GENERATING A 3D VELOCITY MODEL - THE GOCAD APPROACH

R. COGNOT¹, P. LAVEST¹, F. BOSQUET² and J.L. MALLET¹

¹ ENSG, Laboratoire d'Informatique et d'Analyse des Données, rue du doyen Roubaud,
54501 Vandoeuvre Cedex, France

² Elf Aquitaine Production

Introduction

Both of the Overthrust and the Salt Dome (SEG/EAEG) models have been built synthetically from a set of cross sections, into a set of surfaces, then converted into 3D velocity grids. In both cases, the GOCAD software has been intensively used for these operations, and the goal of this presentation is to focus mainly on the grid generation step, more specifically taking the example of the Overthrust model. It must be said that the methods that will be discussed here are very general, and has also been with success to the salt dome model.

The surfacic model

The surfacic model has been built from a set of vertical balanced cross-sections, and is composed of 16 interfaces, and 3 thrust planes. The interfaces divide the model into 17 layers. Two of the thrust planes divide the bloc into 3 separate regions, while the third one is blind and is dying out laterally. A subset of this model is shown in Figure 1. For our purpose, a small grid (100*100*100 cells) has been defined.

Handling grids in GOCAD: the VOXET¹ object

3D grids are an essential element especially when seismic is involved. For seismic modelling, we need to build a huge 3D velocity grid. This velocity grid should reflect the interfaces and fault geometry.

Specifications

They should handle the following functionalities :

- At least one parameter (velocity for example) must be attached to each grid node.
- It should be possible to handle discontinuities.
- It should be possible to handle regions.
- It should be possible to use an interpolation method for estimating unknown values from known ones.
- It should be possible to handle discontinuities acting as barriers for the interpolation of parameters on the grid.
- It should be possible to handle regions.
- It should be possible to use external data sources to initialize all or part of the grid.

Definition of a VOXET object

From these specifications, we have defined the VOXET object. This object is basically constituted of two separated parts:

- A parameter grid, storing the values for one parameter at each node. There may be more than one parameter grid attached to a single object.
- A flag grid. There is only one flag grid for a VOXET object. Each flag is an 8-bit integer holding:
 - 3 bits for connectivities in the UPWARD, RIGHTWARD and FORWARD directions. Connectivities in the other 3 directions are deduced. These flags will be used for modelling the discontinuities.
 - 1 bit is setting this point as DATA, that means fixed in an interpolation sense.
 - 1 bit is marking the point as being or not in the current processing region. That will enable region-clipping for some operations. By default, the current region is the whole grid.

Current limitations

Due to memory consumption, it is wise to limit the size of a grid to something that will be entirely handled in central memory. However, as will be developed, it is possible to compute the grid partially, and then to merge the final result.

For display purpose, and consistency checking, a VVOXET² object has been introduced. It allows the same visual functionalities as the VOXET object, but only the displayed planed of the grid are loaded in central memory. This allows the visualisation of virtually any grid.

Operations on a VOXET object: creating a velocity grid

Concepts

Three important concepts have to be introduced:

- External Painters, objects used to initialise values inside the grid. Currently, the grid can be initialised from :
 - set of points with a property attached.
 - triangulated surfaces with a property attached.
 - tetrahedrised volumes with a property attached.
 - another grid, with a possibly different sampling.
- Region definition. A region is defined by the set of grid nodes connected to a given node. Connectivities can be broken using surfaces, such as the interfaces and the faults defined in our model. Regions can be computed recursively.
- Region Clipping. Any operation can be limited to a region previously defined.

Example 1: assigning an index to a layer

In the case of the example shown on Figure 1, the hanging wall and foot wall are set as discontinuities in the grid, and so are the faults (see figure 2). A region is then defined recursively, from a point given in each subregion. Assigning an index to points in the current region is then straight forward (Figure 3).

This index can then be used to retrieve a given velocity or velocity law attached to that layer.

Example 2: initialising a layer with a painter

In that case, the surfaces are handled as in the previous one, and regions are defined in the same manner. However, a third object, holding a velocity law, is used to paint the grid. The painting is clipped to the current region (Figure 4).

This approach has the advantage to be independant from the grid sampling. It becomes easy to initialise pieces of the final grid with the same painter, and merge the results into the final grid.

¹acronym for volume cell set.

Conclusion

Up to now, for both the EAEG and SEG model, we have generated index grids. That means each layer has been given an index, which in turn can be converted to a velocity value or law. This has the main advantage to make the grid reusable for multiple velocity models for the same geometry.

Bibliography

1. Y. Chipot, (1991). *Génération et Modification de Surfaces Triangulées*. PhD report, Institut National Polytechnique de Lorraine-Nancy/France.
2. J.L. Mallet, (1989). *Discrete Smooth Interpolation*. ACM-Transactions on Graphics, Vol.8, No.2, pp 121-144.
3. J.L. Mallet, (1992). *Discrete Smooth Interpolation in Geometric Modelling*. CAD, Vol.24(4), pp178-191.

²for Virtual VOXET.

Figure 1: Overthrust model: one layer defined by its hanging wall, foot wall and faults. An empty grid has been defined, displayed here as a white box.

Figure 2: The grid is marked with discontinuities, shown here as lines on the external faces. They correspond to the computed intersections between the grid and the surfaces.

Figure 3: Assigning an index to a region. The indexed region is displayed here in black.

Figure 4: Initialising a grid using an external painter. The initialised layer now holds a non-constant velocity field.