Buletinul Ştiinţific al Universităţii "POLITEHNICA" din Timişoara
Seria HIDROTEHNICĂ
TRANSACTIONS on HYDROTECHNICS
Tom 58(72), Fascicola 2, 2013

MODFLOW–USG: An Unstructured Grid Version of MODFLOW for Simulating Groundwater Flow and Tightly Coupled Processes Using a Control Volume Finite-Difference Formulation

Robert F. Beilicci[1], Erika Beilicci[1]

Abstract – A new version of MODFLOW, called MODFLOW–USG (for UnStructured Grid), was developed to support a wide variety of structured and unstructured grid types, including nested grids and grids based on prismatic triangles, rectangles, hexagons, and other cell shapes. Flexibility in grid design can be used to focus resolution along rivers and around wells, for example, or to subdiscretize individual layers to better represent hydrostratigraphic units. MODFLOW–USG is based on an underlying control volume finite difference (CVFD) formulation in which a cell can be connected to an arbitrary number of adjacent cells. To improve accuracy of the CVFD formulation for irregular grid-cell geometries or nested grids, a generalized Ghost Node Correction (GNC)

Key words: structured and unstructured grid types, control volume finite difference

I. INTRODUCTION

The U.S. Geological Survey (USGS) develops and supports the MODFLOW computer program for simulation of three-dimensional, steady-state and transient groundwater flow. The standard MODFLOW releases (McDonald and Harbaugh, 1988; Harbaugh and McDonald, 1996; Harbaugh and others, 2000; Harbaugh, 2005) are all based on a rectangular finite-difference grid. There are two notable restrictions with a standard finite-difference grid. The first is that irregularly shaped domain boundaries cannot be easily fitted with a rectangular grid. Although there are options for inactivating parts of the grid outside the domain of interest, the domain is still bounded by rectangular grid cells that may not follow irregular boundaries; as a result, information about the entire grid, including inactive cells, is read and processed. The second limitation of a rectangular finite-difference grid is that it is difficult to refine the grid resolution in areas of interest. Column and row widths can be variably spaced in order to focus grid resolution, but the added resolution must be carried out to the edges of the grid.

There have been a number of efforts to relieve the restrictions of the rectilinear finite-difference grid required by MODFLOW. These efforts have primarily focused on implementing curvilinear grids and nested grid methods.

Curvilinear grids have been implemented for MODFLOW - based codes by Romero and Silver (2006), for example, but the approach has not been widely used. Use of nested grids with MODFLOW, however, has been a common approach for adding targeted resolution to areas of interest. The simplest nested grid approach is to use heads or fluxes from a regional model as boundary conditions for a higher resolution child model. This one-way coupling is commonly called telescopic mesh refinement (TMR). Leake and Claar (1999) developed the MODTMR computer program to facilitate the design of child grid boundary conditions from the output of a regional parent model. Mehl and Hill (2002, 2004, and 2005) improved upon the TMR approach through the development of the Local Grid Refinement (LGR) capability for MODFLOW – 2005. MODFLOW–LGR iteratively solves the groundwater flow equations for parent and child grids until a converged solution is obtained for all grids. Schaars and Kamps (2001) also prototyped a LGR approach for MODFLOW, but used a single matrix solution as an alternative to iteration between grids.

This paper describes implementation of a generalized control volume finite-difference (CVFD) formulation (sometimes referred to as an integrated finite-difference approach) into the MODFLOW–2005 framework. The formulation is similar to the CVFD formulation implemented in TOUGH2 (Pruess and others, 1999). The formulation is based on an unstructured grid approach, which allows users to design flexible grids that conform to aquifer boundaries and can be refined in areas of interest. This new program is called MODFLOW–USG, to denote a version of MODFLOW that supports UnStructured Grids. The approach implemented in MODFLOW–USG provides an alternative to other MODFLOW approaches for fitting irregularly shaped boundaries and adding targeted resolution to areas of interest. In contrast to the nested grid approaches of TMR and LGR in which individual parent and child grids are linked or coupled in an iterative manner, MODFLOW–USG simulates groundwater flow on all simple and nested grid connections using a fully

1 "Politehnica" University of Timisoara, Faculty of Civil Engineering, Department of Hydrotecnnics, Timisoara, Str. George Enescu, no. 1/A, Timisoara, Romania, www.hidro.upt.ro

implicit solution. A MODFLOW–USG grid can be a typical MODFLOW rectangular finite-difference grid, a combination of an arbitrary number of nested rectangular grids, or a grid composed of triangles, hexagons, irregular shapes, or combinations of these. For nested rectangular grids, the MODFLOW–USG approach is similar to the approach developed by Schaars and Kamps (2001). A key advantage of the MODFLOW–USG design is that groundwater flow over the entire grid is solved using a single matrix solution. For many complex groundwater problems, this approach will require less iteration for convergence than refinement approaches that iterate between grids. This approach also makes it easier to support packages that may move water between different grid nesting levels, such as the Stream Flow Routing (SFR) Package, for example.

The finite-difference formulation implemented in the Groundwater Flow (GWF) Process of MODFLOW generates a set of matrix equations that have a fixed pattern of nonzero entries. For example, a three-dimensional structured rectangular grid generates a 7-point connection pattern that includes a cell and its back and front neighbors in the three principal directions. An unstructured approach, however, generates nonzero matrix connections that are not based on a fixed pattern. An unstructured approach allows for an arbitrary number of connections between cells. In addition, the matrix can be expanded to include other flow processes. For example, this report also describes the Connected Linear Network (CLN) Process, which solves for flow through a connected linear network that may represent karst solution conduits, underground excavations (tunnels), agricultural tile drainage, or wells. For simulations that combine the CLN and GWF Processes, the unstructured matrix approach simultaneously solves for flow within the linear network, within the aquifer, and between the linear network and the aquifer in a single matrix. Furthermore, the linear sparse matrix solvers implemented in MODFLOW–USG can solve matrix equations that have an asymmetric conductance matrix. The option to handle asymmetric matrices is used here to incorporate the Newton-Raphson formulation developed by Niswonger and others (2011), as well as a fully implicit Ghost Node Correction (GNC) Package.

II. MODFLOW–USG OVERVIEW

Most components of MODFLOW–USG are functionally similar to MODFLOW–2005. Both programs use stress periods and time steps for temporal discretization. Like MODFLOW–2005, MODFLOW–USG runs from the command line and reads a name file containing a list of active packages, processes, and input and output data files. MODFLOW–USG is capable of simulating an existing MODFLOW–2005 dataset of supported packages provided the original MODFLOW–2005 solver package is replaced with the Sparse Matrix Solver (SMS) Package described herein. The ability to use cell shapes other than rectangles is another important difference between MODFLOW and MODFLOW–USG. Although cells can be variably shaped in the horizontal direction, MODFLOW–USG requires that cells are prismatic in the vertical direction. Cells can also be grouped into layers for easier processing, and sublayering can be used to further divide cells. To facilitate this new flexibility in grid design, MODFLOW–USG identifies cells by node number when used with an unstructured grid dataset, instead of by layer, row, and column, as is done in MODFLOW. There is an obvious difference between indexing cells on the basis of node numbers and indexing cells on the basis of layer, row, and column; however, the concept of applying hydrologic stresses, using boundary packages, to individual cells remains consistent with MODFLOW concepts, even if the model cells have a nonrectangular shape. Another substantial difference is the way in which the connectivity between cells is represented. In MODFLOW, there is no need to specify connection information, because each cell is logically connected to the six surrounding cells in the principal directions, and connected cells are easily determined from the layer, row, and column indices. With the unstructured-grid option in MODFLOW–USG, users must define this connectivity. Users provide connectivity information to MODFLOW–USG in the form of two arrays; the first array contains the number of connections for each cell, and the second array contains a list of the connected node numbers for each cell. Application of graphical user interfaces (GUIs) greatly simplifies input for unstructured grids by internally generating this connectivity information.

A Newton-Raphson formulation was recently developed for MODFLOW–2005 by Niswonger and others (2011). This formulation eliminates inactivation of dry cells and the abrupt reactivation of rewetted cells, which can cause model convergence problems. MODFLOW–USG contains this Newton-Raphson formulation to help resolve nonlinearities associated with wetting and drying of model grid cells as well as those nonlinearities introduced by some boundary packages and processes.

The matrix solvers distributed with MODFLOW–2005 were specifically developed for a structured grid in which each cell is connected to the six adjacent cells, and the coefficients in the matrix are symmetric about the main diagonal. These matrix solvers cannot be used with an unstructured grid, and are therefore not included in MODFLOW–USG. MODFLOW–USG contains several flexible matrix solvers that can be used with an unstructured grid. These solvers are packaged into the SMS Package and include an asymmetric sparse matrix solver called χMD (Ibaraki, 2005) and an unstructured preconditioned conjugate gradient (PCG) solver developed by White and Hughes (2011) for symmetric equations. Solution of the groundwater flow equation is managed in MODFLOW–USG by the SMS Package. The SMS Package manages the outer (nonlinear) iteration loop by use of either

Picard iteration or the Newton-Raphson formulation described by Niswonger and others (2011) and implements under-relaxation and residual control measures as required. The SMS Package also invokes the selected linear matrix solver. The unstructured matrix solvers managed by the SMS Package provide the foundation for many of the capabilities provided by MODFLOW–USG.

The CLN Process is also included with MODFLOW–USG. One-dimensional features simulated with the CLN Process can represent wells, a network of tile drains, or any other network of tubular conduits (for example, karst conduits and underground excavations) that is present within an aquifer. This new package replicates some of the capabilities of the Conduit Flow Process (Shoemaker and others, 2007) and the Multi-Node Well (MNW) Packages (Halford and Hanson, 2002; Konikow and others, 2009), which are not included with the current version of MODFLOW–USG. The CLN nodes are implemented into the simulation in a fully implicit manner and solved in the same matrix as the groundwater flow equation to improve convergence properties. The fully implicit and tight coupling of CLN nodes to aquifer cells is possible because of the unstructured design of the matrix solvers provided with MODFLOW–USG. The formulation and implementation of the CLN Process is also detailed herein.

For accurate flux calculations, the CVFD method requires certain geometrical properties pertaining to the cell connections. For most grid types, this means that the line connecting the centers of two cells should bisect the shared edge at right angles. This CVFD requirement is violated for irregular polygon cell geometries or nested grids, thus introducing errors in simulated flows and heads. The larger the deviation is from this CVFD requirement, the larger the error. MODFLOW–USG includes a GNC Package for reducing such errors. The GNC Package is optional because no correction is needed for simple grid (as opposed to nested grid) connections of regular polygon, equilateral triangle or rectangular shaped cells. In addition, the package may not be needed for many grids even when they violate these geometric cell properties. But for certain grid types and flow patterns, the corrections may be required to ensure an accurate solution. The GNC Package was developed in an implicit manner such that the corrections are part of the matrix solution; however, options for updating the GNC terms on the right-hand-side vector are also included so that the symmetric linear solvers available with MODFLOW–USG can be used. A simple test problem using a nested grid is presented herein to demonstrate the effectiveness of the GNC Package.

III. COUPLED PROCESSES AND THE GENERALIZED CVFD FORMULATION

MODFLOW–USG provides a framework for tightly coupling multiple hydrologic processes. The tight coupling, in contrast to a sequential or iterative coupling approach, occurs through the formulation of a global conductance matrix that includes the cells for all processes. The framework allows individual MODFLOW–USG processes to add to the global conductance matrix in order to represent fluxes between cells within a process as well as with cells of other processes. The global conductance matrix can be symmetric or asymmetric and is unstructured, indicating that an individual cell may have an arbitrary number of connections with other cells. The CVFD formulation accommodates this unstructured framework of tightly coupling flow processes as well as of allowing flexibility in cell geometry and connectivity within processes.

Following is the general form of a CVFD balance equation for cell n:

$$\sum_{m\in\mu_n} C_{nm}\left(h_n-h_m\right)+HCOF_n\left(h_n\right)=RHS_n \quad (1)$$

where

C_{nm} is the inter-cell conductance between cells n and m,

h_n and h_m are the hydraulic heads at cells n and m,

$HCOF_n$ is the sum of all terms that are coefficients of hn in the balance equation for cell n, and

RHS_n is the right-hand-side value of the balance equation.

Note that the summation of the first term is over all cells m that are an element of η_n, the set of cells that are connected to cell n. C_{nm} is a constant in some cases (for flow between two cells in a confined aquifer, for example) but is often dependent on the values of hn and hm (for flow between two cells in an unconfined aquifer, for example). Further, note that the $HCOF_n$ terms result from changes in storage, and boundary fluxes that are dependent on the value of hn. Also, RHS_n contains terms related to storage and (or) boundary conditions. The first term of equation 1 expresses the volumetric flow, Q_{nm}, between two connected cells, n and m, as

$$Q_{nm}=C_{nm}\left(h_m-h_n\right) \quad (2)$$

Equation 1 is expressed in matrix form as

$$\mathbf{Ah}=\mathbf{b}, \quad (3)$$

where in MODFLOW–USG,

A is the global conductance matrix,

h is the vector of hydraulic heads, and

b is the right-hand-side vector.

The diagonal terms of A (where n = m) correspond to the HCOF vector minus the sum of the off-diagonal conductances. For confined cases, equation 3 is linear and can be solved for h, the distribution of heads, at any given time step or stress period. For unconfined cases, equation 3 is nonlinear whereby one or more of the coefficients in the conductance matrix are functions of hydraulic head. In that case, an iterative Picard solution approach repeatedly solves equation 3 until a specified level of convergence is met. For each Picard iteration, the global conductance matrix is reformulated using heads from the previous iteration. An optional Newton-Raphson approach in MODFLOW–USG can be used to accelerate and improve the convergence of unconfined groundwater simulations and other nonlinear problems.

This paper the tight coupling of the GWF and CLN Processes within MODFLOW–USG; however, the program supports addition of new processes that can be coupled with GWF, CLN, or other processes that may be implemented in the future. MODFLOW–2005 and its predecessors provide a framework for adding packages and processes that interact with GWF. In the MODFLOW–2005 framework, packages interact with the GWF Process primarily as sources and sinks; for the case when the source or sink head is itself a variable, the MODFLOW–2005 framework does not support an approach for adding new cells that can be solved simultaneously with the GWF Process. Instead, the boundary variable is solved separately from, and in an iterative fashion with, the GWF Process solution. For many packages, this is not a problem; however, when hydraulic features are strongly connected to an aquifer, the iterative approach may cause oscillations in the flow solution. Consequently, the solution may not converge efficiently, and in some cases, it may not converge at all. MODFLOW–USG provides a different framework whereby the new cells can be solved simultaneously with the GWF Process. The MODFLOW–USG program also supports implementation of sources and sinks as packages, but it further extends the modular concept to the matrix level so that the packages apply to the GWF Process, the CLN process, and other processes that may be added in the future. This concept is graphically illustrated in figure 1 as a schematic of a conductance matrix for a hypothetical MODFLOW–USG simulation. The conductance matrix is square, with the number of columns and number of rows equal to the total number of cells in the problem. Cells of the GWF Process correspond to rows 1 through the number of GWF cells. Rows for the CLN Process and some other new process that might be added in the future are also shown in figure 1.

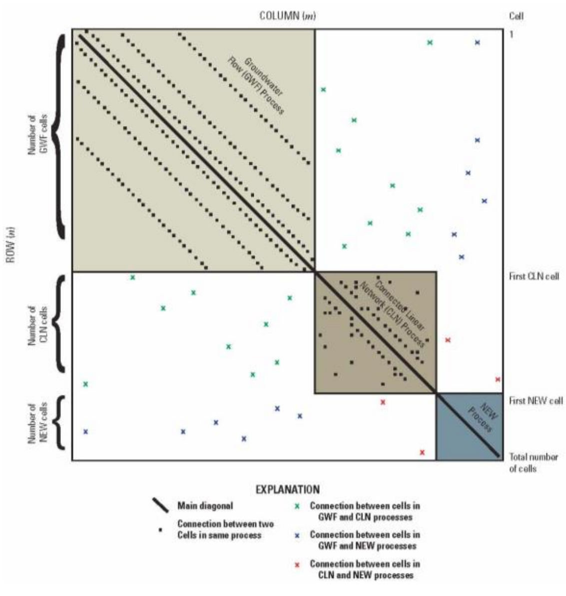

Fig. 1. Conductance matrix in MODFLOW-USG illustrating the framework for tightly coupling cells with cells from other processes.

A key component of the MODFLOW–USG approach is the ability of the linear sparse matrix solvers to handle the unstructured nature of the global conductance matrix. MODFLOW–USG takes advantage of this flexibility by providing a framework for connecting cells of different processes. As described herein, MODFLOW–USG also takes advantage of this flexibility within a process by allowing cells to be connected to an arbitrary number of neighbors. Thus, grids other than the structured grids required by MODFLOW–2005 and its predecessors can be used with MODFLOW–USG. Complicated networks of linear flow features can also be represented.

IV. UNSTRUCTURED GRID DISCRETIZATION

In MODFLOW, space is discretized in three dimensions using a rectangular finite-difference grid (fig. 2–1 of Harbaugh, 2005). The grid is created from layers, rows, and columns of cells ordered in a Cartesian coordinate system with each cell being connected to the two adjacent cells along each coordinate direction. In three dimensions, this results in a 7-point structured connectivity for the discretized set of equations. This means that a single model cell is connected, at most, to six surrounding model cells. Because the number of connections remains constant in space (except along boundaries), the grid used by MODFLOW is called a structured grid. A regular grid of hexagons is also structured, because the number of connections is the same for all cells.

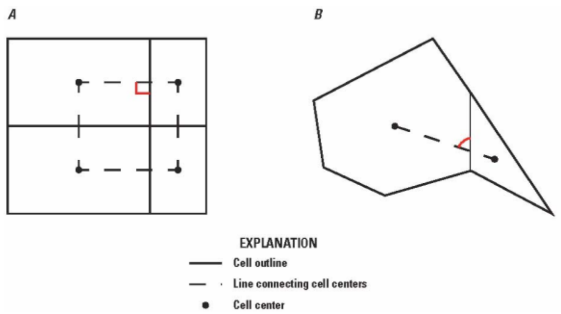

Fig. 2. Examples of two different types of cell connections: A, a line connecting the centers of adjacent cells passes through the shared face at a right angle, and B, a connecting line does not intersect the shared face at a right angle.

The term “unstructured grid” simply means that the number of connections may be variable for each cell. For example, with mesh-centered triangular finite-elements, a node may be common to several elements, and this connectivity may vary for each node. This variability results in an unstructured system of equations. Similarly, in CVFD schemes the connectivity of a cell depends on the number of shared faces, which may vary for each cell.

For accurate solutions, the standard CVFD formulation requires that a line drawn between the centers of two connected cells should intersect the shared face at a right angle (fig. 2). Furthermore, the intersection point should coincide with an appropriate mean position on the shared face (Narasimhan and Witherspoon, 1976). For grids based on a Cartesian coordinate system, this mean position will be the center of the shared face; therefore, a line connecting cell centers should bisect the shared face at a right angle. For cylindrical grids, the mean position on the shared face does not coincide with the midpoint, but rather, the logarithmic mean of the radii (Narasimhan and Witherspoon, 1976). Although this CVFD requirement is met for a simple grid of regular polygons, equilateral triangles, and rectangles, it is violated for nested grids and may be violated for grids with nonregular polygon-shaped cells. The requirement is also violated for cells with a concave shape unless cylindrical or spherical coordinates are used; thus, convex shapes should be used for most grid types (fig. 3). The smaller the deviation from this CVFD requirement, the smaller the loss of accuracy in the groundwater flow solution. In addition, the errors generally decrease as resolution increases, but they are difficult to quantify. The GNC Package, which is described herein, can be used for some grid types to improve accuracy when the CVFD requirement is violated. The possibility of violations of, and corrections to, the CVFD requirement are noted in discussions of grid types or cell geometries throughout this paper.

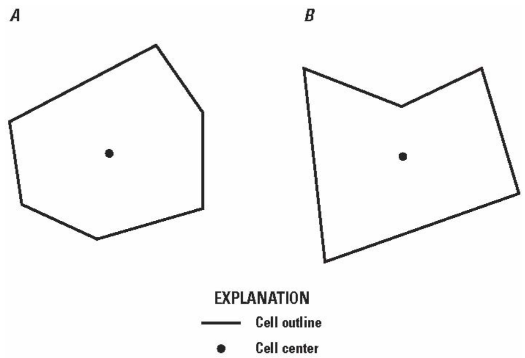

Fig. 3. Examples of A, convex and B, concave polygons. A convex polygon is one in which all interior angles are less than or equal to 180 degrees. A concave polygon has at least one interior angle that is greater than 180 degrees. MODFLOW-USG cells should have a convex shape.

The unstructured grid formulation for MODFLOW–USG is developed in a similar manner to the CVFD methodology implemented in the TOUGH2 code (Pruess and others, 1999). In TOUGH2, the domain is defined by a list of finite volumes and a list of flow connections between them. The geometric information for spatial discretization is provided in the form of a list of volumes, interface areas, and nodal distances, and there is no reference whatsoever to a global system of coordi¬nates for a particular flow problem. MODFLOW–USG also does not require information about cell shapes or how cells are positioned in space. Instead, MODFLOW–USG only requires information about connection and cell properties. This means that users can construct a wide variety of different grid types, even ones that can substantially violate the CVFD require¬ment, as may be created by using common finite-element and finite-volume mesh generation software. With this flexibility, it is incumbent upon the user to ensure that the grid used to discretize the domain is appropriate for the problem geometry and flow system, and that violation of the CVFD requirement does not introduce large errors in the flow solution, or that the appropriate correction is provided with the GNC Package—something that may not be straightforward for connections of irregular grid shapes. Otherwise, errors in the simulated results may be large, even for converged solutions with small mass-balance errors.

Figure 4 shows examples of different types of structured and unstructured grids that may be defined for the GWF Process in MODFLOW–USG. The top part of figure 4 shows structured model grids in which the number of connections is the same for all cells (except for along boundaries). For structured rectangular grids (fig. 4A), the CVFD methodology is identical to a conventional finite-difference formulation (for example, Peaceman, 1977, and Moridis and Pruess, 1992). For the unstructured grids shown in the bottom part of figure 4, the number of connections for each cell is variable throughout the grid. Unstructured grids are useful when the scale of interest or the magnitude of the hydraulic gradient varies throughout the domain. Unwarped structured grids are appealing because they do not violate the standard CVFD requirement. The radial grid (fig. 4J) also meets the CVFD requirement. The warped triangular (fig. 4F) and quadrilateral (fig. 4G) grids and the unstructured grids in figure 4 can be used with MODFLOW–USG, but it may be necessary to use the GNC Package to improve the accuracy of the flow solution.

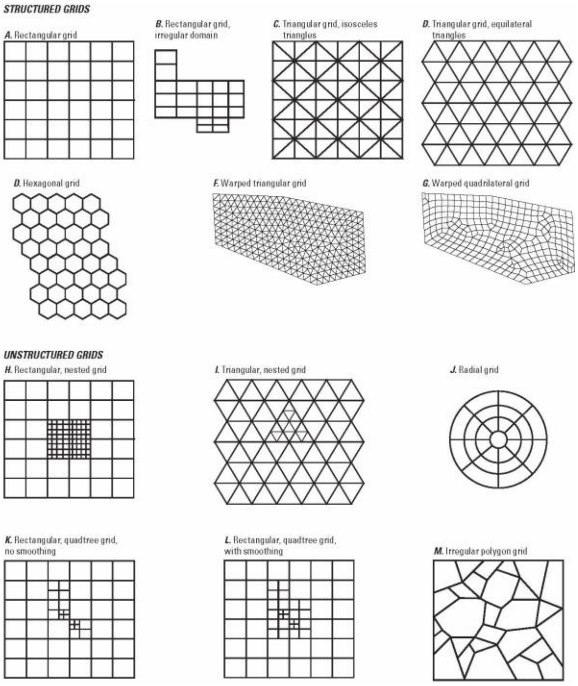

Fig. 4. Examples of different types of structured and unstructured grids.

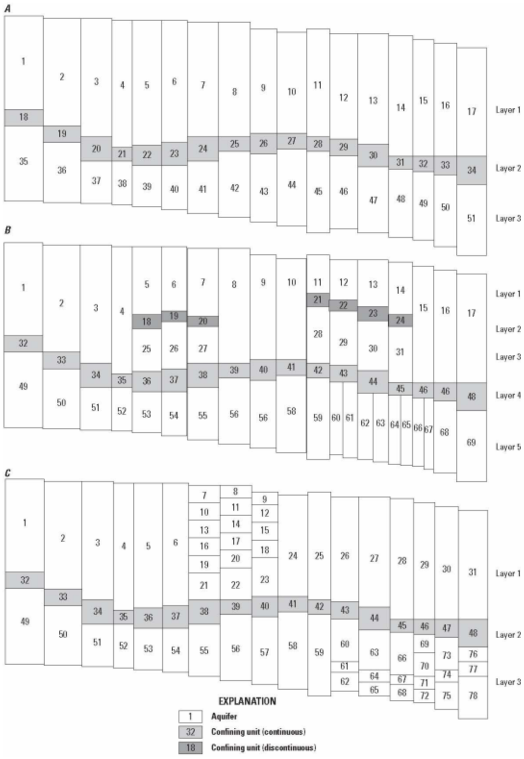

Fig. 5. Types of layering schemes that can be used with MODFLOW-USG: A, the grid configuration is the same for all layers; B, the grid configuration is different for different layers, and C, one or more layers contain vertical subdiscretization.

MODFLOW–USG requires that top and bottom cell faces are horizontal and that side faces are vertical; therefore, cells are prismatic in the vertical direction. The vertices defining the top cell face must have the same x and y coordinates as the vertices defining the bottom face of the cell. Cell tops and bottoms are horizontal and flat so that the transition between unconfined and confined conditions is handled the same way as it is handled in MODFLOW and MODFLOW–NWT. For convertible layers, when the water table is above the cell top, cell transmissivity is a function of cell thickness, whereas when the water table is below the cell top, transmissivity is a function of the cell saturated thickness. Conversion between unconfined and confined storage properties is also dependent on the cell head in relation to the cell top, as it is in MODFLOW.

MODFLOW–USG uses the concept of layers to facilitate easier pre- and post-processing, and the approach is flexible in that the number of cells can differ between layers. Alternatively, a three-dimensional, multi-layer grid can be input as a single layer to MODFLOW–USG, in which case, additional pre- and post-processing may be required to create the model input and analyze the results. If the layer concept is used for a simulation, cells need to be labeled consecutively within a grid from the top layer downward. Therefore, the lowest-numbered cells must be at the top of the grid, and cell numbers must increase downward. This numbering structure is used internally by MODFLOW–USG to identify the downward direction for cell connections. In MODFLOW–USG layers can also be subdiscretized in the vertical dimension. This capability can be useful for adding vertical resolution near partially penetrating wells, for example.

Several different layering schemes are shown in figure 5 for a hypothetical aquifer system in which an upper aquifer is separated from a lower aquifer by a confining unit. In the simplest scheme, the grid configuration is the same for both aquifers and the confining unit (fig. 5A). In plan view, the grid can be unstructured, but the same horizontal grid is used for all layers. MODFLOW–USG will also accept a grid in which a different configuration is used for each layer. This approach can be useful, for example, if the upper aquifer contained a discontinuous confining unit as shown in figure 5B. In this configuration, the upper aquifer is represented as three layers. The cells marked 18 to 24 correspond to a confining unit and are assigned to layer 2. The cells marked 25 to 31 correspond to areas in the upper aquifer that are beneath the discontinuous confining unit. Figure 5C illustrates the vertical subdiscretization concept in MODFLOW–USG. In this configuration, additional vertical resolution was added within the upper and lower aquifer. When vertical subdiscretization is used, cells within a layer do not have to be numbered from top to bottom. Instead, MODFLOW–USG requires an additional input array that indicates whether a connection between two cells is vertical or not. Also, a larger node number should reside below a smaller node number to identify the downward direction.

REFERENCES

[1] Dickinson, J.E., James, S.C., Mehl, S., Hill, M.C., Leake, S.A., Zyvoloski, G.A., Faunt, C.C., Eddebbarh, A., 2007, A new ghost-node method for linking different models and initial investigations

of heterogeneity and nonmatching grids: Advances in Water Resources, v. 30, p. 1722–1736

[2] Halford, K.J., and Hanson, R.T., 2002, User guide for the drawdown- limited, multi-node well (MNW) package for the U.S. Geological Survey's modular three-dimensional finite-difference ground-water flow model, versions MODFLOW–96 and MODFLOW–2000: U.S. Geological Survey Open-File Report 02–293, 33 p.

[3] Harbaugh, A.W., 2005, MODFLOW–2005, the U.S. Geological Survey modular ground-water model—The Ground-Water Flow Process: U.S. Geological Survey Techniques and Methods, book 6, chap. A16, variously paged.

[4] Harbaugh, A.W., Banta, E.R., Hill, M.C., and McDonald, M.G., 2000, MODFLOW–2000, the U.S. Geological Survey modular ground-water model—User guide to modularization concepts and the Ground-Water Flow Process: U.S. Geological Survey Open-File Report 00–92, 121 p.

[5] Harbaugh, A.W., and McDonald, M.G., 1996, User's documentation for MODFLOW–96, an update to the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 96–485, 56 p

[6] Mehl, S.W., and Hill, M.C., 2005, MODFLOW–2005, The U.S. Geological Survey modular ground- water model—Documentation of shared node Local Grid Refinement (LGR) and the Boundary Flow and Head (BFH) Package: U.S. Geological Survey Techniques and Methods, book 6, chap. A12.

[7] Niswonger, R.G., Panday, Sorab, and Ibaraki, Motomu, 2011, MODFLOW–NWT, A Newton formulation for MOD-FLOW–2005: U.S. Geological Survey Techniques and Methods, book 6, chap. A37, 44 p.

[8] White, J.T., and Hughes, J.D., 2011, An unstructured GPGPU preconditioned conjugate gradient solver for MODFLOW 2005, MODFLOW and More 2011, June 5–8, 2011, Golden, Colorado.