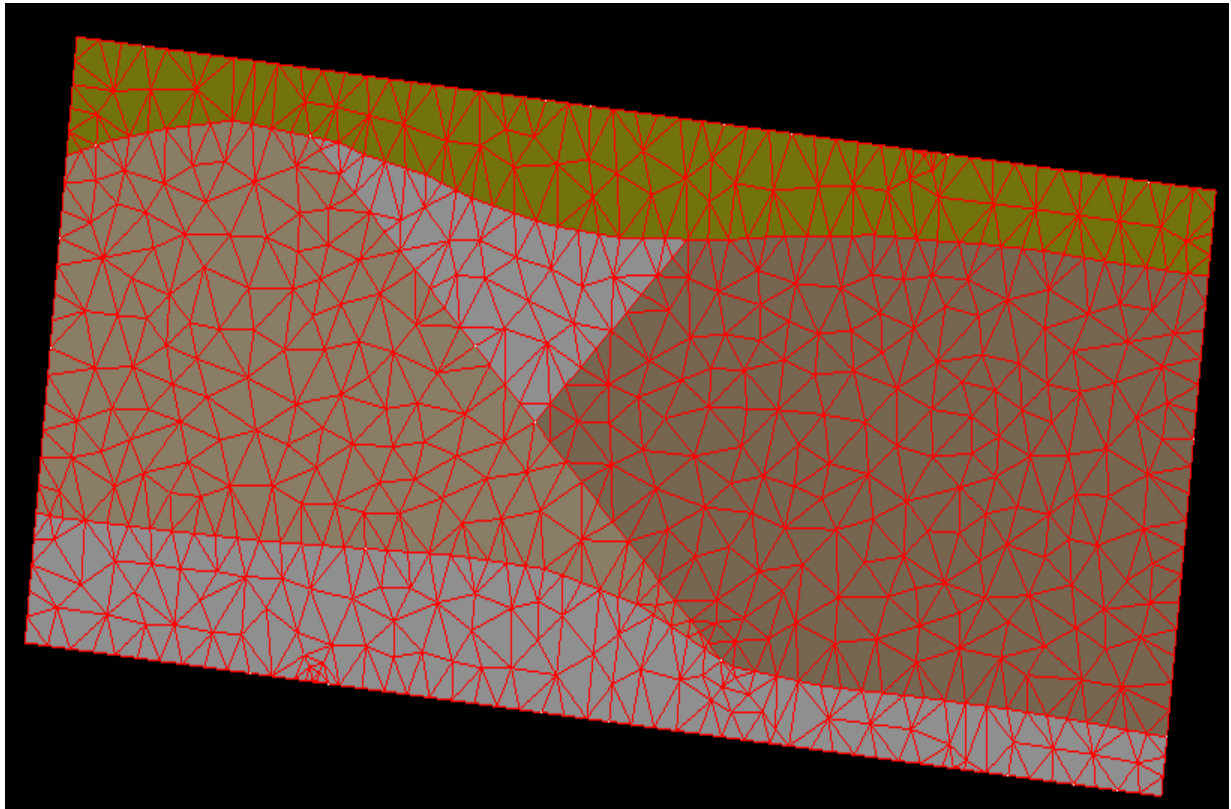
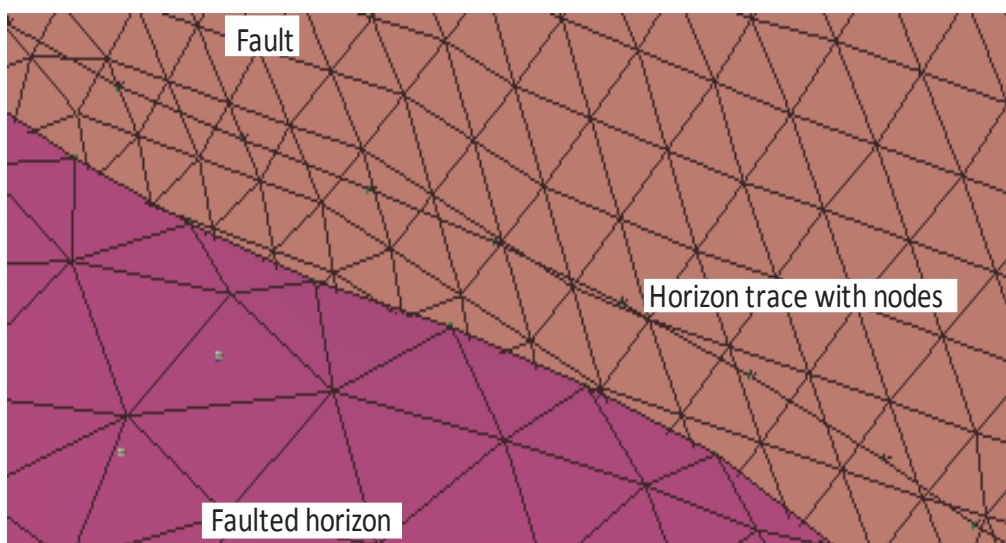


Fig. 11.f. Solid object created from the model3d of the y-fault. All boundary surfaces contain nodes and the regions to their sides are meshed in a consistent manner, each triangle in one region has a corresponding node in the neighbouring region.



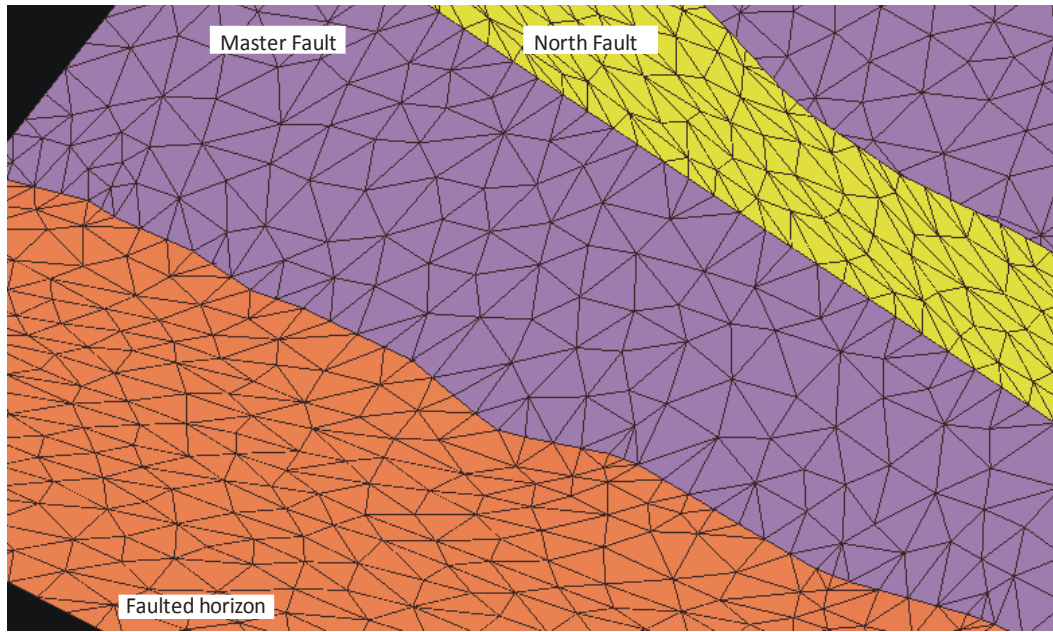


Fig. 11.g. Comparison of the T surface mesh after the initial DSI triangulation and cutting procedure and after remeshing with the Tessellation workflow.

WORKFLOWS

12. 1. Introduction

Workflows are semi-automatic routines allowing to build consistent models with pre-defined steps. Provided that your data is consistent with the demands of the workflow, this way of model building is comfortable and time-saving. The workflows are administrated in the *task panel* → *workflow* menus. While you work with the workflow, you can always switch to the object panel and see the surfaces you have created. You can jump forward and backward in the workflow, possible options are highlighted in the workflow panel. When you model fault-displaced horizons you have to take care that the horizon data points are far enough from the faults, if you want to get good modelling results with the workflow.

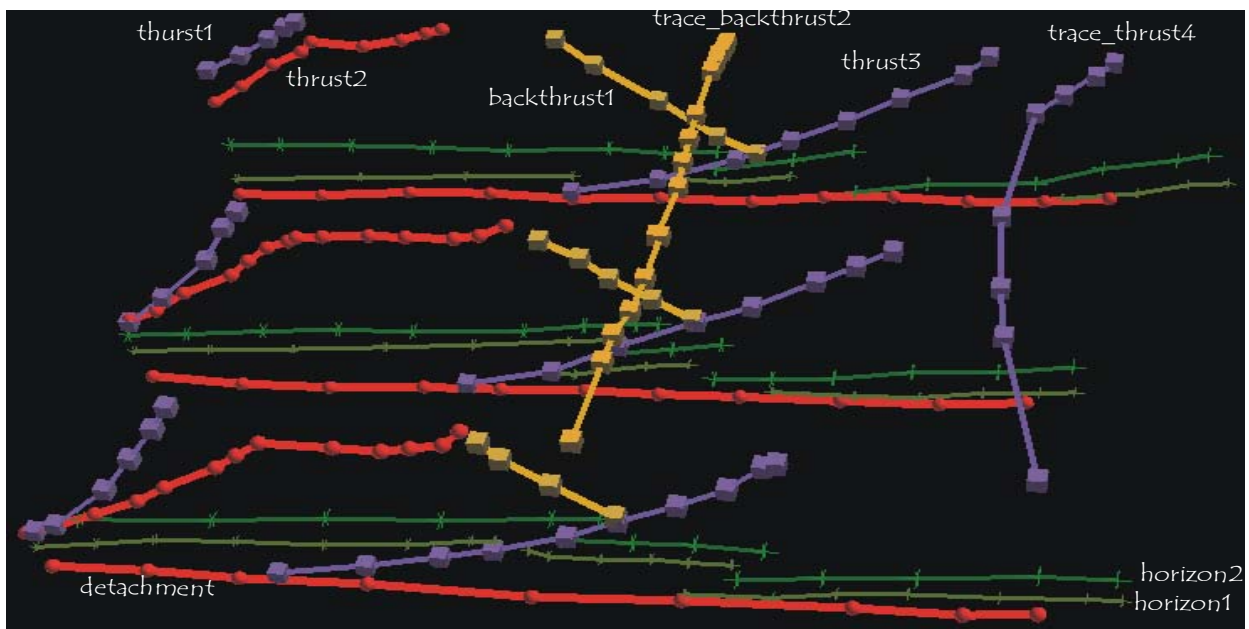


Fig. 12.a. The data set describing the thrustbelt.

The data set used in this exercise comprises fault and horizon data from a thrustbelt (fig. a): Curve objects displaying faults that were digitized from three parallel seismic sections:

- detachment
- thrust1
- thrust2
- thrust3
- backthrust1

Curve objects from surface traces of two faults whose orientation is given by dip direction and dip angle:

- trace_backthrust2 - 91°/45°
- trace_thrust4 - 269°/35°

Curve objects determining the position of two stratigraphic horizons:

- horizon1
- horizon2

The thrustbelt is E-vergent and consists mostly of W-dipping faults (red and violet). Two backthrusts dip to the E (orange). The geometry of the thrusts in the modelling area is complex. The thrust2 and the detachment are primary faults (red), at least in the modelling area. The other faults are secondary and originate in one of the primary or secondary faults (violet). Two faults change their contact-relations. Thrust1 is originating from thrust2 in the south of the modelling area, but is releasing from it in the north. The backthrust1 also has a contact to thrust2 in the south, but a contact to the ground surface in the north. The horizons (green) are displaced along thrust3 and thrust4.

12. 2. Modelling of complex fault patterns

12.2.1. Data management

Start the Workflow

Start a new modelling workflow with:

Workflows → right click on *Structural Modelling* → New → *SM_study*

If you go back to *Workflows* → right click on *Structural Modelling*, you can open or save an existing workflow.

Data managing

First you have to determine, which gOcad objects shall be included in the workflow modelling. One gOcad object is used for the interpolation of one surface. Or the other way around: You have to organize all data constraining one surface in one PointsSet or Curve object.

Do you want to add data in: → tick *depth*

Do you want to add/remove fault data? Yes

Define the data type of each gOcad object used for fault modelling by ticking on the object and choosing the data type in the box at the bottom of the list. The data type „*fault center lines*“ is used for modelling surfaces by orientation data. Tick „*trace_backthrust2*“ and „*trace_thrust4*“ and assign „*Fault center lines*“. If the position of a fault surface is constrained directly by curve objects, use the data type „*Fault Sticks*“. Tick „*thrust1*“, „*thrust2*“, „*thrust3*“, „*backthrust1*“ and „*detachment*“ and assign the data type „*Fault Sticks*“. Move the objects to the „*selected faults*“ window by marking them and ticking the arrow in the middle of the dialog box.

Now, you can see the „*advanced*“ button next to the „*Fault center lines*“. Tick on it and define the dip direction and angle. If you tick on the dip direction symbol in the azimuth column, you can change the dip direction. The columns „*lower extend*“ and „*upper extend*“ are used to specify an interval of absolute Z values in which the fault surface is created. Add -3000 in the column „*lower extend*“ and 0.0 in the column „*upper extend*“. Close the window.

→ok

Do you want to add/remove horizon data? → Yes

Define the horizon objects as data type „*interpretation line*“ and add them to the „*selected horizons*“ window → ok

Define volume of interest → Yes

Do you plan to use a stratigraphic column? → Yes (necessary to remove crossing horizons)

Select stratigraphic column → create → New column → ok → Build column → remove faults and ground surfaces with the red button in the centre column → Change the horizon sequence → move with the upward/downward arrows → ok → Next

Volume of interest

Default volume of interest → Choose all objects that shall be surrounded by the bounding box → Edit area of interest → set top view → move edge with mouse (define a box that is about 5 km wider and longer then the extension of the

data set in order to better the workflow builds contacts of faults with the box, define the edges halfwise parallel to the seismic lines) → Edit vertical extension → set north view → Edit top → define the absolute Z value of the upper edge (0) Edit base → define the Z value of the lower edge (-12000) → Create/re-create volume of interest → Next

Modelling of complex fault pattern is performed in two steps, the fault surface modelling and the fault contact modelling that includes the definition of contact lines between branch faults and between faults and the bounding box. First, gOcad creates a convex hull around the fault sticks and creates a surface whose border is identical with the convex hull. The surfaces created in this manner do not yet form closed polyhedrons and their extend is limited to the extend of the data set (fig. b). In the second step of fault contact modelling, the intersections of the fault surfaces with other faults or the volume of interest can be specified by the user.

From the viewpoint of fault contact modelling the model contains two types faults: The thrust2 and the detachment are primary faults, at least in the modelling area. The borders of thrust 2 and detachment are exclusively intersection lines with the bounding box. All the other structures are secondary faults originating in a fault, so that the triangulated surfaces must not be extended to the bounding box. Some of the borders of these faults are intersection lines with other faults, some of their borders are intersection lines with the bounding box. We will model both types of faults separately.

12.2.2. Modelling faults that are completely bordered by the bounding box (detachment and thrust2)

Fault modelling

Build fault surfaces → Faults: detachment thrust2 → tick the black little arrow in front of „build fault surfaces“ → specify a default size → 500 → ok → build fault surfaces

The default size of the mesh is usually too big to guarantee a proper interpolation, therefore the mesh should be specified by hand. The mesh can be visualized in the Attribute Tool Bar at the bottom of the workflow window.

Next → Next

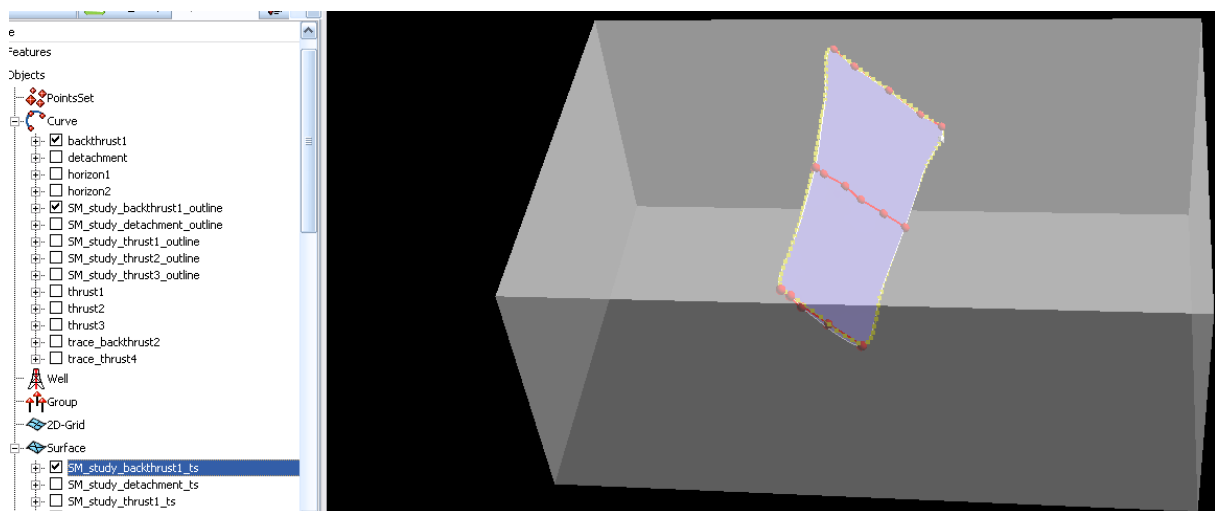


Fig. 12. b. Modelling of a fault surface in the workflow from fault sticks surrounded by an outline curve.

Fault contact modelling

Propose fault contacts → Choose detachment and thrust2 as Main fault candidates and as Branch fault candidates → tick the black little arrow in front of „propose contacts“ → all boundaries → define fault contacts using distance → 5000m → ok

Propose contacts → undo

define fault contacts using distance → 10000m → ok → Propose contacts

If gOcad finds a face of the volume of interest within the given distance from a surface border, constraints are generated that extrapolate the fault surface to the bounding box. If the distance defined by the user is too small, only those parts of the surface boundaries that are near enough to face of the volume of interest are extrapolated (fig. c). Check, whether the complete surface is extrapolated to the bounding box, before you proceed.

Next → Honor fault contacts → Faults: detachment thrust2 → Honor fault contacts

Save project

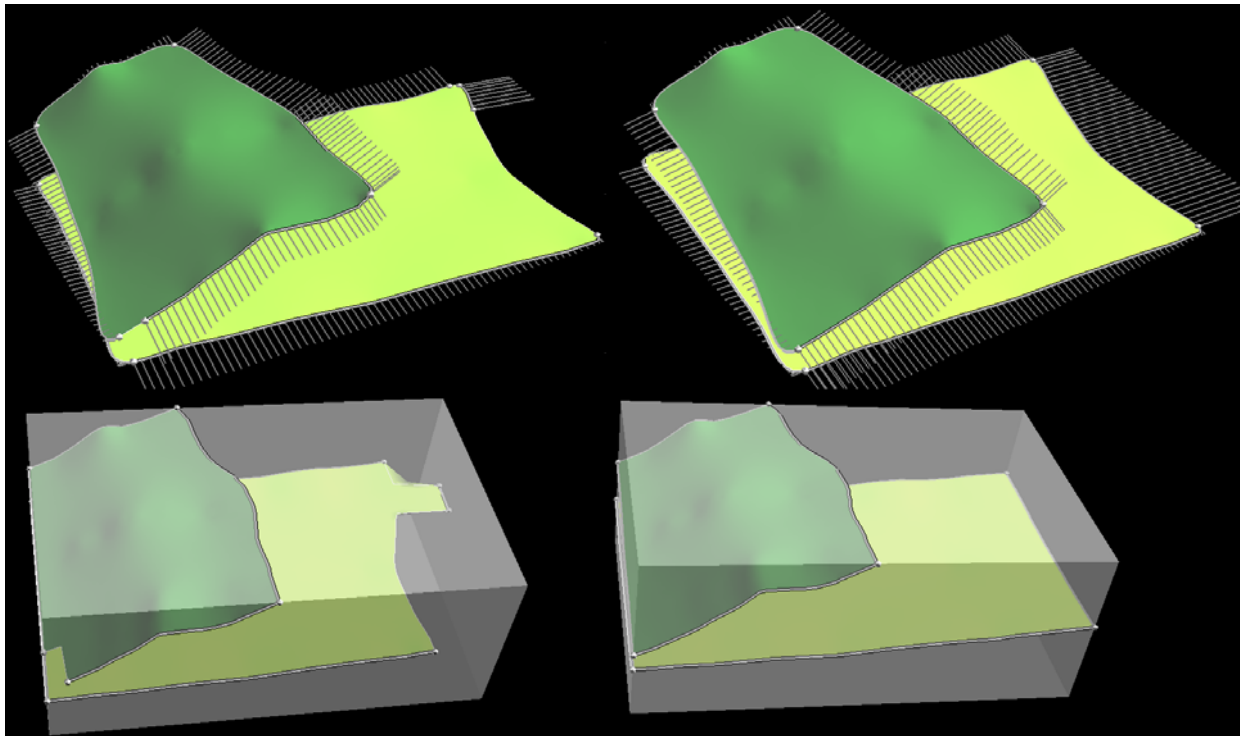


Fig. 12. c. Setting border constraints using distances; left side: the distance defined as border constraint is too short to connect the whole surface boundary to the volume of interest. The interpolated surfaces are not completely intersecting with the volume of interest and have sharp edges; right side: the distance defined as constraint on border is big enough to connect the whole surface boundary with the volume of interest. Interpolation results in a four-sided surfaces whose boundaries are intersection lines of the surface and the volume of interest.

12.2.3. Modelling of branch faults rooting in primary faults

Go back to fault modelling → Faults: backthrust1 thrust1 thrust3 trace_backthrust2 trace_thrust4 → Build fault surfaces

Fault contact modelling → choose all faults as main and branch fault candidates → propose fault contacts → black arrow → define fault contacts by hand → Propose contacts → Next → Next

Edit contacts

Main Fault: thrust3 → Show selected main fault → Selection method → any faults → Secondary fault: backthrust1 → Show selected secondary fault

Add one border extremity to each corner of backthrust1 by clicking on „Add border extremity“ and on the corner of the surface

Edit contact → Click on „add contact“ and on the border that shall be extrapolated to thrust3

proceed with trace_backthrust2

Add border extremities

Add a contact to thrust3

connect thrust3 and detachment, trace_thrust4 and detachment

If you clicked on the wrong border, you can delete the contact with „remove contact“.

Create contacts to the bounding box faces in a similar manner:

Main Fault: voi_NE179_ribbon →Secondary Fault: backthrust1 →Check whether the contact line is already limited by border extremities →If not add border extremity →Add contact →Secondary fault: trace_backthrust2 →Add contact

Add all fault-box contacts **except** those between backthrust1 - top and thrust1 - sidewall (these two contacts are releasing to a fault and will be modelled separately in our last fault modelling step). Interpolate with:

Honor fault contacts →Faults: backthrust1 thrust1 thrust3 trace_backthrust2 trace_thrust4 →honor fault contacts →Save project

12.2.4. Modelling fault borders with changing contact (backthrust1 - top and thrust1 - sidewall)

Go back to Edit Fault contacts →Main Fault: thrust2 →Secondary fault: backthrust1 →Add border extremity where backthrust1 is releasing from thrust2 →Add contact between backthrust1 and thrust2 →Main Fault: voi_top →Secondary Fault: backthrust1 →Add contact

Honor fault contacts →Faults: backthrust1 →Honor fault contacts

Go back to Edit Fault contacts → Main Fault: thrust2 →Secondary fault: thrust1 →Add border extremity where thrust1 is releasing from thrust2 →Add contact between thrust1 and thrust2 →Main Fault: voi_NW_93_ribbon →Secondary Fault: thrust1 →Add contact

Honor fault contacts →Faults: thrust1 →Honor fault contacts

click on the undo button and move the border extremity if the modelling result is not satisfactory

Reinterpolate the surface thrust1 with „Honor fault contacts“

Save project

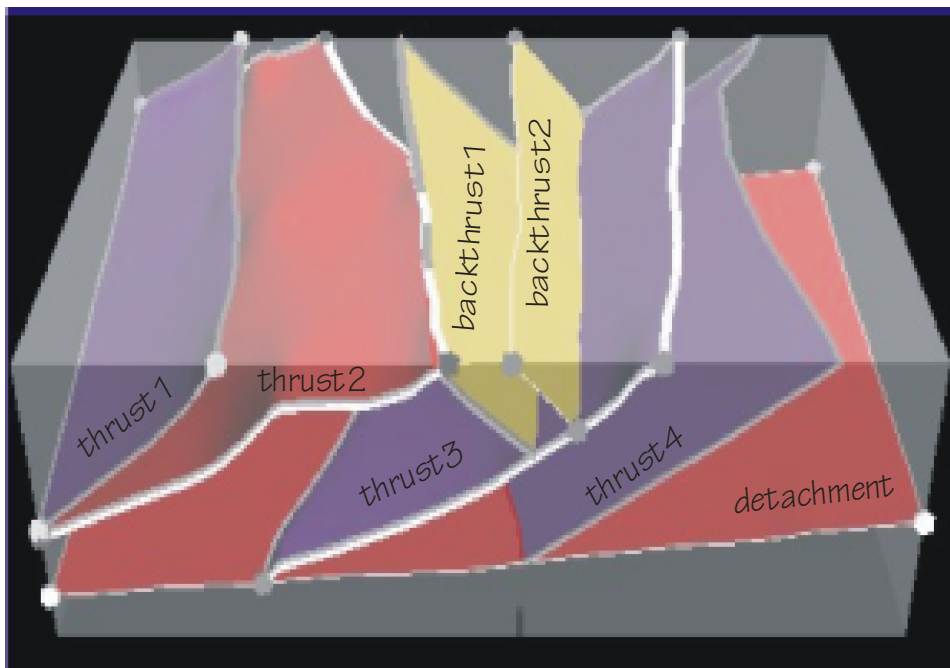


Fig. 12. d. The fault model.

12. 3. 3D Model

The faulted blocks can be visualized in a 3D block model. The 3D model shares each face between neighboring 3D volume regions. We have to guarantee that we work with intersecting input surfaces. Therefore, it is necessary to create the geological surfaces larger than the bounding box. Our project offers seven open surfaces providing the base of the 3D model (fig. d). First, we have to create a bounding box smaller than the surfaces (scaling factor 0.9) and to create boundary surfaces from the box. These surfaces have to be „glued“ to one 3D body. Then, we can create the 3D model from all surfaces. Since the modelling algorithm honors geological relationships between surfaces like the priority of the Earth's surface or of an intrusive body cutting through horizons. Therefore, we have to assign the geological meaning of each surface included in the 3D model by setting geological information. Then, we can create the 3D model

Voxel → *New* → *From Objects Box* → *box Objects* → *choose: all surfaces* → *scale U;V;W dimension by 0.9*

Surface → *New from Voxel Cage* → *name: boxsurface* → *voxet: box*

Surface → *Tools* → *Part* → *Merge all* → *boxsurface*

Surface → *Tools* → *Set geologic information* → *boxsurface* → *Model boundary* → *apply* → *faults* → *fault* → *ok*

Model3d → *From Surfaces* → *name: faultblocks* → *surfaces: choose all fault surfaces and the boxsurface, exclude the voi-surface*

Surface → *Model3d* → *Layers* → *Create Defaults*

Visualize the 3D model in the object panel:

3dmodel → *faultblocks* → *regions*

Create an exploded view using the Quick Attribute Tool Bar tick on the „exploded view“ icon choose a number between 0 and 1 (fig. e).

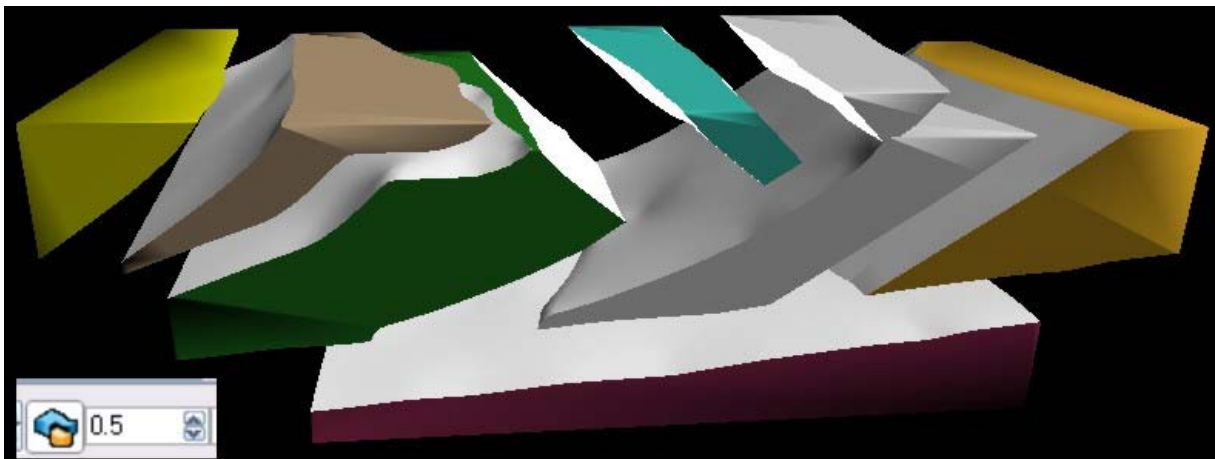


Fig. 12. e. 3D model of the fault blocks in an exploded view and „exploded view“ icon of the Quick Attribute Tool Bar.

12. 4. Modelling of faulted horizons

The steps needed for horizon modelling are similar to the fault modelling. First, a surface with data extend is created, then the fault contacts are specified and the fault displacement is applied. Finally the surface is re-interpolated (fig. f).

Horizon modelling

Build surfaces → *Horizons: horizon1 horizon2* → *Black arrow* → *use data extension* → *specify a default size* → *500* → *build surfaces*

Horizon-Fault contact modelling

Horizons: horizon1 horizon2 →Faults: thrust3 trace_thrust4 →Connect on volume of interest →Propose default contact lines →Apply contact lines →Next

Honor contacts

Horizons: horizon1 →Ignore/include data Ignore data points between →50 and faults

Create/update vertical throw →Create stratigraphic heave

Finetune the throw: tick on the „Get XYZ Coordinate“ button and read the z-values of the hanging and footwall horizons near the faults →Add the values to the table →ok

Create stratigraphic heave →Next

Refine Horizon contacts

choose one horizon and one fault →Edit contacts (blue arrow on the right side) → reshape contact with 4 control points move the control points until you get a straight contact line in the correct position →choose a low fault influence distance (100m) →Update horizon surface →Reshape the next contact

Remove horizon crossings →Horizons: horizon1 horizon2 →check crossings between horizons →set minimum thickness to a positive value (500m) →Remove crossings between horizons

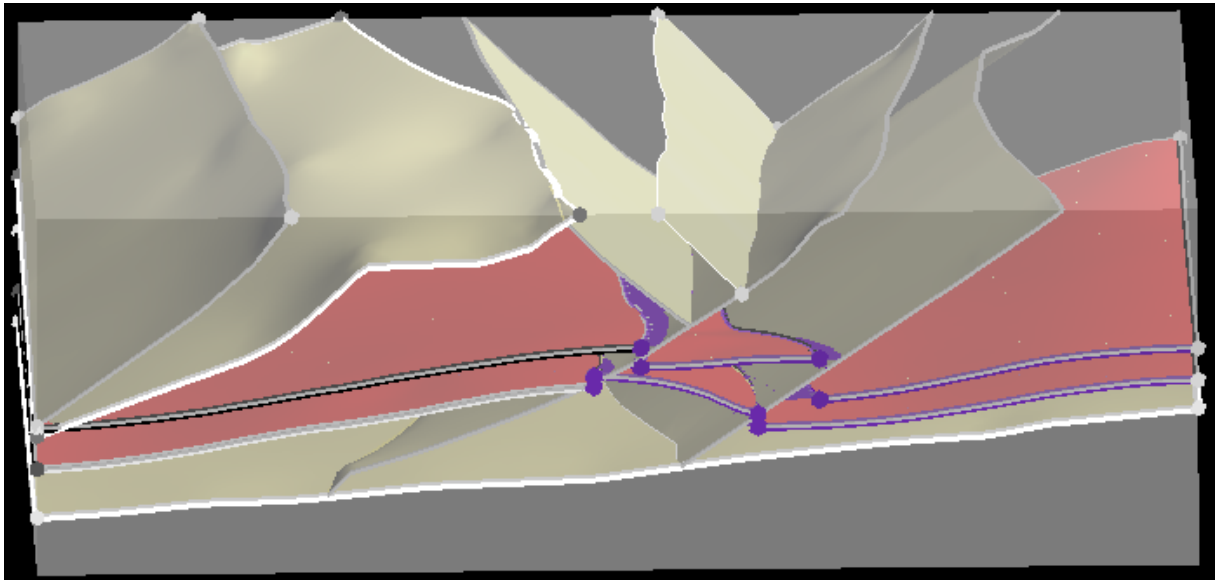


Fig. 12. f. Fault model with a displaced horizon (red).

12. 5. 3D Reservoir Grid Builder

The 3D Reservoir Grid Builder allows to generate SGrids with defined and varying grid size. The orientation of the grid can be chosen in any orientation in space and the grid can be displaced along faults (fig. g). So, modelling of grids in the workflow is much more comfortable than by hand.

Start the workflow

right click on 3D reservoir grid builder → new →ok →Using triangulated surfaces

Select horizons

Use top reservoir horizon: SM_study_horizon1_ts →Use base reservoir horizon: SM_study_horizon2_ts →Orientation →from top to base →area of interest → none →Next

Specify Gridding

Interactive → Prepare Top/Base Horizon → you can add/delete/edit fault traces replacing the SGrid

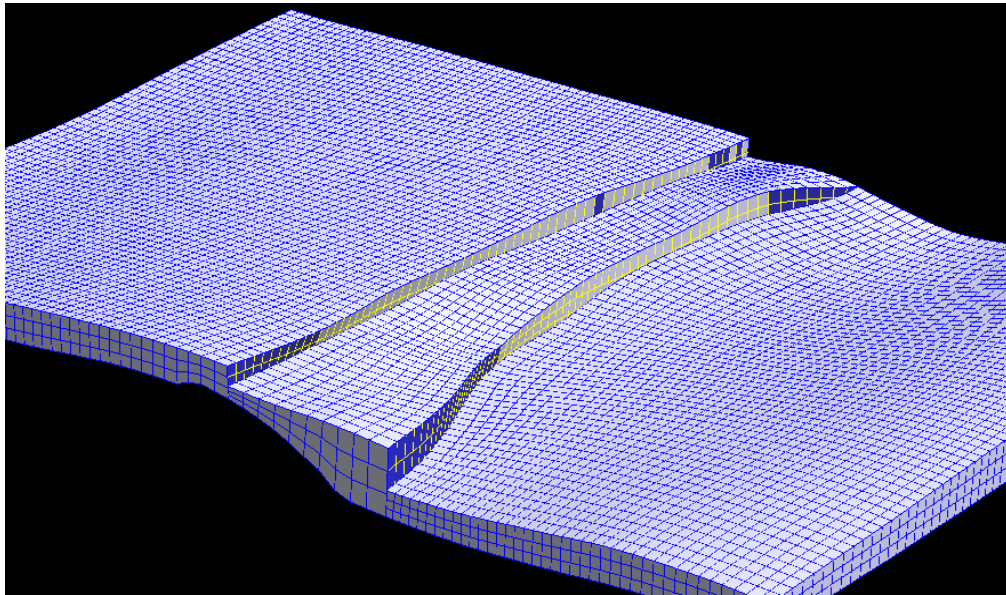


Fig. 12. g. Smooth fault-displaced SGrid that is aligned parallel to the fault traces.

Create Gridding

Create alignment on → X and Y axis creates voxel boundaries parallel to the coordinate system

Outer limits creates voxel boundaries parallel to the surface boundaries → Direction from angle/vector creates a grid with user-defined orientation (angle from north-direction) →

Align gridding → align on fault trace → pick the fault traces

Pillars from two horizons → Initialize Pillars → Next

Build Grid

Layering → increase number of layers to 3 → Next → Next

Build → Cell number: 200 → Guess J from I

Options → Fault traces → smooth

Create Grid

Move to the object panel → SGrid → strat_grid_3D_Grid_study → regions and tick block1, block2, block3

HOW TO COMMUNICATE MODELLING RESULTS

13. 1. Presentation of scales, compass direction and object names

In most cases, you will work for a client, but not for private purposes. The client gets a report, but he does not know the details of the project and of gOcad modelling. He needs to get explained the basics of the work to be enabled to understand modelling results. Additionally, he is especially interested in information about the modelled object that has to be extracted by the modeller..

Therefore, each print or slide of a gOcad project necessarily has to comprise descriptive information, otherwise the model is meaningless. An unlabeled picture is completely useless! So, don't forget to provide:

- scale (horizontal and vertical)
- vertical exaggeration
- compass direction
- object names
- legend for colorcoding of properties

A scale has to be drawn by hand as gOcad object that changes its size, when the 3D camera settings are changed. Probably the best way of adding scale bar, is to locate it along the bounding box. Measure the length of one bounding box edge and decide, how long the scale bar shall be. We choose 1 km. Next, digitize a curve object of two points starting at the corner of the box. Read the coordinates of the point at the box corner and then change the coordinate of the second point, such that the curve has the length that is wanted for the scale bar (1 km in this exercise).

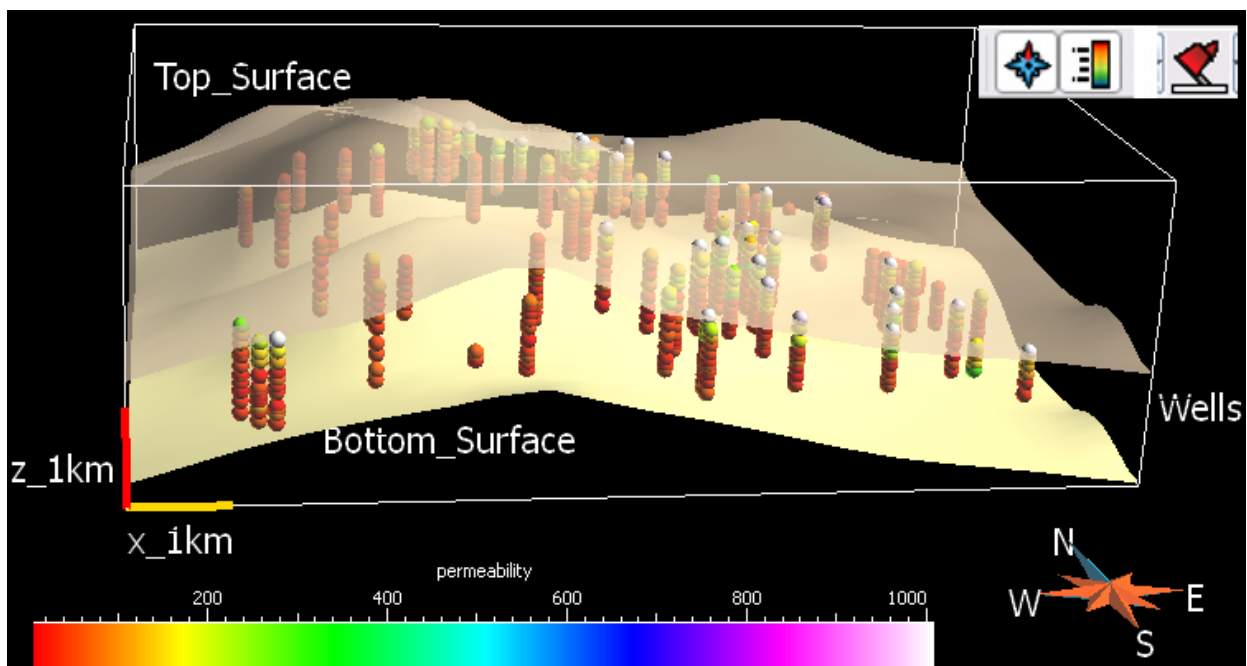


Fig. 13. a. Screenshot of a gOcad model with labels, scales and compass direction. The icons in the upper right corner are used for showing the orientation rose, the colormap and the object names.

Object panel → curve objects → right click → new → from digitized polyline → left click at the corner of the box → right click on the edge of the box

Camera tools → Get XYZ coordinates → pick the point at the corner → a window opens, leave it open

Curve → Tools → Node → Move to → add the coordinates of the second point (first point + 1000m) → ok → click on the second point

Change the name of the curve object to the length of the scale bar and show the name

A compass rose and a colocoding bar can be found in the Camera Settings Bar and have simply to be switched on. The banner symbol in the Quick Attribute Bar is used for displaying the name of an object. If this is not well visible, create a two-point curve object with the name you want to show, move it to the proper position and make the curve invisible. The compass rose and the colorbar usually have to take up the whole width of the picture that the numbers are readable.

	Median	Mean	Standard derivation	1st Quartil	3rd Quartil
data	44	108	191	26	115
DSI	35	44	160	-4	79
UVW space	47	87	110	34	82
XYZ space	41	94	138	30	88

Table 13. a. Statistical numbers for the property estimation

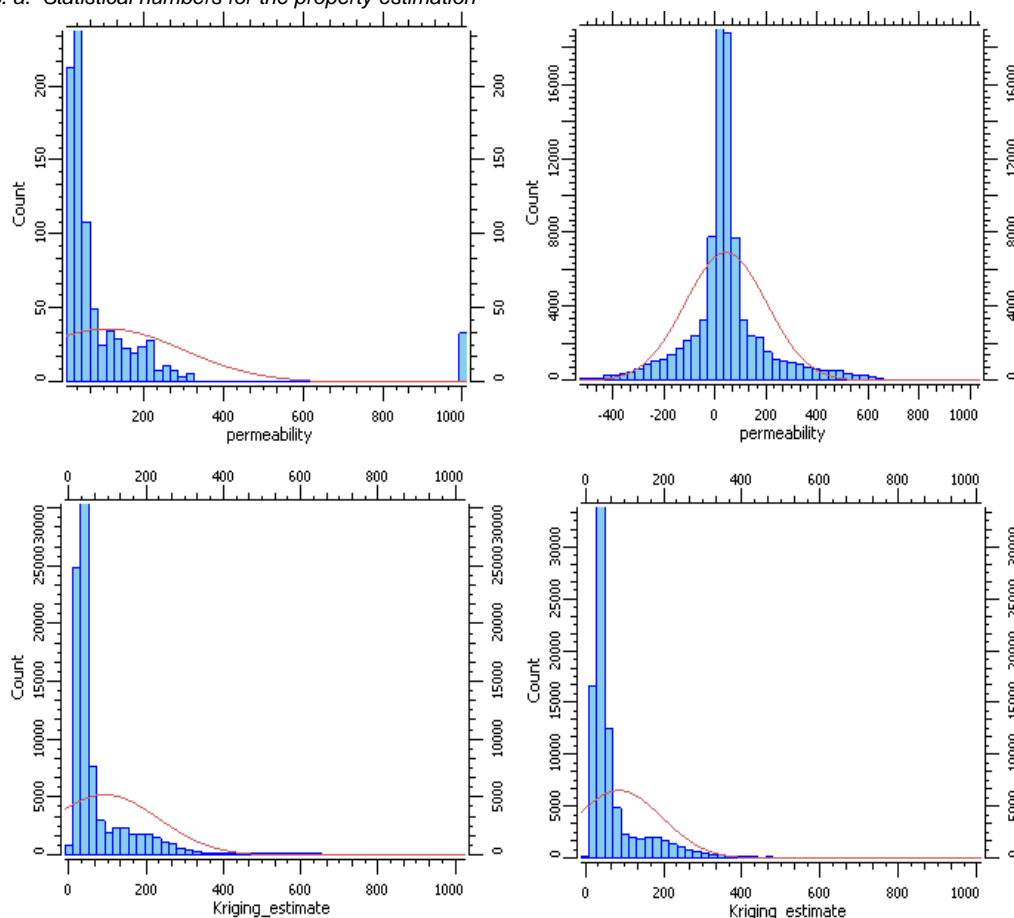


Fig. 13. b. Histograms of the permeability in the primary data set (1st row left), interpolated with DSI (1st row right) and with kriging in a XYZ (2d row left) and UVW space (2d row right).

13. 2. Analysis of the results of property modelling

If a property was modelled, the modelling result has to be analysed with simple statistic tools and the data distribution of the estimates has to be presented visually and described by characteristic numbers. You can work with the SGrid in the same way as with the primary data, this means you can produce cross plots and histograms and read statistical numbers. Use these tools to compare the interpolation results with the primary data. Usually, you will try several interpolation methods and make a decision, which one(s) is (are) the best.

In this example, the histograms and the statistical numbers show that DSI brings the worst modelling results. The distributions of the estimates in XYZ and UVW space are similar to the primary data and to one another. What do you recommend to your client concerning his problem? Where it's worth to drill? Where it's worth to proceed with mining, where not? Where to be careful? Where are the modelling results uncertain? Where do you need more data, what sort of data?

13. 3. Cutting through the model

13.3.1. The slicer

gOcad provides two tools that you can view your model from "inside".

The slicer allows moving through the model and creates a sequence of sections parallel to the faces of the slicer box. The slicer can be found in the camera tool bar. First, you have to activate the slicer function by clicking on the icon "*slicer*". Then, click on "*edit slicer*" and on one surface of the slicer box to define the slicing direction. Now, you can move through the model, if you keep the left mouse button pressed and move the mouse. The translation axis is displayed in yellow. If you don't see anything, click on "*reset slicer*" in the camera tool bar. The default slicing directions are parallel to the coordinate axis. If you want to change this, click on the "*rotate slicer*" icon and then on the slicer box and move it while pressing the left mouse button. The rotation axis is displayed in yellow. Finish the slicer mode by a second click on the slicer icon.

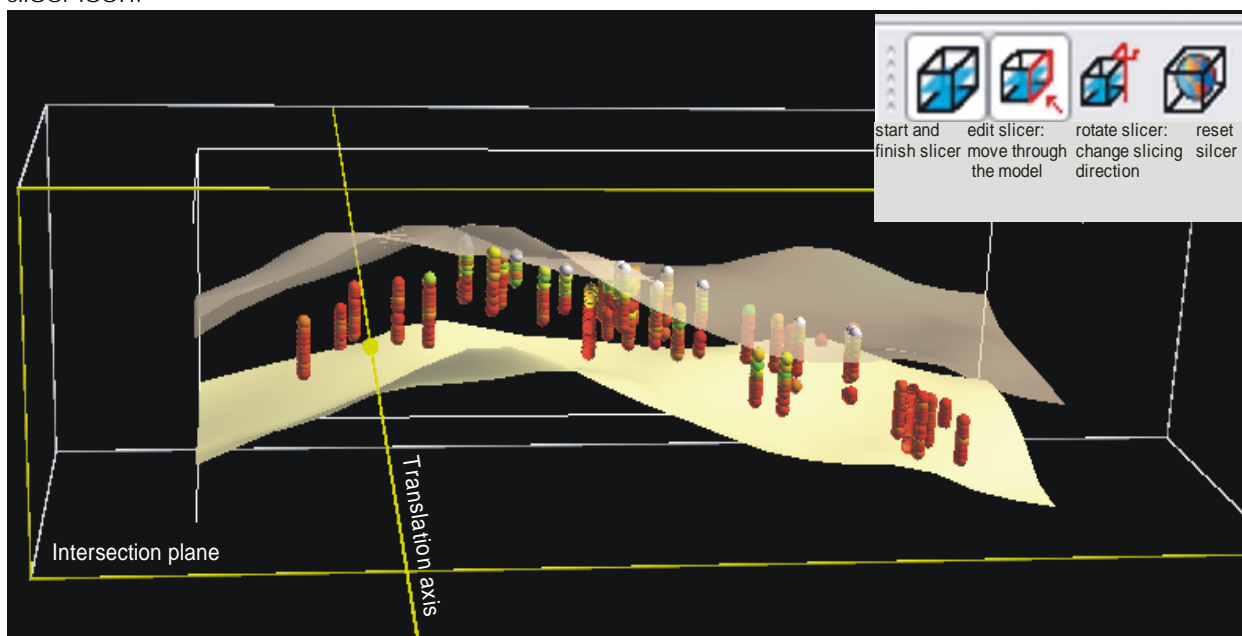


Fig. 13. c. The slicer is active. Upper right corner: the slicer icons

13.3.2. Cross sections

You can produce and export cross sections projecting gOcad objects in a plane. First, we want to create a cross section correlating drilling holes.

Object panel → cross section → right click → New → From digitized polyline → Digitize the cross section line

When you press the right mouse button for finishing the line, the cross section view is created. On the left side of the window, you see the object panel. Pick those objects you want to see in the cross section. You can also display the pillars.

At the right side of the window, you can see the cross section toolbar with icons similar to the other gOcad toolbars. Click on „*global view*“ to show the whole cross section. You can choose various zoom options to display a part of the cross section: *the box zoom and the strip zoom*, you may show or hide the grid. If you click on one object in the object panel, a *Quick Edit Bar* appears at the bottom of the window and you can edit the layout of the object. You can create a cross section showing one horizon in one color, if you tick the „*solid painting*“ option. Finally, you can display colorcoded properties. The cross section can be exported by clicking on the „*export as file*“ icon in the cross section toolbar.

If you close the cross section window, you see the location of the section in the 3D camera.

13. 4. Geocando

Geocando is an open source 3D viewer. The program is tailor-made for loading data produced in gOcad, this means your client can view gOcad objects without owning a gOcad licence, but since Geocando is a pure viewer, he cannot change the original data. Geocando was developed by Manuel Feige, a geoinformatics student in Freiberg, and is still developed in diploma and bachelor thesis.

The Geocando software and user guide can be downloaded from

<http://sourceforge.net/projects/geocando/>

<http://www.geo.tu-freiberg.de/~melzer/GeocandoManual.pdf>

Open Geocando by clicking on *Start → Alle Programme → Geocando*

Geocando can only load gOcad objects, but no project files. The objects can be merged as Geocando projects.

File → Add data → sgrid\objects

Select file type: gOcad

Load the surfaces *Top_surface*, *bottom_surface* and an *SGrid*!

The layout of the objects is edited in the „*style*“ folder, the layout of properties is edited in the „*property*“ folder.

The objects and visualisation settings are saved with

File → save project

13. 5. 3D PDFs

gOcad objects can be saved as 3D PDFs with the software Acrobat Pro Extended after making a screenshot. These PDFs can be included in PowerPoint presentations and the view direction is modified with the mouse.