2003-04-08

Building solid models from boreholes and user-defined cross-sections

Alan M. Lemon
Brigham Young University - Provo

Norman L. Jones
Brigham Young University - Provo, njones@byu.edu

Original Publication Citation

Lemon, A.M., N.L. Jones, “Building solid models from boreholes and user-defined cross-sections,”Computers and Geosciences, Vol. 29, No. 5, June, 2003, pp 547-555.

BYU ScholarsArchive Citation

Lemon, Alan M. and Jones, Norman L., "Building solid models from boreholes and user-defined cross-sections" (2003). *Faculty Publications*. 4293.
https://scholarsarchive.byu.edu/facpub/4293

PERGAMON

Computers & Geosciences 29 (2003) 547–555

COMPUTERS & GEOSCIENCES

www.elsevier.com/locate/cageo

Building solid models from boreholes and user-defined cross-sections

Alan M. Lemon, Norman L. Jones*

Environmental Modeling Research Laboratory, Brigham Young University, P.O. Box 24080 242 Clyde Building BYU, Provo, UT 84602, USA

Received 19 July 2002; received in revised form 27 January 2003; accepted 29 January 2003

Abstract

Solid models of geologic structures are useful tools for geologists and engineers. Solid models completely and unambiguously define the stratigraphy for the site being modeled, including complex boundaries and embedded seams. Past research has focused on the "set operations" approach to create solid models. Whereas the set operations approach is flexible, it requires significant user intervention and is therefore difficult to use. A simple approach for generating solid models from borehole data, called the horizons method, is presented. The horizons method can be used to build solids directly from borehole data with minimal user intervention. The user first assigns horizon ids to each of the borehole contacts. The horizon ids represent the depositional sequence and increase from the bottom to the top of the boreholes. The solids are then built by interpolating each of the surfaces defined by the horizons and extruding the surface into a solid. In each case, the solid is built by extruding the solid from the current surface down to the uppermost surface defined by the top of all previous horizons. In cases where more control over the resulting solids is necessary, the horizons method can be easily modified to honor user-defined cross-sections in addition to the borehole data.

Keywords: Solid modeling; Cross-sections; Stratigraphy modeling; Horizons method

1. Introduction

The "solid modeling" approach has been investigated by several researchers as a tool for constructing three-dimensional models of geologic structures (Bak and Mill, 1989; Bayer and Dooley, 1990; Fisher and Wales, 1990; Gjoystdal et al., 1985; Jones et al., 1993). The solid modeling approach was originally developed for representing three-dimensional objects in the Computer-Aided Design/Computer-Aided Manufacturing (CAD/CAM) industry (Braid, 1975; Krouse, 1985; Mantyla and Tamminen, 1983). The solid modeling approach completely and unambiguously defines the volume of a three-dimensional object. Solid models can be manipulated via set operations. For example, a new solid can be created by computing the volumetric union, difference or intersection of two solids. This feature can be a powerful tool when constructing detailed three-dimensional mechanical parts.

While several methods have been developed for defining solid models, the most popular method is called "boundary representation." With the boundary representation method, a solid model is defined by representing the outer surface of the solid. This surface is a collection of individual faces, typically quadrilaterals or triangles. Each face defines either a linear surface patch or a higher order surface such as a Bezier or B-spline surface.

Where the solid modeling method was originally designed for CAD/CAM applications in mechanical and

*Corresponding author. Tel.: +1-801-378-2812; fax: +1-801-378-2478.

E-mail address: njones@et.byu.edu (N.M. Jones).

0098-3004/03/$ - see front matter © 2003 Elsevier Science Ltd. All rights reserved.
doi:10.1016/S0098-3004(03)00051-7

aerospace engineering, it can be applied successfully to three-dimensional geologic structures. A sample solid model of a set of geologic units is shown in Fig. 1. Each component of the stratigraphy is represented by a separate solid. With a properly constructed set of solids, the boundaries of the solids all match precisely with no voids or overlaps. Solid modeling can be used to model stratigraphy at almost any level of complexity. Pinch-outs, embedded seams and faults can all be directly represented in the solid model geometry.

Our motivation for utilizing the solid modeling method is to build geologic model for applications in ground water modeling. Most ground water models are based on the finite difference or finite element methods. These methods require the construction of 3D structured grids or unstructured meshes. In both cases, the grid or mesh geometry is manipulated and the aquifer properties are assigned to cells and elements in a manner that approximates the stratigraphy at the site being modeled. This discretization process can be extremely challenging with complex geology. Alternatively, solid modeling can be used to develop a conceptual model of the site stratigraphy that is, independent of the grid or mesh geometry. A mesh or grid can then be automatically generated from the solid model geometry (Jones et al., 2002). This approach is particularly powerful when coupled with the new Hydrogeologic Unit Flow (HUF) package (Anderman and Hill, 2000) utilized by MODFLOW 2000 (Harbaugh et al., 2000). The HUF package allows the model stratigraphy to be defined in a grid-independent fashion. When the simulation is performed, equivalent hydraulic properties for individual cells are computed by the HUF package based on the stratigraphy overlapped by each grid cell. The solid modeling approach is an ideal tool for developing input data for the HUF package. The top elevation and thickness arrays required by the HUF package for each geologic unit can be easily extracted from solid model geometry.

While solid models have a variety of applications, constructing solid models of complex stratigraphy can be challenging. One method that has been used considerably is the "set operations" approach illustrated in Fig. 2. In the first step of the process, triangulated irregular networks (TINs) are created at the tops and bottoms of geologic units (Fig. 2A). In the next step, the TINs are extruded vertically to build primitive solids (Fig. 2B). The overlapping primitive solids are then modified using set operations (Fig. 2C) to generate the final non-overlapping solids (Fig. 2D). While the set operations method is highly flexible, it has some serious limitations. First of all, each step of the TIN extrusion and set operations processes requires user intervention and guidance and must be done in a specific order. While this process is fairly straightforward for simple sites, it can be quite difficult for sites with complex geology to keep track of all of the extrusions and set operations required to correctly represent the stratigraphy without overlaps or voids between the solids. Furthermore, the calculations involved in the set operation process have traditionally been prone to accumulated round off error leading to numerical instability. This is particularly true when the two solids involved in a set operation have coincident faces. The TIN extrusion and set operation process illustrated in Fig. 2 is guaranteed to result in numerous coincident faces, making round off error a significant problem.

In this paper, we present a new method for developing solid models of geologic units called the "horizons" approach. This method is ideally suited for alluvial systems and is simpler and more robust than the set operations approach. The horizons method can be used to build solids directly from borehole data with minimal user intervention. In cases where more control over the resulting solids is necessary, the horizons method can be easily modified to honor user-defined cross-sections in addition to the borehole data. The horizons concept itself is not entirely new. For example, Tipper (1993) describes a horizon-based approach using a triangular decomposition for reconstructing surfaces from boreholes and seismic profiles. We believe that the contributions made by our approach are: (1) it generates boundary representation solid models, (2) it is extremely

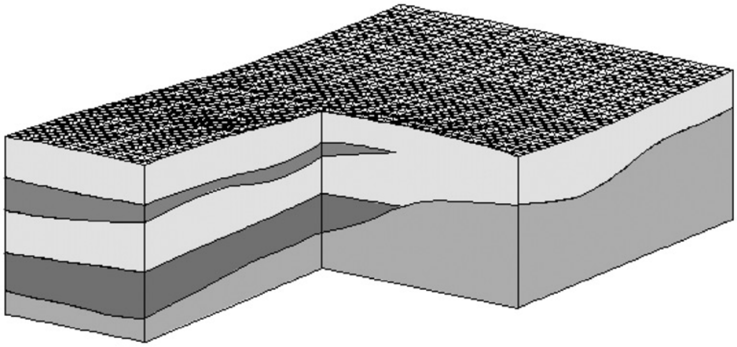

Fig. 1. Sample solid model in cut-away view.

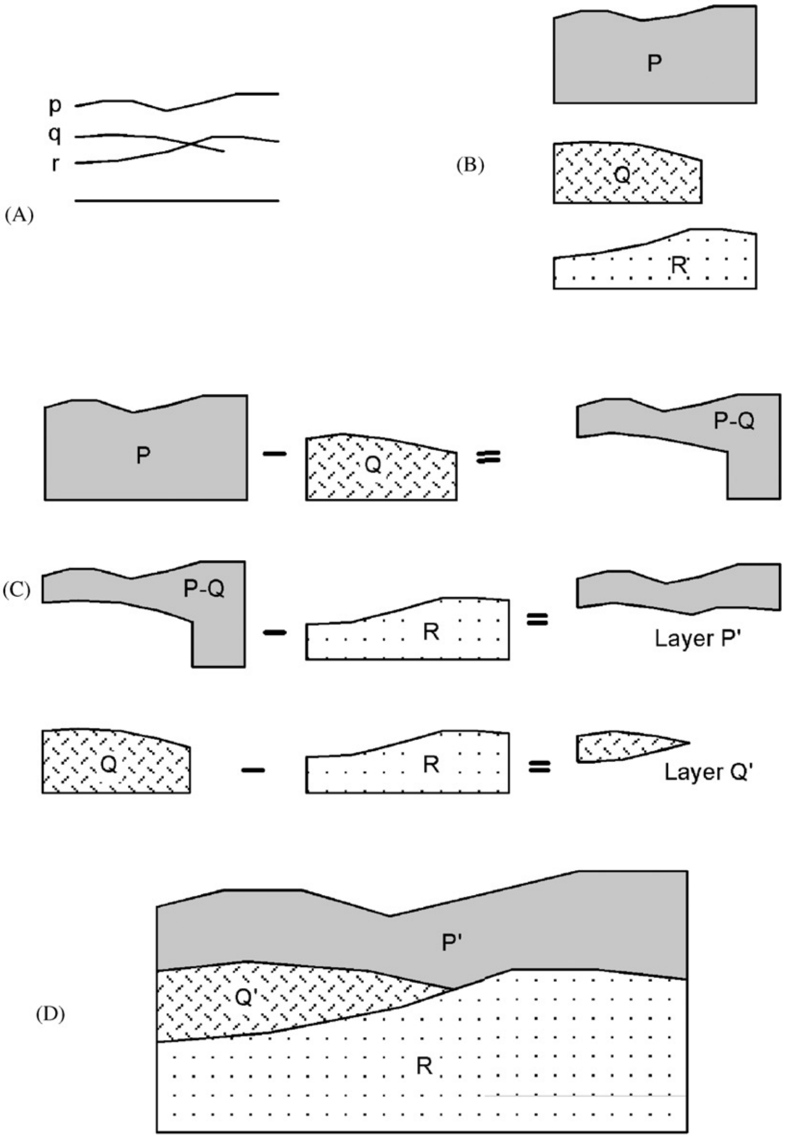

Fig. 2. Set operations approach for building solid models: (A) TINs representing tops and bottoms of geologic units, (B) primitive solids produced by extruding TINs, (C) set operations used to modify primitive solids, and (D) final solids resulting from set operations.

simple and robust, and (3) it utilizes user-defined cross-sectional data.

2. Horizons to solids algorithm

The horizons-to-solids algorithm uses borehole data to create solid models of stratigraphy in a simple, intuitive fashion. Borehole data are organized into segments and contacts. A contact is defined as the interface between two adjacent stratigraphic units. Segments occur between contacts and are associated with a material (silt, sand, clay, etc.). Each contact has a location (x, y, z), a horizon id, and two segment ids (one for the material above and one for the material below). The following is an explanation of the main steps in the horizons approach. The steps defined below represent the horizons approach applied to borehole data only. The modifications required to the algorithm in order to support cross-sections in addition to the borehole data are described in the next section.

Step 1: *Assign horizon ids*. The first step in the process is to assign horizon ids to the borehole contacts. In the context of this algorithm, a horizon is defined as a surface representing the top of a geologic unit in a depositional sequence. The horizon id represents the order in the sequence. The horizon ids should start at 1 and increase from the bottom to the top, where 1 is associated with the top elevation of the bottom most geologic unit.

The horizons concept is illustrated in Fig. 3. A set of boreholes with horizon id's assigned to the contacts is shown in Fig. 3A. The set of solids resulting from the horizon assignments is shown in Fig. 3B. Conceptually, the solids are formed by an ordered extrusion process that proceeds from the bottom to the top. First, a surface is created by interpolating all of the contacts with a horizon id equal to 1. The solid corresponding to unit 1 is then found by extruding the resulting surface down to a bottom elevation. A second horizon surface is then formed by interpolating the contacts with a horizon id = 2. This surface is then extruded down to the top of the solid for horizon 1. In some regions of the site, the surface for horizon 2 will be below the top of the solid for horizon 1. The solid for horizon 2 is clipped at the intersection of the surfaces for horizons 1 and 2 so that it does not extend into these regions. In a similar fashion, the solid for horizon 3 is extruded down to the current uppermost elevation of horizons 1 and 2. In general, each horizon is extruded down to a surface representing the topmost profile of all of the preceding horizons.

When assigning horizon ids to contacts, care must be taken to number the horizons in a fashion that is consistent with this bottom-to-top depositional sequence. There is no limit to the number of horizons that may be used. If a horizon has an id of zero, the corresponding contact will be ignored in the extrusion process. This makes it possible to ignore small seams in the borehole data that are not sufficiently significant to be explicitly represented in the final model.

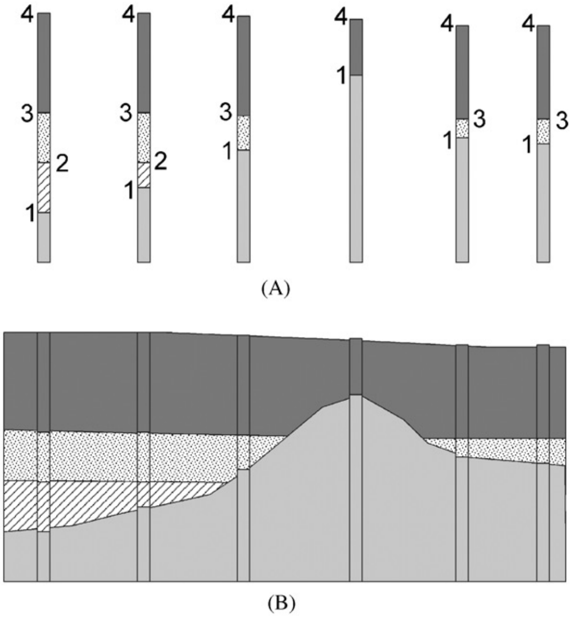

Fig. 3. Horizons concept: (A) Horizon ID's assigned to borehole contacts, and (B) Solids resulting from horizon assignments made in (A).

Step 2: *Define the primary TIN*. The second step is to define the "primary TIN" using a standard triangulation algorithm (Field, 1991; Lawson, 1986; Watson, 1981). The primary TIN serves two basic purposes: (1) It defines the outer boundary of the solids, and (2) it is used to establish the topology of the solids. The faces defining the volume enclosed by a solid model are composed of triangles. In the sample solids of Fig. 1, the triangular faces can be seen on the top of the solids. The polygonal faces on the sides of the solid are also composed of triangles, but the triangle edges on the interior of the polygons have been hidden since the triangles in each polygon are co-planar. Also, the density of the triangles in the primary TIN controls the density of the triangles that make up the solid. The primary TIN defines a common triangle topology or "template" that is used for extruding each of the horizon surfaces. Using a consistent topology for the horizons is a key to simplicity and robustness of the horizons approach. This will be illustrated in the discussion for Step 4.

Step 3: *Interpolate horizon elevations*. The third step is to interpolate the horizon elevations from the borehole contacts to the primary TIN to define the horizon surfaces. Conceptually, each of these surfaces can be thought of as an independent TIN. However, since each TIN has the same topology, there is no point in duplicating the primary TIN for each of the N horizons. A simpler approach is to represent each horizon surface as a separate elevation array associated with the vertices of the primary TIN.

Any interpolation scheme could be used to interpolate the horizon elevations. However, the selected scheme must support extrapolation. This is necessary since the primary TIN may cover an area larger than the convex hull of the boreholes. It is also helpful to use a relatively simple interpolation scheme since it makes it easier to automate the interpolation process. For example, the kriging technique produces excellent results, but requires the user to build a variogram for each set of horizons. This can be difficult for sites with large numbers of horizons and/or horizons associated with an insufficient number of contacts to develop a meaningful variogram. In our implementation of the algorithm, we utilize the inverse distance weighted (IDW) (Shepard, 1968), natural neighbor (Watson and Phillip, 1987), and kriging (Deutsch and Journel, 1992) methods.

In addition to the TINs defined by interpolating the horizon elevations, it is often useful to define two additional TINs: a top TIN and a bottom TIN. The top

TIN is used to define the very top of the depositional sequence and it corresponds to the terrain elevations. The top TIN is generally interpolated from digital elevation data, resulting in a more accurate surface since these data are typically more dense than the elevations from the tops of the boreholes. The bottom TIN typically represents the bedrock elevations.

Often not all of the horizons are present on a given borehole. For example, in Fig. 3A, the third borehole from the left is missing horizon 2. When interpolating horizon 2 the information from this borehole will be ignored because the horizon is not present. In some cases it is desirable to include a missing horizon at a borehole so that a horizon will pinchout. This can be accomplished using an "implicit pinchout" option. If the implicit pinchout option is on, then when horizon 2 is being interpolated to the primary TIN the elevation of the contact associated with horizon 1 at this borehole is also used. Thus the elevation of horizon 2 will be the same as horizon 1 at this borehole causing horizon 2 to pinchout.

Step 4: *Intersect horizon surfaces*. The fourth step is to intersect the TIN surfaces defining the horizons. Each TIN is intersected with each of the other TINs. Normally, intersecting two TINs can be a computationally expensive process since each triangle of one TIN must be checked against each triangle of the other TIN. However, since each of the horizon TINs have the same topology (they are identical in plan view), the intersection process can be significantly accelerated. This is because a triangle from the first TIN can only intersect the corresponding triangle from the second TIN. Thus, the intersection process is $O(N)$ rather than $O(N^2)$. This leads to a substantial advantage in terms of processing time compared to the set operations approach (which involves numerous surface intersections) described in the introduction.

Starting with the bottom TIN, each TIN is intersected with each of the other TINs. Fig. 4A shows a set of triangles from two different TINs that intersect. When considering the middle triangle, we find the points of intersection (P, Q) on the edges of the triangle. Then we insert each of the points into the TIN. Fig. 4B and C shows how the TIN is modified with the insertion of each of the intersection points. Notice that not only is edge P–Q now honored, but the adjacent triangles have also been modified.

When the new points are inserted we also compute the elevation for each horizon at these points. Since the points lie on the edge of a TIN triangle we use a simple linear interpolation. The modified TIN is used to perform all subsequent surface intersections and points are continuously added as intersections occur. In this manner, the primary TIN is modified so that each line of intersection resulting from all possible TIN intersections is explicitly represented in the primary TIN as an edge.

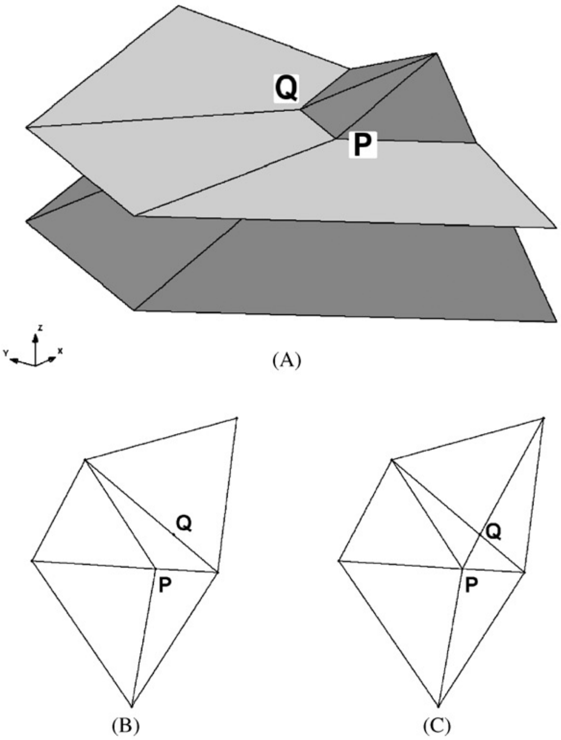

Fig. 4. Horizon intersections: (A) Example of triangles from different horizon TINs intersecting, (B) Modified "primary" TIN after inserting point A, and (C) Modified "primary" TIN after inserting point B.

Step 5: *Adjust horizon elevations*. The fifth step is to adjust the elevations of the different horizons on the primary TIN. In keeping with the "bottom to top" horizons concept described in Step 1, the elevation of a given horizon cannot go below the elevation of any of the lower horizons. We accomplish this by looping through the vertices of the primary TIN. For a given TIN vertex, we loop through each horizon from the bottom to the top. At each horizon, we compare the current elevation with the elevation of the next (higher) horizon. If the elevation of the next horizon is below the current horizon then the elevation of the next horizon is set equal to the elevation of the current horizon. This process is repeated for all horizons.

Step 6: *Build solids*. At this point, we are ready to extrude the horizon surfaces and build the solids. In the simplest case, one solid is constructed for each horizon. Each solid is constructed by building a set of triangles defining the faces of the solid from the horizon surfaces. This includes a set of triangles at the top and bottom of the solid coinciding with the triangles of the primary TIN and it may include a set of vertical triangles on the outer boundary of the site connecting the top and bottom of the solid. To generate the solids, we loop through the horizons. For each horizon we loop through the triangles of the primary TIN. For a given triangle, if the elevation for the current horizon for any of the three triangle vertices is above the elevation of the previous horizon then we build two solid faces, one at the top and one at the bottom. Also, if any of the edges of the triangle coincide with the outer site boundary, two vertical triangles are created representing the four-sided face on the side of the solid connecting the top and bottom triangles.

The process of forcing the TIN intersections into the primary TIN as described in Step 5 is critical to the success of formulating the solid boundaries. Since the intersections between horizons are explicitly represented in the primary TIN, none of the triangles in the primary TIN crosses the boundary for the solid corresponding to a particular horizon. This ensures that the faces for each solid will precisely match the faces of each adjacent solid and there will be no voids or overlapping solids.

Once all of the triangle faces are created for a particular horizon, these faces are used to build the solid corresponding to the horizon. In some cases, this solid can be disjoint. Disjoint solids occur as a result of being truncating by upward thrusting lower horizons, as is the case with horizon 3 in Fig. 3B. In other cases, the disjoint solid may have no connectivity to any borehole with the same material as the solid. This type of disjoint solid is called an "orphan solid" and is illustrated in Fig. 5. Orphan solids are sometimes the result of oscillation in the interpolation scheme being used. The orphan solids are deleted prior to proceeding to the next horizon.

2.1. Horizons to solids example

A sample application of the horizons algorithm for building solids is shown in Fig. 6. The borehole data with the horizons assigned to the contacts are shown in Fig. 6A. A cut-away view of the horizon TINs resulting from interpolating the horizon elevations is shown in Fig. 6B. The solids resulting from extruding the horizon TINs are shown in Fig. 6C. A set of cross-sections cut from the solids is shown in Fig. 6D.

3. Supplementing borehole data with cross-sections

One disadvantage of the horizons algorithm for building solid models is that it is highly dependent on the interpolation process. If the number of contacts associated with a particular horizon is sparse (which is often the case), the interpolation option selected may have a significant effect on the results. In some cases, none of the interpolation options may result in a satisfactory solution. This problem may be compounded by the fact that the interpolation process occurs globally over the entire model domain, when the geologic unit being modeled is local in nature. This problem is illustrated in Fig. 6. Notice the dark lens corresponding to horizon 3 in the borehole logs in Fig. 6A. This lens ends up being distributed over most of the model domain as shown in the cross-sections in Fig. 6D.

One of the more powerful features of the horizons method is that it can be easily modified with additional user data in cases where the interpolation from the boreholes is unsatisfactory. An effective approach is to allow the user to manually sketch cross-sections between boreholes indicating the desired geologic interpretation. In our implementation of this algorithm, we allow the user to define a cross-section by selecting any two boreholes. The boreholes appear in a "cross-section editor" with one hole on the left side of the window and one on the right. The user then draws a set of polylines that connect the contacts and define polygonal zones

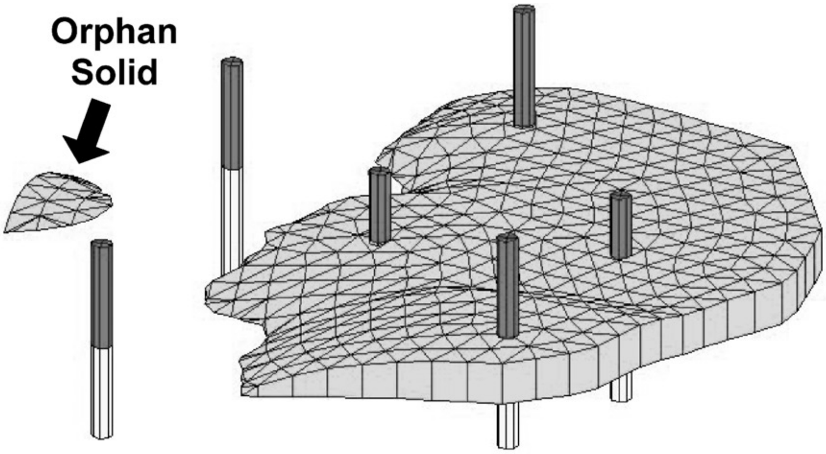

Fig. 5. Disjoint solid resulting from horizon extrusion process.

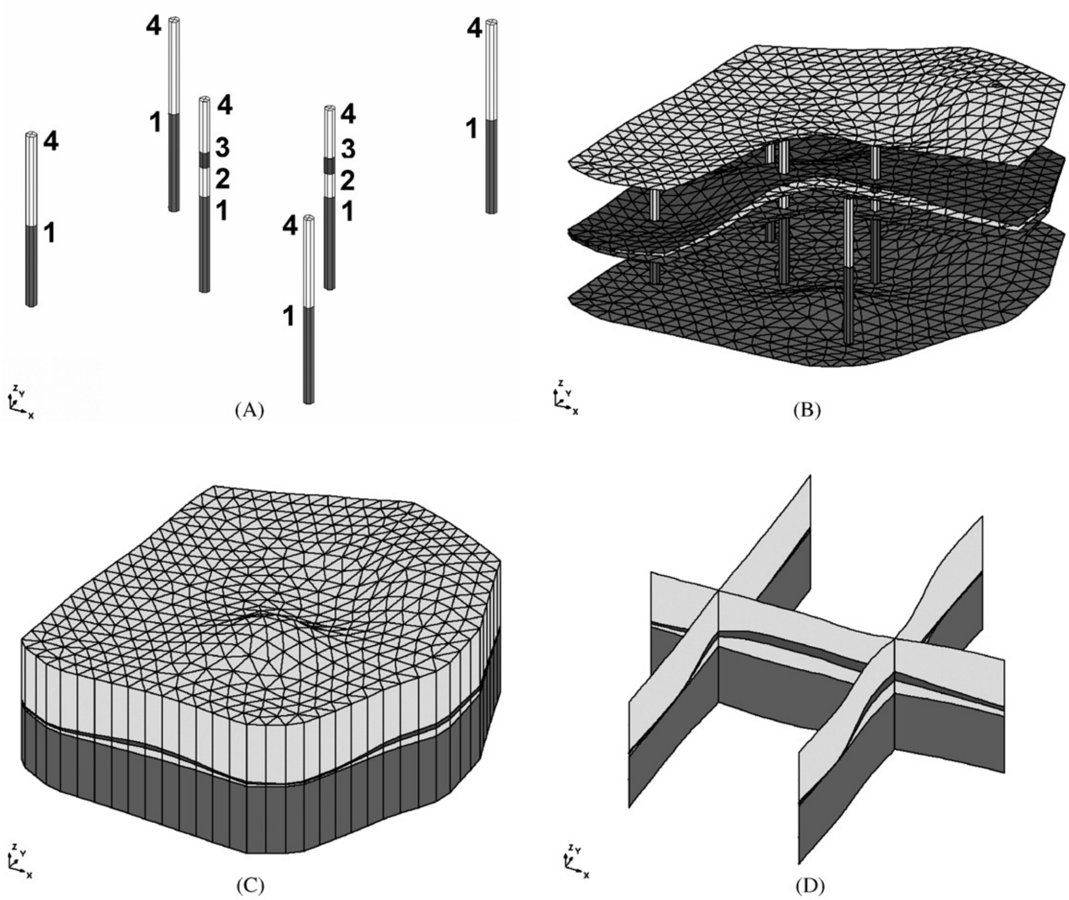

Fig. 6. Example of horizons algorithm: (A) Boreholes with assigned horizons, (B) cut-away of TINs from horizon interpolation, (C) solids created from horizons algorithm, and (D) cross-sections through solids.

representing the geologic units. Cross-sections can be created for any number of pairs of boreholes.

When the horizons algorithm is initiated with both borehole data and cross-sections, we repeat the process described above, but add an additional step just prior to Step 3. Before the horizons are interpolated, we extract a set of horizon elevations from the cross-sections and add them to the horizons from the borehole contacts prior to the interpolation step. Each of the points defining the cross-section boundaries is assigned a horizon id consistent with the borehole contact attached to the cross-section boundary. This process is illustrated in Fig. 7. The interpolation for each horizon surface then includes the elevations from both the contacts on the boreholes and the points on the user-defined cross-sections.

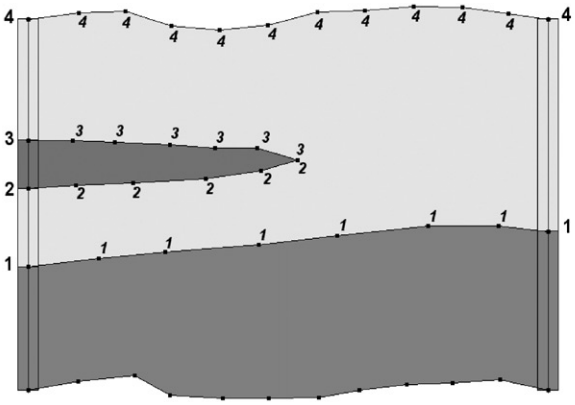

Fig. 7. Example of how cross-section data inherits horizon id from boreholes.

A sample application of the cross-section approach is illustrated in Fig. 8. A user-defined cross-section explicitly defining the extent of the lens corresponding to horizon 3 is shown in Fig. 8B. Fig. 8C shows a cross-section through the solids created by the horizons algorithm applied to the borehole contacts only. In this case, horizon 3 is a continuous layer in the generated solids. Fig. 8D illustrates a cross-section through the

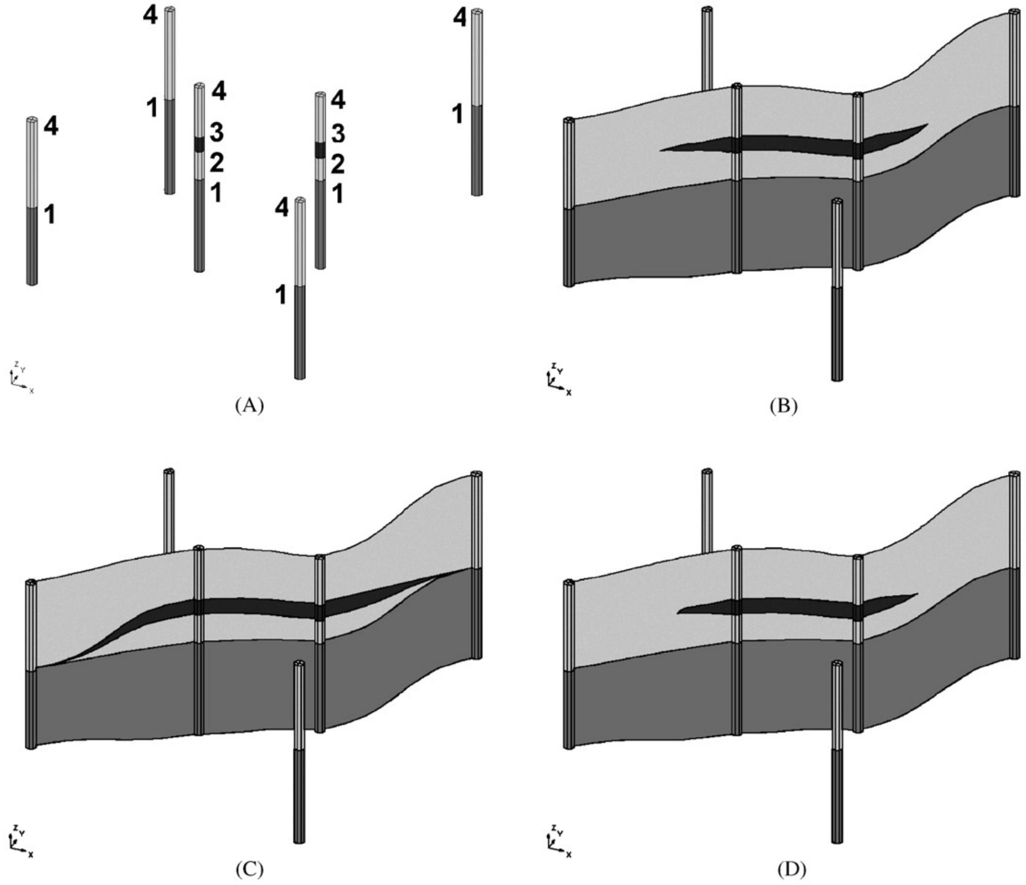

Fig. 8. Example of horizons algorithm with cross-section data: (A) boreholes with assigned horizons, (B) user-defined cross-sections, (C) cross-section through solids created without user-defined cross-sections, and (D) cross-section through solids created with user-defined cross-sections.

solids created by supplementing the horizon ids at the borehole contacts with user-defined cross-sections. Notice that the cross-section in Fig. 8D accurately reproduces the user-defined cross-section in Fig. 8B.

4. Pence Ranch site

The horizons method was applied to the Pence Ranch site in Idaho to test the usefulness of the method on a site with complex geology. The site was modeled using ten horizons. Cross-sections were created between the boreholes to help guide the interpolation of horizons. Fig. 9A shows the boreholes and user-defined cross-sections for the site. Fig. 9B shows cross-sections through the solids created by the horizons method.

The horizons algorithm was tested with this set of boreholes using several different "primary" TINs. The testing was performed on a PIII 933 MHz PC. The TINs all had the same outer boundary, but differed in the number of triangles. The smallest TIN had 1290 triangles and the largest 32,250. The computation times for the various TINs are shown in Table 1. As expected, the computation time is linear with respect to the number of triangles in the primary TIN.

Table 1
Computation time of horizons algorithm for Pence Ranch data using different triangle densities for "primary" TIN

Number of TIN triangles	Computation time (s)
1290	34
5160	64
11,610	108
20,640	172
32,250	251

5. Conclusions

The horizons method for building solid models of geologic structures is a significant improvement over

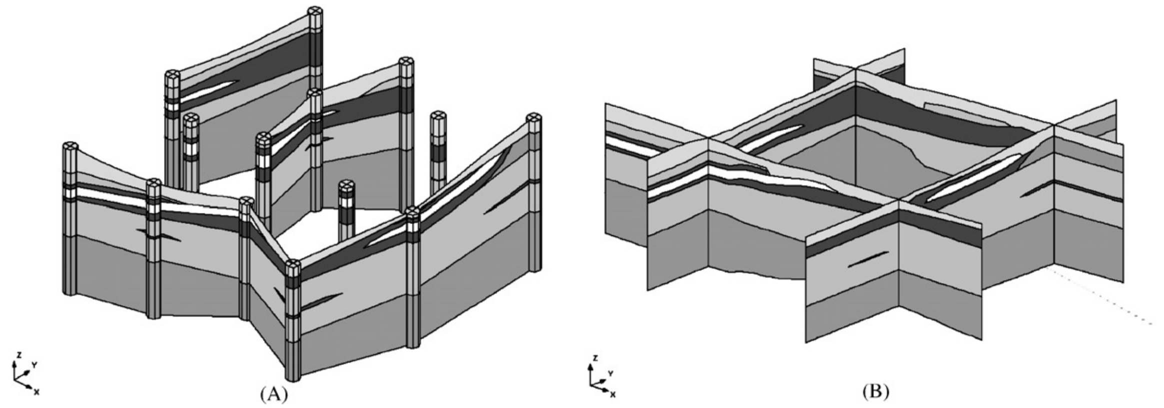

Fig. 9. Pence Ranch site: (A) boreholes and user-defined cross-sections for Pence Ranch data, and (B) Cross-sections from solids created with horizons algorithm.

previous methods based on user-defined set operations. The horizons technique is more robust, since most of the co-linear and co-planar checks inherent with set operation calculations can be avoided. It is also substantially more efficient since the surface-to-surface intersections occur with TINs that are identical in the *xy* plane, resulting in intersection calculations that are O(N) rather than O(N^2). Perhaps the most significant feature of the horizons algorithm is that it can be augmented with user-defined cross-section data. This allows the user to precisely control the shape of the resulting solids using a tool that is simple and familiar to geologists and hydrogeologists.

Acknowledgements

The authors wish to thank Ron Andrus and Ken Stokoe for providing the data used to the build the Pence Ranch model. We would also like to thank John Tipper and Michael McCullagh for their helpful suggestions for improving the paper.

References

Anderman, E.R., Hill, M.C., 2000. MODFLOW-2000, The US Geological Survey modular ground-water model—documentation of the hydrogeologic-unit flow (HUF) package. US Geological Survey Open-File Report 00–342, Denver, Colorado, 89pp.

Bak, P.R.G., Mill, A.J.B., 1989. Three dimensional representation in a Geoscientific Resource Management System for the minerals industry. In: Raper, J. (Ed.), Three Dimensional Applications in Geographic Information Systems. Taylor & Francis, New York, pp. 155–182.

Bayer, E., Dooley, K., 1990. New techniques for the generation of subsurface models. In: The 22nd Annual Offshore Technology Conference. Houston, Texas, May 7–10, pp. 243–248.

Braid, I.C., 1975. The synthesis of solids bounded by many faces. Communications of the ACM 18 (4), 209–216.

Deutsch, C.V., Journel, A.G., 1992. GSLIB: Geostatistical Software Library and User's Guide. Oxford University Press, New York, 369pp.

Field, D.A., 1991. A generic Delaunay triangulation algorithm for finite element meshes. Advances in Engineering Software 13 (5/6), 263–272.

Fisher, T.R., Wales, R.Q., 1990. 3D solid modeling of sandstone reservoirs using NURBS. Geobyte 5 (1), 39–41.

Gjoystdal, H., Reinhardsen, J.E., Astebol, K., 1985. Computer representation of complex 3-D geological structures using a new solid modeling technique. Geophysical Prospecting 33, 1195–1211.

Harbaugh, A.W., Banta, E.R., Hill, M.C., McDonald, M.G., 2000. MODFLOW-2000, The US Geological Survey modular ground-water model—user guide to modularization concepts and the ground-water flow process. US Geological Survey, Open-File Report 00–92, Reston, VA, 130pp.

Jones, N.L., Wright, S.G., 1993. Subsurface characterization with solid models. American Society of Civil Engineers, Geotechnical Engineering Journal 119 (11), 1823–1839.

Jones, N.L., Budge, T.J., Lemon, A.M., Zundel, A.K., 2002. Generating MODFLOW grids from boundary representation solid models. Ground Water 40 (2), 194–200.

Krouse, J.K., 1985. Solid modeling catches on. Machine Design 57 (3), 60–64.

Lawson, C.L., 1986. Properties of *n*-dimensional triangulations. Computer-Aided Geometric Design 3, 231–246.

Mantyla, M., Tamminen, M., 1983. Localized set operations for solid modeling. Computer Graphics 17 (3), 279–288.

Shepard, D., 1968. A two-dimensional interpolation function for irregularly spaced data. In: Proceedings of the 23rd National Conference of the ACM, pp. 517–523.

Tipper, J.C., 1993. Reconstructing three-dimensional geological systems: 1. The topology and geometry of horizons in the subsurface. Geoinformatics 4 (3), 199–207.

Watson, D.F., 1981. Computing the *n*-dimensional Delaunay tessellation with application to Voronoi polytopes. The Computer Journal 8 (2), 167–172.

Watson, D.F., Phillip, G.M., 1987. Neighborhood based interpolation. Geobyte 2 (2), 12–16.