Procedures and Computer Programs for Telescopic Mesh Refinement Using MODFLOW

U.S. GEOLOGICAL SURVEY

Open-File Report 99—238

Procedures and Computer Programs for Telescopic Mesh Refinement Using MODFLOW

By S.A. Leake *and* David V. Claar

Open-File Report 99—238

Tucson, Arizona
1999

U.S. DEPARTMENT OF THE INTERIOR
BRUCE BABBITT, Secretary

U.S. GEOLOGICAL SURVEY
Charles G. Groat, Director

PREFACE

This report presents computer programs for telescopic mesh refinement using the U.S. Geological Survey (USGS) ground-water model, MODFLOW. The performance of these computer programs has been tested in models of hypothetical and actual ground-water flow systems; however, future applications of the programs could reveal errors that were not detected in the test simulations. Users are requested to notify the USGS if errors are found in the report or in the computer programs. Correspondence regarding the report or program should be sent to

U.S. Geological Survey
Water Resources Division
520 North Park Avenue, Suite 221
Tucson, Arizona 85719-5035

Although this program has been used by the USGS, no warranty, expressed or implied, is made by the USGS or the United States Government as to the accuracy and functioning of the program and related program material. Nor shall the fact of distribution constitute any such warranty, and no responsibility is assumed by the USGS in connection therewith.

CONTENTS

FIGURES

TABLE

CONVERSION FACTORS

Multiply	By	To obtain
foot (ft)	0.3048	meter
foot squared per day (ft^2/d)	0.09290	meter squared per day
cubic foot (ft^3)	0.02832	cubic meter
meter (m)	3.281	foot

Procedures and Computer Programs for Telescopic Mesh Refinement Using MODFLOW

By S.A. Leake *and* David V. Claar

Abstract

Ground-water models are commonly used to evaluate flow systems in areas that are small relative to entire aquifer systems. In many of these analyses, simulation of the entire flow system is not desirable or will not allow sufficient detail in the area of interest. The procedure of telescopic mesh refinement allows use of a small, detailed model in the area of interest by taking boundary conditions from a larger model that encompasses the model in the area of interest. Some previous studies have used telescopic mesh refinement; however, better procedures are needed in carrying out telescopic mesh refinement using the U.S. Geological Survey ground-water flow model, referred to as MODFLOW. This report presents general procedures and three computer programs for use in telescopic mesh refinement with MODFLOW. The first computer program, MODTMR, constructs MODFLOW data sets for a local or embedded model using MODFLOW data sets and simulation results from a regional or encompassing model. The second computer program, TMRDIFF, provides a means of comparing head or drawdown in the local model with head or drawdown in the corresponding area of the regional model. The third program, RIVGRID, provides a means of constructing data sets for the River Package, Drain Package, General-Head Boundary Package, and Stream Package for regional and local models using grid-independent data specifying locations of these features. RIVGRID may be needed in some applications of telescopic mesh refinement because regional-model data sets do not contain enough information on locations of head-dependent flow features to properly locate the features in local models. The program is a general utility program that can be used in constructing data sets for head-dependent flow packages for any MODFLOW model under construction.

INTRODUCTION

Telescopic mesh refinement in ground-water flow modeling is a method in which a larger encompassing model is used to define the boundary conditions and model parameters for a smaller embedded model (Townley and Wilson, 1980; Buxton and Reilly, 1986; Miller and Voss, 1987; Ward and others, 1987). The method is used for situations in which a detailed model is needed for an area that is small relative to the entire aquifer system. Example applications include analyses of well fields and contaminant plumes. If a larger ground-water flow model exists for the aquifer-system surrounding an area of interest, then a more detailed model can be constructed using information from the larger model. The procedure is called "telescopic mesh refinement" because it can be applied repeatedly to construct successively smaller embedded models (Ward and others, 1987). For consistency with previous studies, this report uses the terms

"regional model" and "local model" to refer to a larger encompassing model and smaller embedded model, respectively. The larger model, however, need not be truly "regional" in areal extent and the smaller model need not be "local" in areal extent. The only requirement with regards to areal extent is that the smaller model be encompassed by the larger model.

This report documents three programs for use in Telescopic Mesh Refinement (TMR) using the U.S. Geological Survey three-dimensional finite-difference ground-water flow model commonly referred to as MODFLOW (McDonald and Harbaugh, 1988; Harbaugh and McDonald, 1996). The first program, MODTMR (MODFLOW Telescopic Mesh Refinement), will construct the data set for the perimeter boundary conditions of a local model using cell-by-cell flow terms and head values in a regional model. The "perimeter" of a local model includes not only the vertical faces of cells on the outer edge of the local model but also the horizontal faces of cells on the upper and lower surfaces of the model. The data set constructed is the input for the Flow and Head Boundary Package (Leake and Lilly, 1997). If desired, MODTMR also will construct embedded-model data sets for most other MODFLOW packages used by the regional model.

The second program, TMRDIFF (Telescopic Mesh Refinement DIFFerence), constructs files with comparisons of computed head or drawdown in the local model with computed head or drawdown in the regional model. Output from TMRDIFF can include arrays of head or drawdown in the local model; arrays of head or drawdown in the regional model, interpolated at locations of local-model cell centers; time series of head or drawdown in the local model for selected cells; and time series of head or drawdown in the regional model, interpolated at locations of selected local-model cell centers.

The third program, RIVGRID, will construct MODFLOW input data sets for the River Package, the Drain Package, General-Head Boundary Package, and Stream Package (Prudic, 1989). To construct the MODFLOW input data sets, RIVGRID uses coordinates of paths of rivers, drains, and streams; information such as elevations, widths, bed thickness, and other properties at selected points; and the model grid location, angle of rotation, and spacing. This program is needed in the procedure of telescopic mesh refinement because information on the traces of rivers and other features are not included in the MODFLOW model data sets for these features. When a local model is constructed, it is not possible to determine the correct location of these features from the data sets in the regional model. Although RIVGRID was constructed to aid in the process of telescopic mesh refinement, it is applicable for constructing data sets for stand-alone models as well.

CONSIDERATIONS FOR TELESCOPIC MESH REFINEMENT

Approaches to Constructing Local-Model Data Sets

Two basic approaches can be used to construct data sets for local ground-water models. Approach 1 uses a TMR program such as MODTMR to construct local-model data sets from information contained in regional-model data sets. Approach 2 uses a manual-gridding process, a geographic information system (GIS), or other data-processing programs to construct local-model data sets from grid-independent data (fig. 1). For either approach, the data sets specifying boundary conditions on the perimeter of the local model must be constructed using the output (head and flow) from the regional model. The main advantages of approach 1 is that the consistency between the local and regional model can be maximized. Also, many existing regional models were constructed using manual-gridding processes. Grid-independent data sets do not exist in forms that can be processed without additional effort in importing data into GIS or other programs. In these cases, use of approach 1 with a computer program such as MODTMR will require less effort than approach 2. The main advantage of approach 2 is that the local-model data sets are most consistent with the locations and distributions of aquifer properties and sources and sinks contained in the grid-independent data sets. Location information is not contained in the regional-model data sets to properly locate point and line sources and sinks in the local-model grid. These sources and sinks include wells,
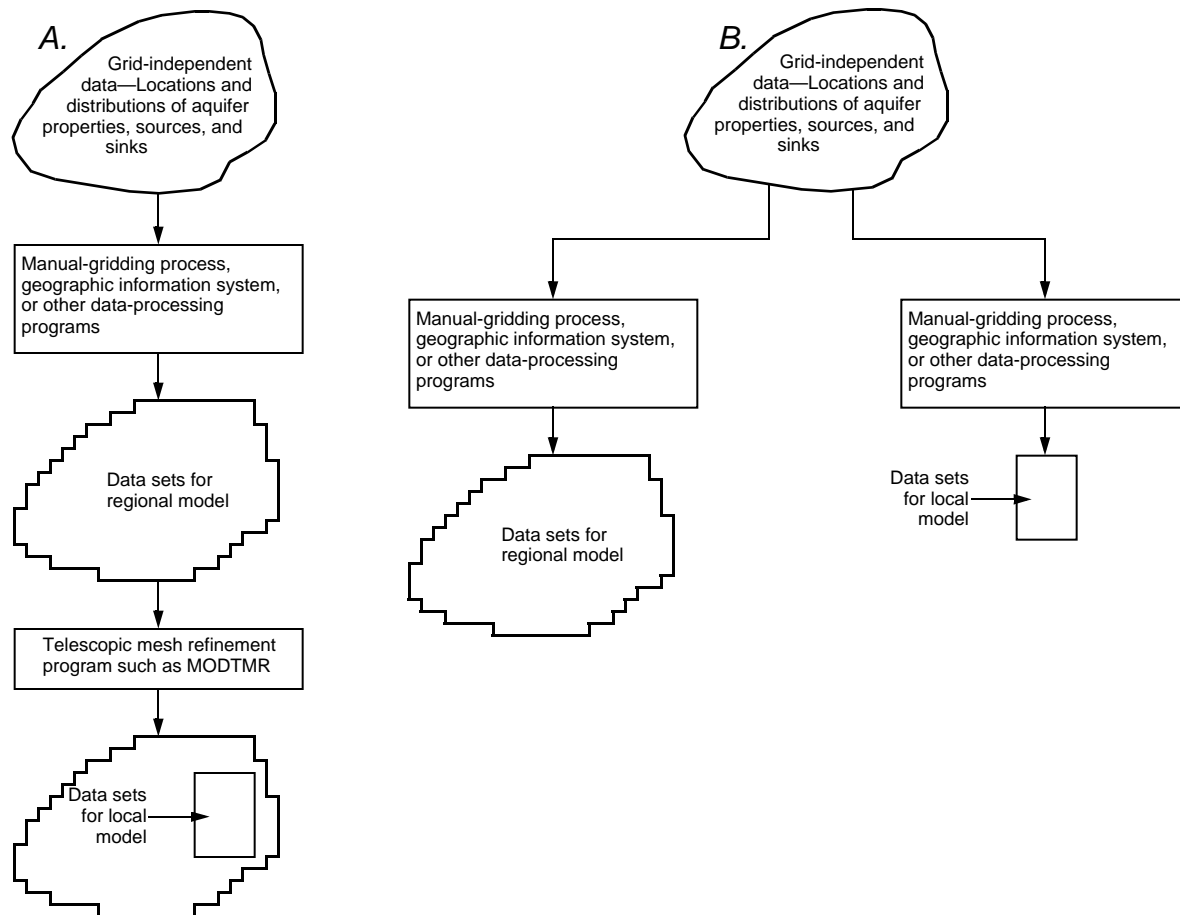
Figure 1. Approaches to constructing local- and regional-model data sets. *A*, Construct regional-model data sets from grid-independent data and local-model data sets from regional-model data sets. *B*, Construct regional- and local-model data sets from grid-independent data.

springs, rivers, streams and drains. To accurately locate these features in a local model, approach 2 is almost always better than approach 1.

Regional Model

For construction of local models using programs documented in this report, the existence of a regional model with data sets in the MODFLOW-96 format (Harbaugh and McDonald, 1996) is required. Construction of a local model requires a regional model that has a reasonable representation of the hydrologic system in the area of local interest. If model calibration is poor or validity of model results is uncertain in the area of interest, then the local model most likely will incorporate misrepresentation of the hydrologic system. A number of specific problems might render a regional model of little value in development of boundary conditions for a local model. Such problems in the area of interest in the regional model include incorrect direction of movement of ground water, gradients that are too steep or too flat, inappropriate aquifer properties, heads that are too high or too low, and inappropriate model layering. All of these problems can result in similar problems in the local model constructed from the regional-model data sets. To overcome these problems, improvement of the regional model in the area of interest may be required before constructing a local model.

Local-Model Grid Extent, Spacing, and Orientation

Ideally, the horizontal extent of a local model should be designed to encompass major nonareal sources and sinks near the local area of interest and exclude such features that are distant from the area of interest. If possible, the local-model boundary should not pass through or near points of concentrated inflow or outflow, such as wells, rivers, drains, etc. (fig. 2). The reason for this consideration is that flow and head distributions around the lateral perimeter of a local model will be most consistent with those of the regional model in the absence of points of concentrated inflow and outflow. For many regional models, however, it is difficult to define a local-model area that does not cut across rivers, canals, well fields, and other features. In these cases, the local-model perimeter should be far enough away from the area of interest in the local model so that the representation of perimeter boundary conditions using head and flow distributions in the regional model does not have much effect on results in the area of interest. Another consideration regarding grid extent is that the local model should encompass more than a few cells of the regional model. If dimensions of the area of interest are within the order of magnitude of the grid spacing of the regional model, then the local model should encompass an area larger than the area of interest. If more detail is needed in the area of interest, then another local model can be constructed within the first local model using the telescoping procedure described by Ward and others (1987).
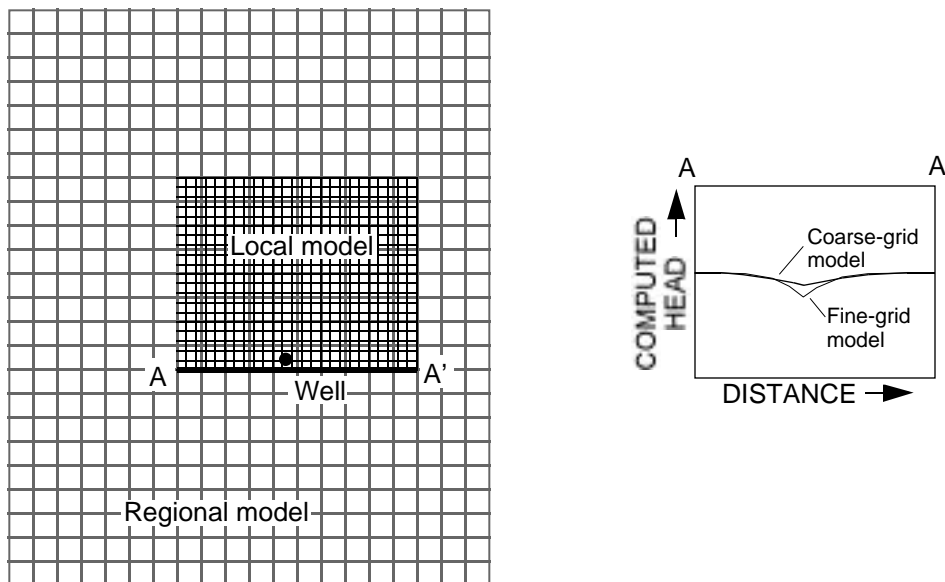
Figure 2. Example of a concentrated outflow point (a well) near the boundary of a local model. Distributions of computed head and flow in course-grid and fine-grid models are most likely to differ near areas of concentrated inflow and outflow.

MODTMR does not require that the vertical extent of local models be the same as in the corresponding areas of regional models. The vertical extent of a local model can be selected by including a subset of layers in a regional model. Constructing local models in this way requires that perimeter boundary conditions be applied on upper and (or) lower horizontal cell faces in the local model. If the local-model area includes concentrated vertical flow to or from nonareal sources and sinks, then layers adjacent to the sources and sinks probably should be included in the local model.

For most regional models, MODTMR allows the local grid to be at any angle to the regional grid. The option of allowing arbitrary rotation angles between the local- and regional-model grids allows flexibility in constructing local models to desired specifications. In regional models that include horizontal anisotropy, however, the angle between the local grid and the regional grid must be an integer multiple of 90 degrees.

Maintaining consistency in water-budget components between the local model and the corresponding area in the regional model (see section entitled, "Consistency Between Local and Regional Models") requires that components for regional-model cells be included in proportion to the fraction of cell area within the area of the local model. For example, if a regional-model cell is outside the area of the local model, then none of the budget components should be included in the local model. Similarly, if half of a cell is in the local-model area, then half of the budget components should be included in the local model. This could result in the situation where part of the pumping from a well inside the local model is excluded from the local model (Well A in fig. 3) and part of the pumping from a well outside the local model area is included in the local model (Well B in fig. 3). If these conditions are not desired, they can be avoided by making the orientation of the local grid the same as that of the regional grid and making the perimeter of the local grid coincident with grid lines (row and column boundaries) in the regional model. With this approach, regional-model cells are entirely inside or outside the local model area.
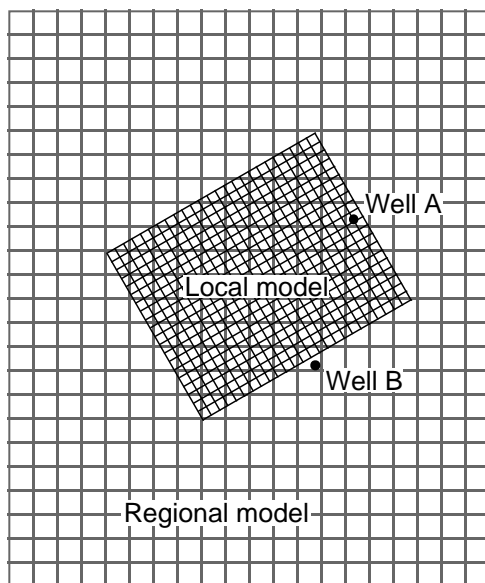
Figure 3. Example of a local-model grid that is not aligned with the regional-model grid. To make water-budget components in the local model consistent with components in the corresponding area of the regional model, fractions of the pumping rates from both wells must be included in the local model.

Consistency Between Local and Regional Models

The TMR procedure uses flow and (or) head from a regional model to define boundary conditions for a local model. Although the grid spacings are different, the models share a common set of conditions along the perimeter of the local model. The interior of the local model, therefore, should be as consistent as possible with the corresponding subarea of the regional model. The primary consideration for consistency is that the water-budget components in the local model should be the same or nearly the same as those in the corresponding area of the regional model. This consideration means that totals of each water-budget component should be the same in each model. For example, the total recharge in the local model should be the same as the recharge in the corresponding area of the regional model. Similarly, total discharge by wells should be the same in the two models. Also, parameter distributions in the local model should match distributions in the corresponding area of the regional model as closely as possible. In constructing data sets for local models, the TMR program presented in this report (MODTMR) uses interpolation methods that

maximize consistency in parameter distributions and water-budget components. For models with head-dependent sources and sinks, however, some recalibration of conductance values in the local model may be necessary to match the water-budget components for these features in the regional model.

Finer grids in local models allow more detailed representation of locations of hydrologic features than is possible in coarser-gridded regional models. Such refinement is often the primary reason for constructing local models. Consistency between local and regional models also should be maintained during this process of refining or calibrating local models. In this process, revised source/sink amounts and parameter distributions in the local model should be incorporated into the regional model. Subsequently, the regional model should be rerun and new boundary conditions should be constructed for the local model. The process of transferring detailed distributions of sources and sinks and parameters from a finer-grid local model to a coarser-grid regional model is referred to as "upscaling." Programs presented in this report do not provide a means of upscaling from local to regional models.

Types of Perimeter Boundary Conditions

Possible types of boundary conditions for perimeters of local models include specified flow, specified head, and head-dependent flow. Only the first two types have been implemented as options in MODTMR. For boundary conditions in cells along the outer local-model perimeter (fig. 4), MODTMR allows selection

Regional model

Local model

Vertical perimeter faces of local model

Horizontal perimeter faces of local model
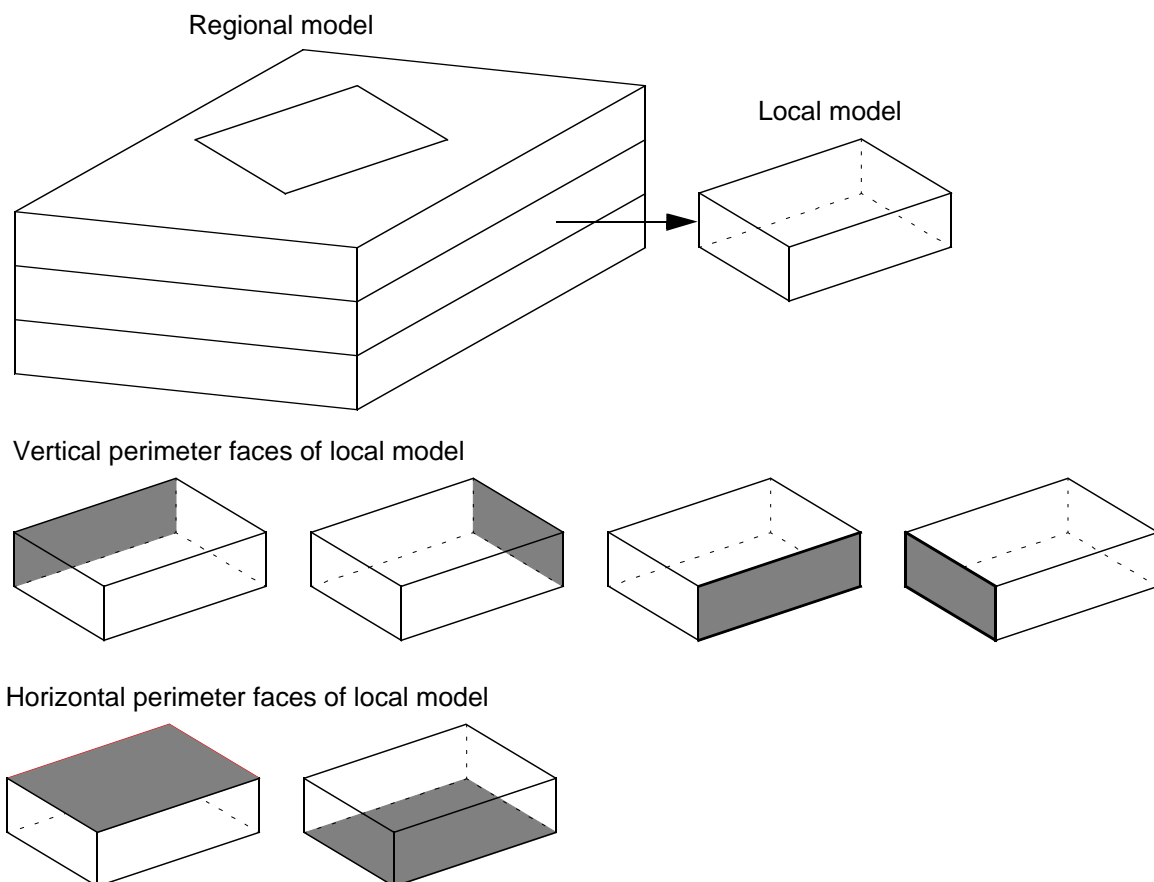
Figure 4. Perimeter faces of local models for which boundary conditions from regional models are specified.

of specified flow across the outer (vertical) cell face or specified head in the cell. For perimeter boundary conditions on upper or lower boundaries, MODTMR allows only specified flow across the outer horizontal-cell face. Where head is specified along the outer perimeter of a local model, comparisons of flow to or from specified-head cells with flow in the regional model crossing local-model specified-head locations will provide a measure of consistency between the local and regional models. Similarly, where flow is specified along the perimeter of a local model, comparisons of computed head in the local and regional models will provide a measure of consistency between the two models. For two reasons, specified-flow boundary conditions for all cells or a combination of specified-flow and specified-head conditions are preferred over specified-head conditions for all cells. First, checking head values for differences between the two models is easier than checking flow values along the local model perimeter. TMRDIFF provides a means of comparing head values in local and regional models. Second, use of specified-head conditions for all perimeter cells can result in head in the interior of the local model to be constrained and insensitive to changes in parameter values. Because head is the most common type of calibration data, overconstraining the head distribution may be a limitation in the calibration process. For some local models, use of specified-flow conditions for all cells will not allow MODFLOW to solve for a head distribution. This condition occurs in local models without specification of head or elevation surfaces in any other part of the model. In these cases, setting one of the outer perimeter cells to specified-head conditions will allow a solution. In summary, use of specified-flow conditions or a combination of specified-flow conditions and specified-head conditions is the preferred approach.

Initial Conditions in Transient-State Models

MODTMR constructs local-model data sets from regional-model data sets and computed head and flow from regional-model simulations. The common approach in construction of a transient-state model includes specifying the starting head as the ending head from a steady-state model (fig. 5). In construction of a local model, MODTMR computes local-model starting heads by interpolation of starting head in a regional model. For many local models, however, specification of starting head from the local steady-state model is a better approach. Along specified-flow perimeter cells and in interior local-model cells, the head surface commonly differs slightly from the head surface in the corresponding area in the regional model. Use of the interpolated head surface in the local model might result in some readjustment of computed head in the early part of the transient simulation. This readjustment can be eliminated by specifying starting head as the ending head from the local-model steady-state simulation.
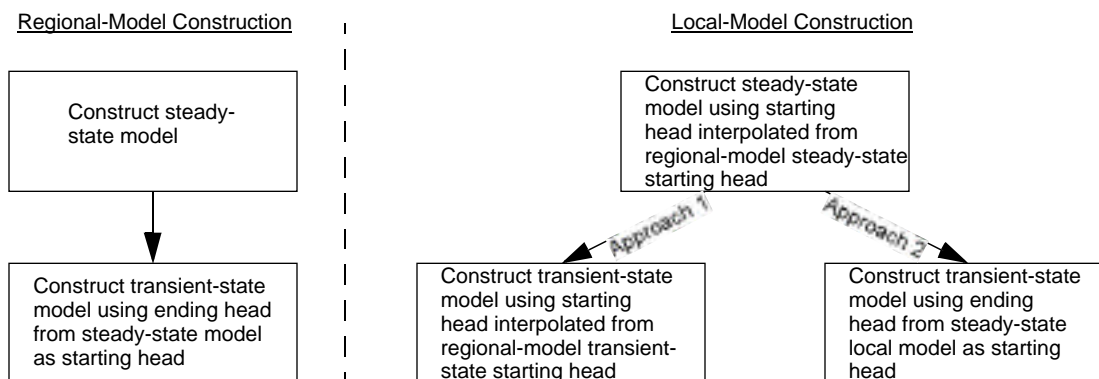
Figure 5. Common approach to specifying starting head in a regional model and alternate approaches to specifying starting head in local models.

MODTMR

The purpose of MODTMR is to construct data sets for local models embedded within regional models constructed with MODFLOW-96 (Harbaugh and McDonald, 1996). MODTMR allows construction of local-model data sets that specify perimeter boundary conditions of embedded models and local-model data sets for most MODFLOW packages. Major features of the program are as follows:

1. MODTMR is self contained and will run on a variety of computer types and operating systems.
2. Combinations of flow and head boundary conditions can be specified on the perimeters of local models.
3. Local-model grids can be at any angle to regional-model grids.
4. Local models can include a subset of layers in regional models.
5. Some regional-model layers can be subdivided into multiple layers in local models.

Requirements for running MODTMR are as follows:

- Regional-model input data sets compatible with MODFLOW-96 with no Fortran unit numbers in the range of 90–99 and including a regional-model Output Control data set that specifies "COMPACT BUDGET FILES",
- Unformatted flow and head terms from steady-state or steady-state and transient-state regional-model simulations,
- An executable version of MODTMR that was compiled with a Fortran compiler compatible with that used to compile the executable version of MODFLOW-96 used to run the regional model, and
- An input data set that includes information on the regional and local models and specification of how the local model is to be constructed.

Conceptualization of MODTMR

MODTMR carries out construction of local-model data sets in two steps. The first step is to construct local-model perimeter boundary condition data sets, and the second step is to construct all other local-model MODFLOW data sets. The program always carries out the first step; however, the second step is optional. Some users may choose to construct local-model data sets using a GIS or a preprocessing program for MODFLOW.

Coordinate Systems and Time Units

Construction of data sets for a local model requires the use of three cartesian-coordinate systems (fig. 6). The origin, orientation, and units of length may be different for all three systems. The first system is a real-world or global-coordinate system. The real-world system commonly is a cartesian-coordinate system based on map projection such as Universal Transverse Mercator; however, it can be any cartesian-coordinate system. The second and third systems define coordinates within the regional and local models, respectively. The origins for these systems are the lower-left corner of each model grid, at the intersections of lines along the outer edges of the last model row and the first column. For both of these systems, positive x coordinates increase along rows in the direction of increasing column number and positive y coordinates increase along columns in the direction of decreasing row number.

Relations between coordinates for regional and local models are determined by the relation of each to the real-world coordinate system. A scale factor, f, can be computed to convert a length in the local model, l_2, to a length in the regional model, l_1, as $l_1=f \times l_2$. If scale factor f_1 converts units of length in the real-world coordinate system to units of length in the regional model, and scale factor f_2 converts units of length in the real-world coordinate system to units of length in the local model, then scale factor f can be computed as $f=f_1/f_2$. If the angle between the x axis in the real-world coordinate system and the x axis in the regional-
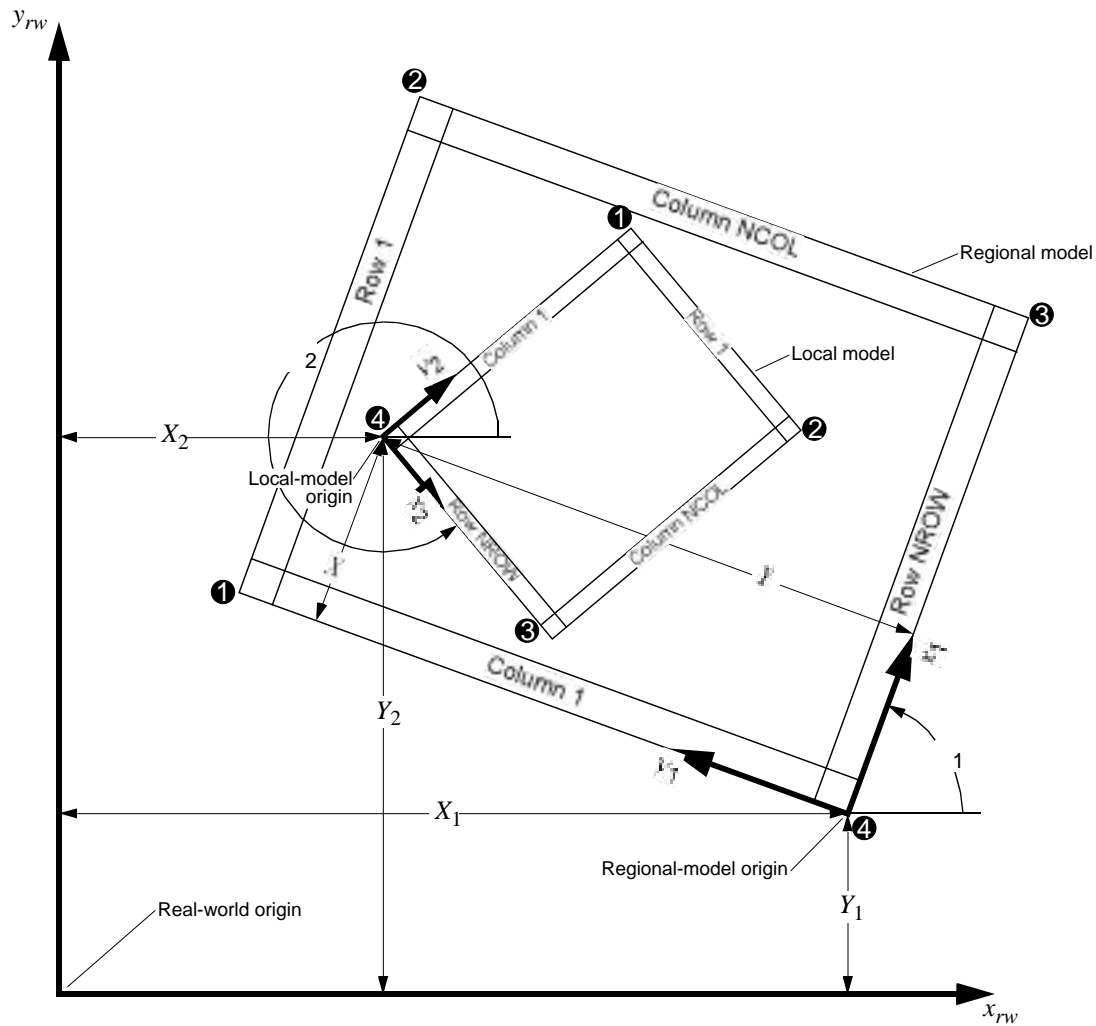
EXPLANATION

x_{rw}, y_{rw} Real-world cartesian coordinates

x_1, y_1 Regional-model cartesian coordinates

x_2, y_2 Local-model cartesian coordinates

X_1, Y_1 Distances in real-world x and y directions from real-world origin to regional-model origin

X_2, Y_2 Distances in real-world x and y directions from real-world origin to local-model origin

X, Y Distances in regional-model x and y directions from regional-model origin to local-model origin

1   Counter-clockwise angle between real-world x axis and regional-model x axis

2   Counter-clockwise angle between real-world x axis and local-model x axis

❹   Number of corner of regional- and local-model grid

Figure 6. Coordinate systems used in programs for telescopic mesh refinement.

model coordinate system is θ_1, and the angle between the x axis in the real-world coordinate system and the x axis in the local-model coordinate system is θ_2, then the angle between the x axis in the regional-model coordinate system and the x axis in the local-model coordinate system, θ, can be computed as $\theta = \theta_2 - \theta_1$. The other two parameters needed in determining the relations of coordinates in the regional model and local model are the distance in the regional-model x direction from the regional-model origin to the local-model origin, and the distance in the regional-model y direction from the regional-model origin to the local-model origin. These distances will be denoted as X and Y, respectively. In this discussion, distances from the origin of one system to the origin of another system will be referred to as x and y offsets, in units of the first system. Using a series of coordinate transformations, x and y offsets between the regional-model and local-model origins can be computed from other parameters, including x and y offsets from the real-world system to the regional-model system, X_1 and Y_1; and x and y offsets from the real-world system to the local-model system, X_2 and Y_2. For coordinate system a and coordinate system b with differing origins, orientations, and units of measurement, coordinates x_a,y_a in system a of a known point x_b,y_b in coordinate system b can be computed as

$$x_a = fx_b \cos\theta_b - fy_b \sin\theta_b + X_b ; \text{ and} \tag{1}$$

$$y_a = fx_b \sin\theta_b + fy_b \cos\theta_b + Y_b ; \tag{2}$$

where f is a scale factor to convert units of length in system b to units of length in system a; θ_b is the angle between the x axis in coordinate system a and the x axis in coordinate system b, measured in a counter-clockwise direction from the axis with system a to the axis with system b; X_b and Y_b are x and y offsets from system a to system b. Similarly, coordinates x_b,y_b in system b of a known point x_a,y_a in coordinate system a can be computed as:

$$x_b = ((x_a - X_b)\cos\theta_b + (y_a - Y_b)\sin\theta_b)/f ; \text{ and} \tag{3}$$

$$y_b = (-(x_a - X_b)\sin\theta_b + (y_a - Y_b)\cos\theta_b)/f . \tag{4}$$

Offsets X and Y from the regional-model system to the local-model system (fig. 6) can be computed by applying equations 3 and 4 with $x_b=X$, $y_b=Y$, $x_a=X_2$, $y_a=Y_2$, $X_b=X_1$, $Y_b=Y_1$, and $\theta_b=\theta_1$. In this case, f is a multiplication factor that converts regional-model distance units to real-world distance units. Lengths X, Y, X_2, Y_2, X_1, Y_1, and angle θ_1 are shown in figure 6. Input to MODTMR includes variables equivalent to f_1, f_2, X_1, Y_1, X_2, Y_2, θ_1, and θ_2, which are used to compute variables equivalent to X, Y, f, and θ.

MODFLOW (Harbaugh and McDonald, 1996) allows selection of time units for models through input variable ITMUNI. Possible values for ITMUNI and corresponding time units are:

Value	Time Unit
0	Undefined
1	Seconds
2	Minutes
3	Hours
4	Days
5	Years

MODTMR requires specification of the ITMUNI variable for both the regional and local models; however, use of a value of 0 (for undefined time units) is not allowed for either the regional or local models. MODTMR computes a conversion factor from an array or table that specifies the number of seconds in each of the possible time units of the model. Values of the table are:

Time unit	Number of seconds in time unit
Seconds	1
Minutes	60
Hours	3,600
Days	86,400
Years	31,557,600

If the number of seconds in one time unit of the regional model is designated as S_1, and the number of seconds in one time unit of the local model is designated as S_2, then a time conversion factor, f_t, can be computed as $f_t = S_2/S_1$.

Factor f_t is used in MODTMR in combination with f to convert units of parameters in the regional model to units in the local model. For MODFLOW parameters that include a multiplier, the conversion of length and time units in the local-model data sets is carried out through selection of the appropriate multiplier for the local-model data sets in combination with parameter values written in units of the regional model. For other parameters, the values written to the local-model data sets are in the length and time units of the local model.

Methods of Interpolation

A major part of construction of local-model data sets includes assigning properties and conditions to cells in the local-model grid. This process should be carried out in such a way that consistency between the local and regional models is maintained. Leake (1998) described the best methods of assigning parameters.

Method 1—Value from Single Regional-Model Cell

The simplest method of assigning values to a local-model grid is to take the value from regional-model cells in which the local-model cell center is located (fig. 7). In this report, the method is referred to as Method 1, or the "cell-value method." Use of the cell-value method produces parameter arrays that are consistent with areal distributions of parameters in the regional-model arrays, but the method is not good for representing smooth variations in original grid-independent data sets. Application of this method results in local-model data arrays in which groups of adjacent cells in the local model have the same value, each in a group taking on the value from the same regional-model cell. For example, if the method was used to compute an elevation surface such as starting head or bottom elevation, the resulting distribution in the local model will have a "stepped" appearance—each step having the elevation from one regional-model cell (fig. 8). Use of the cell-value method for bottom surfaces can result in difficulties of cells going dry near the edges of the steps. For most elevation surfaces, a method that smoothly varies the parameters over the local-model area is preferred. In MODTMR, the cell-value method is used to compute integer array values such as IBOUND in the Basic Package and IRCH in the Recharge Package; and real-array values including values in all arrays in the Evapotranspiration Package, and WETDRY in the Block-Centered Flow Package.

REGIONAL-MODEL COLUMN
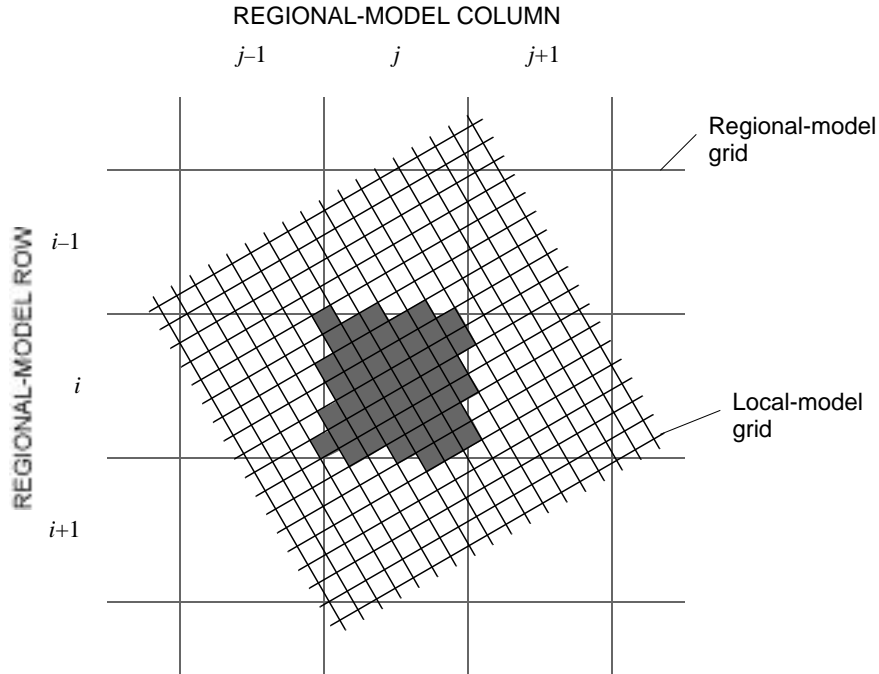
Figure 7. Method of interpolation that assigns parameters to local-model cells from values in underlying regional-model cells. Regional-model parameters from cell in row i, column j, will be assigned to the group of shaded local model cells, because the centers of these local-model cells are within that regional-model cell.

Method 2—Area-Weighted Average of Values in Multiple Regional-Model Cells

Interpolation method 2 is the same as interpolation method 1 for local-model cells that are completely within a regional-model cell. For local-model cells that overlie parts of two or more regional-model cells, however, parameter values are computed as area-weighted averages of parameters in the regional-model cells that are overlain. This method is used for certain parameters to maintain consistency between regional- and local-model flow terms (Leake, 1998). For a local-model cell that overlies n regional-model cells with parameter values $P_1, P_2, ...P_n$, a general form of an area-weighted average is

$$\bar{P} = \sum_{k=1}^{n} a_k \times P_k \bigg/ \sum_{k=1}^{n} a_k \; , \tag{5}$$

where $\bar{P}$ is the average-parameter value and $a_1, a_2, ...a_n$, are subareas of the local-model cell overlying regional-model cells with parameters $P_1, P_2, ...P_n$, respectively (fig. 9). An alternative form for averaging some quantities is

$$\bar{P} = \sum_{k=1}^{n} a_k \times P_k / A_k \; , \tag{6}$$

where $A_1, A_2, ...A_n$ are the total areas of the regional-model cells intersected.

The most complex part of applying method 2 is computing the number of cells overlain, n, and subareas $a_1, a_2, ...a_n$, for each local-model cell. MODTMR includes an algorithm for computing vertices of a polygon that has the same area as the area of intersection of any regional- and local-model cell (fig. 10). After
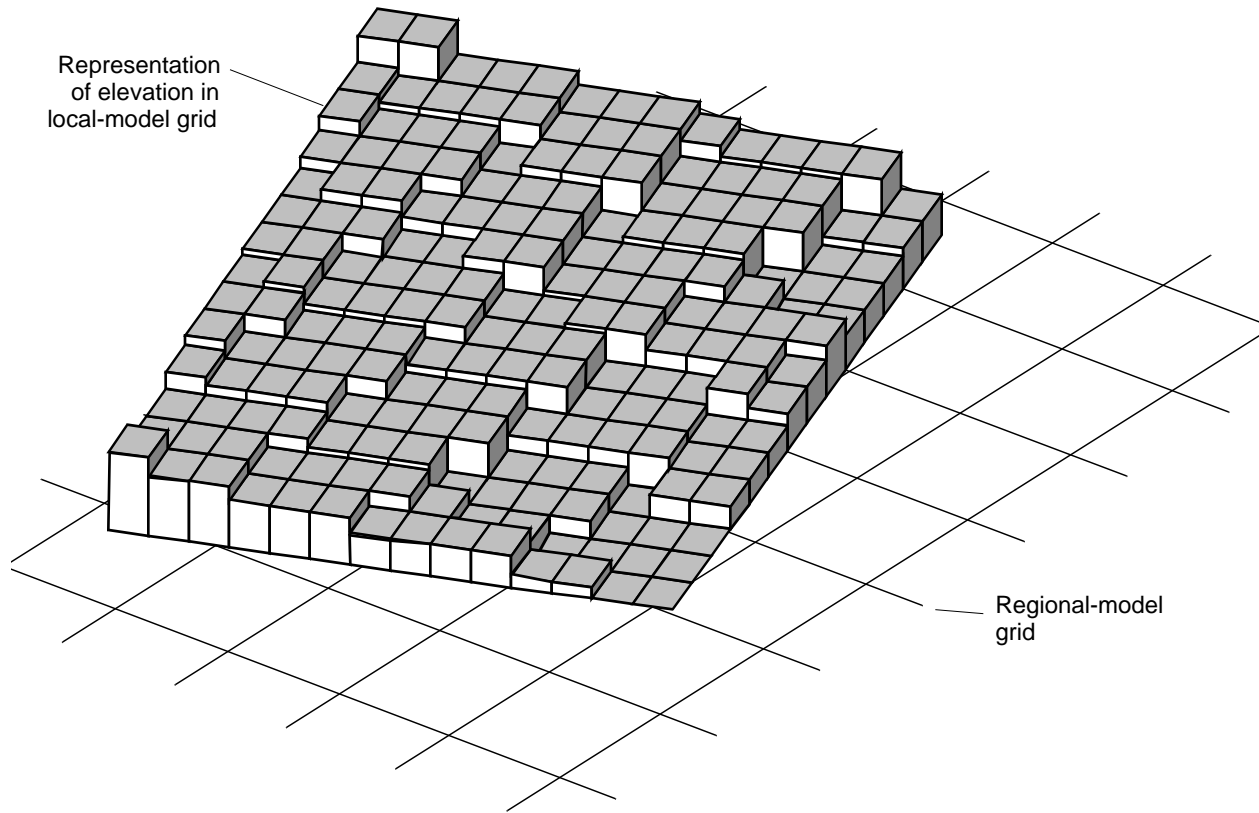
Figure 8. Representation of an elevation surface in a local-model grid computed by the cell-value method. Heights of bars represent elevation in local model cells.
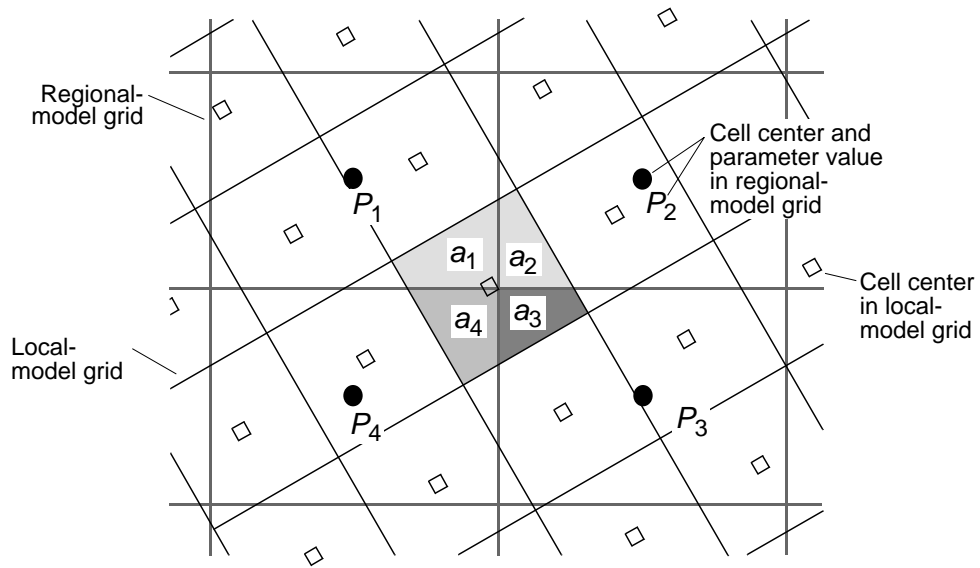
Figure 9. Part of local-model grid overlying part of regional model grid. Shaded areas a_1, a_2, a_3, and a_4 are subareas of regional-model cells intersected by a particular local model cell (Leake, 1998).

Regional-model cell

Vertex in polygon defining regional-model cell

Local-model cell

y_2

y

x

y_1

Local-model coordinate system

x_1 x_2
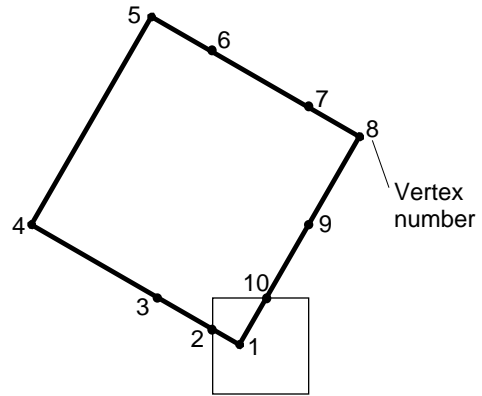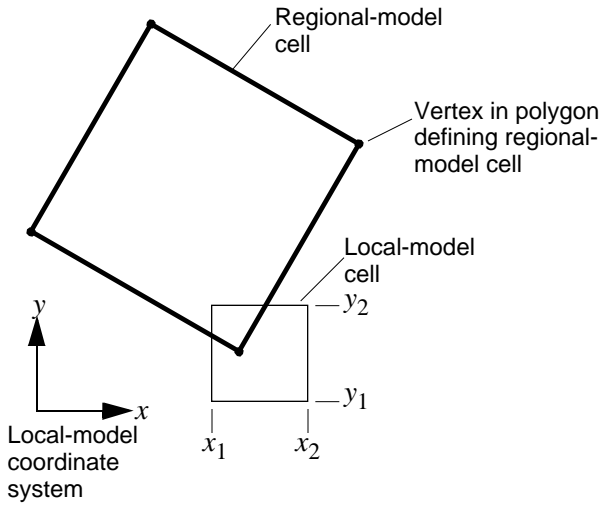
Vertex number

Step 1—Add vertices where polygon crosses $x=x_1$, $y=y_2$, $x=x_2$, and $y=y_1$. Number vertices sequentially in a clockwise direction.

x_1
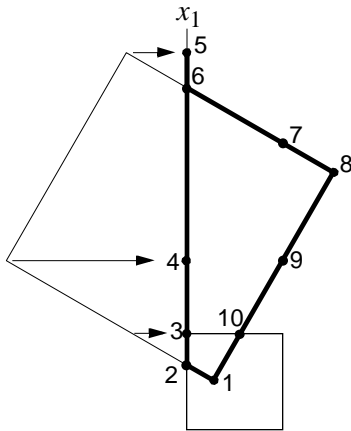
Step 2—Set any vertex x coordinates less than x_1 equal to x_1.

y_2
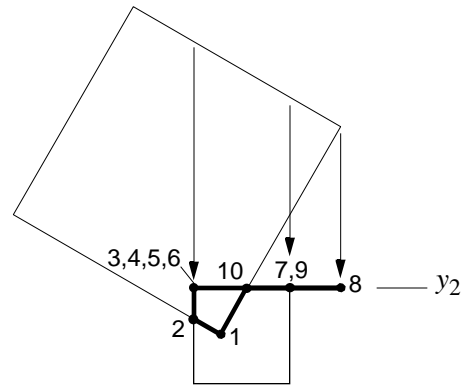
Step 3—Set any vertex y coordinates greater than y_2 equal to y_2.

x_2
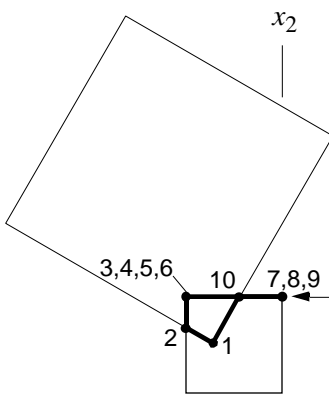
Step 4—Set any vertex x coordinates greater than x_2 equal to x_2.

y_1
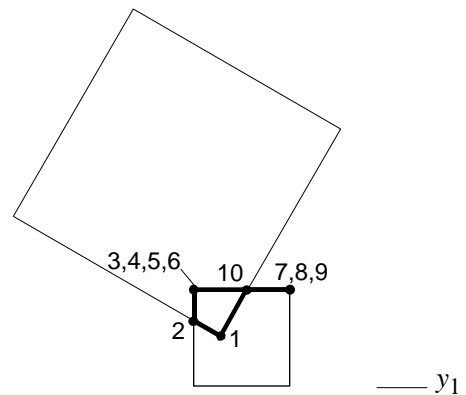
Step 4—Set any vertex y coordinates less than y_1 equal to y_1. (In this example, no coordinates moved.)

Figure 10. Procedure used for constructing a polygon for which area of intersection of a regional-model cell with a local-model cell can be computed.

MODTMR reads information on both regional- and local-model grid locations, orientations, and spacings, the following steps are carried out: (1) Compute the horizontal area of intersection of each local-model cell with each regional-model cell. This involves carrying out the procedure in figure 10 and another procedure for calculating the area of the resulting polygon. Regional-model cells that do not intersect with local-model cells have a zero area of intersection. For all local-model cells, the total number of nonzero fractional areas containing parts of regional-model cells is saved. (2) Allocate array space to store regional-model row and column locations and fractional areas for each nonzero regional-model cell fraction contained in each local-model cell. (3) Repeat step 1, this time saving row and column indices of nonzero fractional areas of each regional-model cell in each local-model cell, and the corresponding fractional area. The fractional area saved is the ratio of the area of intersection to the area of the regional-model cell, which is the ratio a_k/A_k in equation 6. With the cell indices and fractional areas saved in arrays, area-weighted averages using equation 6 can be computed directly and area-weighted averages using equation 5 can be computed by first multiplying the saved fractional areas by the area of the corresponding regional-model cell to get a_k.

Method 3—Interpolation Using Linear Combinations of Values in Surrounding or Adjacent Regional-Model Cells

Interpolation using values in surrounding or adjacent regional-model cells can be applied to calculate smoothly varying local-model parameter distributions in cases where groups of adjacent cells having the same value (fig. 8) are not desirable. Smoothly varying areal distributions generally are best for elevation surfaces such as heads in model layers and bottoms and tops of model layers. A general form for computing parameter value, $\overline{P}$, from parameters at n locations, $P_1, P_2, ...P_n$, is

$$\overline{P} = \sum_{k=1}^{n} w_k \times P_k , \tag{7}$$

where w_k is a numerical weight applied to the kth parameter value, P_k. Values of weights are selected so that $w_1 + w_2 + ... w_n = 1$. A number of methods can be used to compute the weights including kriging and inverse-distance weighting. The fact that parameter values in the regional model are in a rectangular grid, however, allows use of bilinear interpolation, which is a simple method that returns exact values of parameters at known grid points and smoothly varies parameter values between grid points. Bilinear interpolation uses known values at corners of a rectangle to compute unknown values within the rectangle. As will be discussed later, the procedure can be used to extrapolate unknown values outside the rectangle. In the example in figure 11, the corners of the rectangle are formed by cell centers in the regional-model grid. A simple procedure for computing the unknown parameter value, $\overline{P}$, is to first compute P_a between P_1 and P_4 using simple linear interpolation, next compute P_b between P_2 and P_3 using simple linear interpolation, and finally compute $\overline{P}$ between P_a and P_b using simple linear interpolation. These three steps, however, can be combined to compute the unknown parameter value as

$$\overline{P} = w_1 P_1 + w_2 P_2 + w_3 P_3 + w_4 P_4, \tag{8}$$

where $w_1 = 1 - D_x - D_y + D_x D_y$, $w_2 = D_y - D_x D_y$, $w_3 = D_x D_y$, $w_4 = D_x - D_x D_y$, with $D_x = (x_0 - x_1)/(x_2 - x_1)$ and $D_y = (y_0 - y_1)/(y_2 - y_1)$.

Bilinear interpolation can easily be applied if each local-model cell center is surrounded by four active regional-model cell centers; however, if part of the local-model grid is adjacent to the edge of the grid or an inactive area in the regional model (fig. 12), a different procedure must be used to compute the parameter value for some local-model cells. Equation 8 can be used to extrapolate a parameter value outside the area defined by four active regional-model cell centers. For a point in a regional-model cell, as many as four
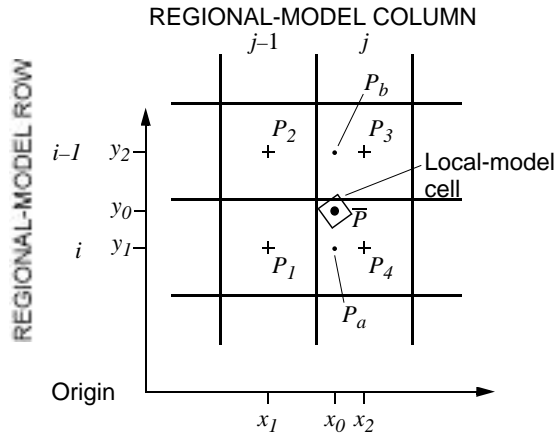
Figure 11. Coordinates used in bilinear interpolation of parameter values for the center of a local-model cell.

A.

B.
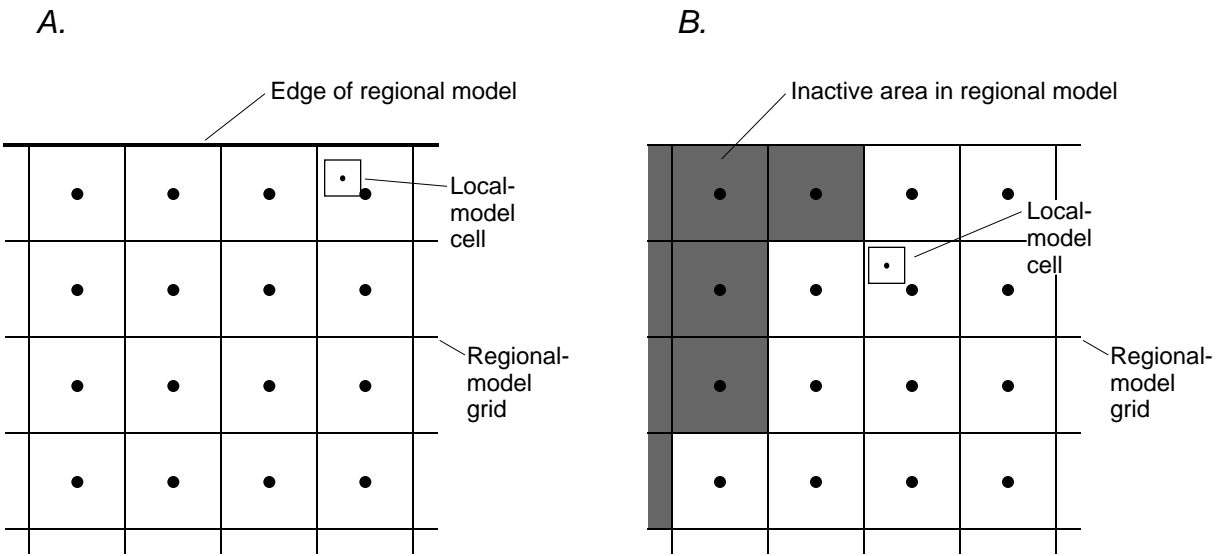
Figure 12. Examples of local-model cell locations where bilinear interpolation cannot be used directly to compute parameter values. *A*, Local-model cell center is between outermost regional-model cell centers and edge of regional-model grid. *B*, One or more regional-model cell centers surrounding local-model cell center are in inactive cells.

groups of four regional-model cell centers can be used to interpolate or extrapolate parameter values at the point (fig. 13). The first priority group of cell centers for computing the parameter is the group that interpolates rather than extrapolates the parameter value (Group 2 as shown in figure 13). If that group does not contain four active cells, the next priority is to use a group of four active cells, which includes the cell with the point of interest, to extrapolate the parameter value. The priority for determining which group of cells are used to extrapolate a parameter value is based on the distances from the point of interest to the corners of the regional-model cell containing the point of interest (fig. 14), denoted d_1, d_2, d_3, d_4. The subscript for each distance is the group number of cells (fig. 13) associated with the distance. The highest priority for selecting a group of four cells for extrapolation is the shortest distance to a cell corner associated with a group number. In the example in figure 14, the order of distances to the corners from shortest to longest distance is d_2, d_3, d_1, d_4. Group 2 has the shortest associated distance and would be the normal highest priority for use because this group interpolates at the point of interest. If group 2 does not contain four active
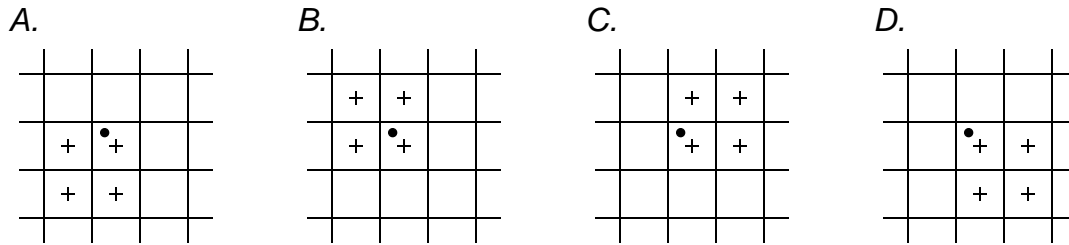
Figure 13. Possible groups of four cell centers in regional-model grid, denoted by plus symbols, for use in interpolating or extrapolating a parameter value at a point, denoted by dot. *A*, Group 1. *B*, Group 2. *C*, Group 3. *D*, Group 4.
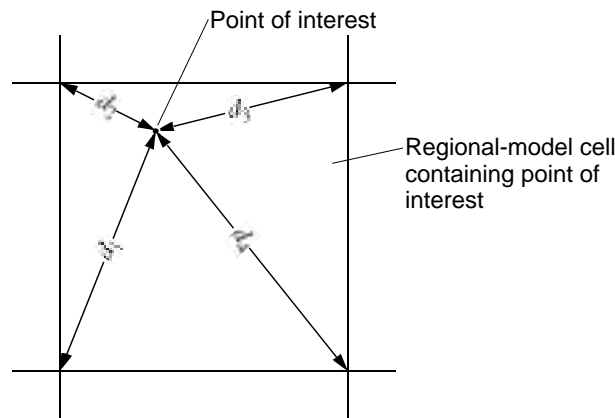
Figure 14. Distances from point of interest at which parameters will be calculated to corners of regional-model cell containing point of interest.

cells, group 3 would be the first priority group for extrapolation because d_3 is the next shortest distance. Groups 1 and 4 would follow in order of priority for use in extrapolating to the point of interest.

If none of the four groups of cells contain four active cells, the next priority is to search for three active cells for use in interpolating or extrapolating the parameter value at the point of interest. Interpolation or extrapolation using three points is based on an equation for a plane passing through the three active model-cell centers. Four groups of three cells are considered for interpolation or extrapolation (fig. 15). If more than one group of three points is available for interpolation or extrapolation, the priority for selecting the group is based on the distances in figure 14 as described previously for groups of four cells. For convenience in the computer program, the equation for interpolation or extrapolation can be put into the form of equation 8 with one of the weights set to zero. The three nonzero weights in the equation can be calculated directly from the four weights computed for bilinear interpolation. If a group of four cells includes one cell with a missing parameter value, the bilinear weight for the missing parameter value is added to each of the weights for adjacent corners of the rectangle and is subtracted from the weight for the diagonally opposite corner. The weight for the corner with the missing parameter values is then set to zero. For example, if parameter value P_2 (fig. 11) is missing, new weights w_1', w_2', w_3', and w_4' can be computed as follows: $w_1'=w_1+w_2$, $w_2'=0$, $w_3'=w_3+w_2$, and $w_4'=w_4-w_2$.

If groups of four active cells cannot be found and groups of three active cells cannot be found, the next priority is to use linear interpolation or extrapolation using two active cells. Four groups of two cells are considered for interpolation or extrapolation (fig. 16). If more than one group of two points is available for
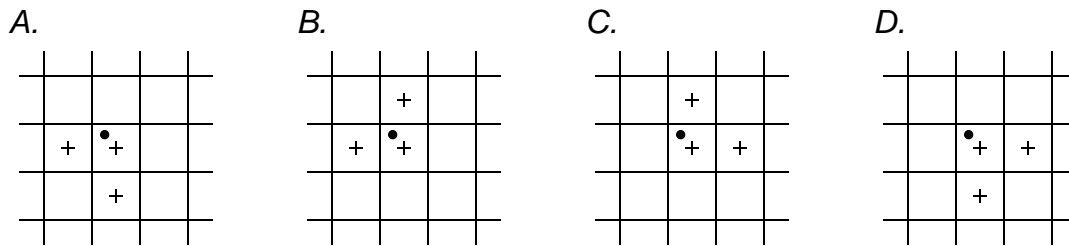
Figure 15 . Groups of three cell centers in regional-model grid, denoted by plus symbols, for use in interpolating or extrapolating a parameter value at a point, denoted by dot. *A*, Group 1. *B*, Group 2. *C*, Group 3. *D*, Group 4.
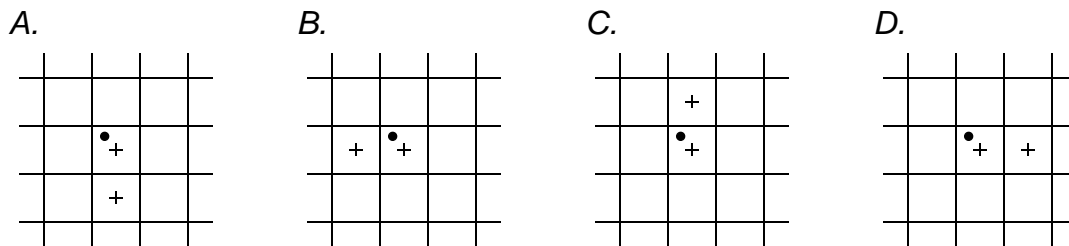
Figure 16. Groups of two cell centers in regional-model grid, denoted by plus symbols, for use in interpolating or extrapolating a parameter value at a point, denoted by dot. *A*, Group 1. *B*, Group 2. *C*, Group 3. *D*, Group 4.

interpolation or extrapolation, the priority for selecting the group is based on the distances in figure 14, which was described previously for groups of four cells. The equation for interpolation or extrapolation can be put into the form of equation 8 with two of the weights set to zero. The two nonzero weights in the equation can be calculated directly from the four weights computed for bilinear interpolation. If a group of four cells includes two adjacent cells with missing parameter values, the bilinear weight for each missing parameter value is added to the weight for the adjacent corner of the rectangle without a missing parameter. The weight for the corners with the missing parameter values are then set to zero. For example, if parameter value P_1 and P_2 (fig. 11) are missing, new weights $w_1´$, $w_2´$, $w_3´$, and $w_4´$ can be computed as follows: $w_1´=0$, $w_2´=0$, $w_3´=w_3+w_2$, and $w_4´=w_4+w_1$.

If groups of four, three, or two cells cannot be used for interpolation or extrapolation of parameter values, then the active regional-model cell containing the point of interest is isolated horizontally from adjacent cells. In this case, the parameter value for the cell of interest is taken to be the value of the parameter in the regional-model cell containing the point of interest, which is the same as application of interpolation method 1.

Computed parameter distributions adjacent to edges of regional-model grids and inactive areas are likely to be less smooth than distributions inside the grid where four points can be used for interpolation over large areas. MODTMR does not allow specification of specified-head perimeter boundary conditions in areas where head values cannot be interpolated from the four surrounding regional-model head values. If local models extend into such areas, specified-flow boundary conditions must be used in those areas. For other parameters, such as starting head and layer bottom and top elevations, computed local-parameter distributions should be checked to make sure distributions are reasonable for areas where bilinear interpolation cannot be used.

Method 4—Linear Interpolation of Cell-Wall Flow Terms

Leake and others (1998) presented a general method of computing horizontal flow across any line segment within the domain of a finite-difference model. The method presented here is slightly modified from the method of Leake and others (1998). With this method, specified flows can be determined for smaller-scale model grids that are not orthogonal to larger-scale model grids. For this discussion, an arbitrary horizontal line segment in a model domain is denoted as l, the x axis is parallel to rows, and the y axis is parallel to columns of a finite-difference grid (fig. 17). The discussion here deals with interpolation of flow rates for one time step and within one model layer using linear interpolation of rates of horizontal flow between adjacent cells for that time step and model layer. Rates of flow across cell faces are referred to as "cell-wall flow terms" in this report.
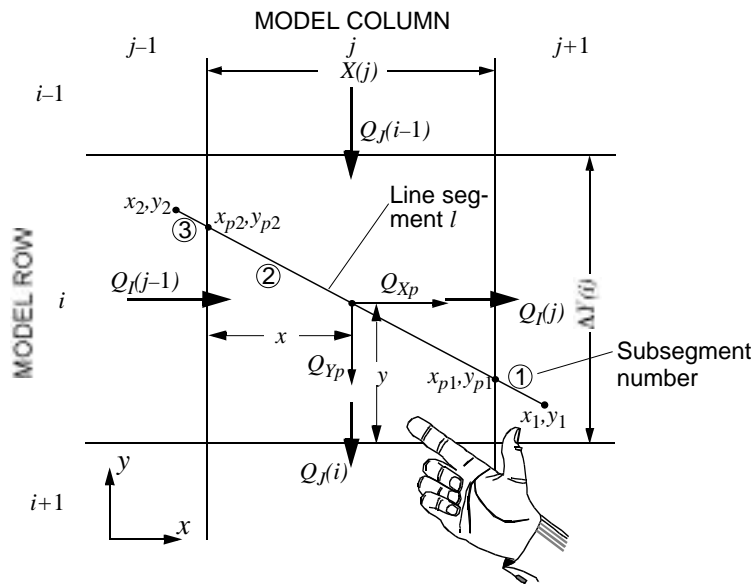
Figure 17. Variables used in formulation of equations for linear interpolation of cell-wall flow terms and convention for right-hand rule.

MODFLOW computes flow across all faces of finite-difference cells. Particle-tracking and solute-transport programs, such as MODPATH (Pollock, 1994) and MOC3D (Konikow and others, 1996), use linear interpolation of cell-wall flow terms to compute ground-water velocity anywhere in the model domain. Linear interpolation represents velocities in the x and y directions, V_x and V_y, respectively, as $V_x = f(x)$ and $V_y = f(y)$. According to Goode (1990), this approach is consistent with basic assumptions in the block-centered finite-difference modeling approach. Simple linear interpolation of flow components is used in this analysis to compute horizontal flow at any point in the model domain. Bilinear interpolation of flow components was not tested; however, Goode (1990) discusses the merits of bilinear interpolation of velocity.

Values of cell-wall flow for an individual cell in model row i and column j are denoted here as $Q_I(j–1)$ and $Q_I(j)$ for flow along rows and as $Q_J(i)$ and $Q_J(i–1)$ for flow along columns (fig. 17). These quantities are positive if water is moving in the directions of the arrows crossing the cell walls in figure 17. This analysis uses a right-hand rule in which flow crossing a line segment is positive in the direction of the extended thumb with the palm up and the index finger pointing from x_1,y_1 to x_2,y_2 as shown in figure 17.

To compute horizontal flow across line segment l, first divide l into n subsegments, where n is the number of model cells traversed by l. The lengths of the subsegments, l_1, l_2, ... l_n, are lengths of l in each model cell traversed. Next, compute flow components in the x and y directions at the center of each subsegment, and finally sum the flow components for all subsegments to compute flow for the entire length of l. For subsegment p in cell i,j with endpoints x_{p1},y_{p1} and x_{p2},y_{p2}, components of flow Q_X and Q_Y can be computed as

$$Q_{Xp} = -\left[\frac{Q_I(j-1) + \frac{x}{X(j)}[Q_I(j) - Q_I(j-1)]}{Y(i)}\right] y_p, \text{ and} \tag{9}$$

$$Q_{Yp} = -\left[\frac{Q_J(i) + \frac{y}{Y(i)}[Q_J(i-1) - Q_J(i)]}{X(j)}\right] x_p, \tag{10}$$

where $x_p = x_{p2} - x_{p1}$, $y_p = y_{p2} - y_{p1}$, and other variables are defined in figure 17. The magnitude of the component of flow normal to l is the sum of Q_X and Q_Y for all n subsegments. This result is correct for both isotropic cases (hydraulic conductivity or transmissivity along rows and columns are equal) and anisotropic cases. The procedure can be repeated for different model layers.

Perimeter Boundary Conditions

MODTMR allows use of either specified-flow or specified-head boundary conditions on the perimeter of a local model. Specified-flow and specified-head boundary conditions are computed from the grid geometries of the local and regional models and cell-wall flow terms and computed head values from simulations of the regional model. If boundary conditions are entirely specified flow, then regional-model head values are not needed. Similarly, if boundary conditions are entirely specified-head, then regional-model flow terms are not needed. If the local model is steady state, then only steady-state flow terms and head values are needed to construct the local-model boundary conditions; however, if the local model is transient, both steady-state and transient-state simulations are needed. In the latter case, information from the steady-state regional-model simulation is needed to specify the initial boundary conditions.

Boundary conditions can be specified on horizontal perimeter faces of the regional model as well as on vertical perimeter faces (fig. 18). MODTMR treats vertical and horizontal perimeter faces differently because boundary-condition values must be computed differently for horizontal and vertical faces. Boundary-condition types can be mixed on vertical perimeter faces; however, specified-flow boundary conditions must be used where parts of the local model extend into inactive areas of the regional model and computed regional-model head values do not surround the boundary. Boundary conditions for horizontal faces are limited to specified flow. Values for these faces are needed when the uppermost layer in the local model is not the uppermost layer in the regional model or the lowermost layer in the local model is not the lowermost layer in the regional model. Vertical-flow rates for horizontal perimeter faces are computed by area-weighted averaging of vertical flow in the regional model using equation 6. This method preserves the correct total flow rate over the domain of the local model.

Boundary-condition types for the vertical perimeter faces of a local model are specified by four integer arrays. A zero value in an array will specify a specified-flow boundary-condition type and a nonzero value will specify a specified-head boundary-condition type. The first array, FCBCT (First Column Boundary Condition Type), represents cells in local-model column 1; the second array, LCBCT (Last Column Boundary Condition Type), represents cells in local-model column NCOL2; the third array, FRBCT (First Row Boundary Condition Type), represents cells in local-model row 1; and the fourth array, LRBCT (Last Row Boundary Condition Type), represents cells in local-model row NROW2 (fig. 18). NCOL2 and
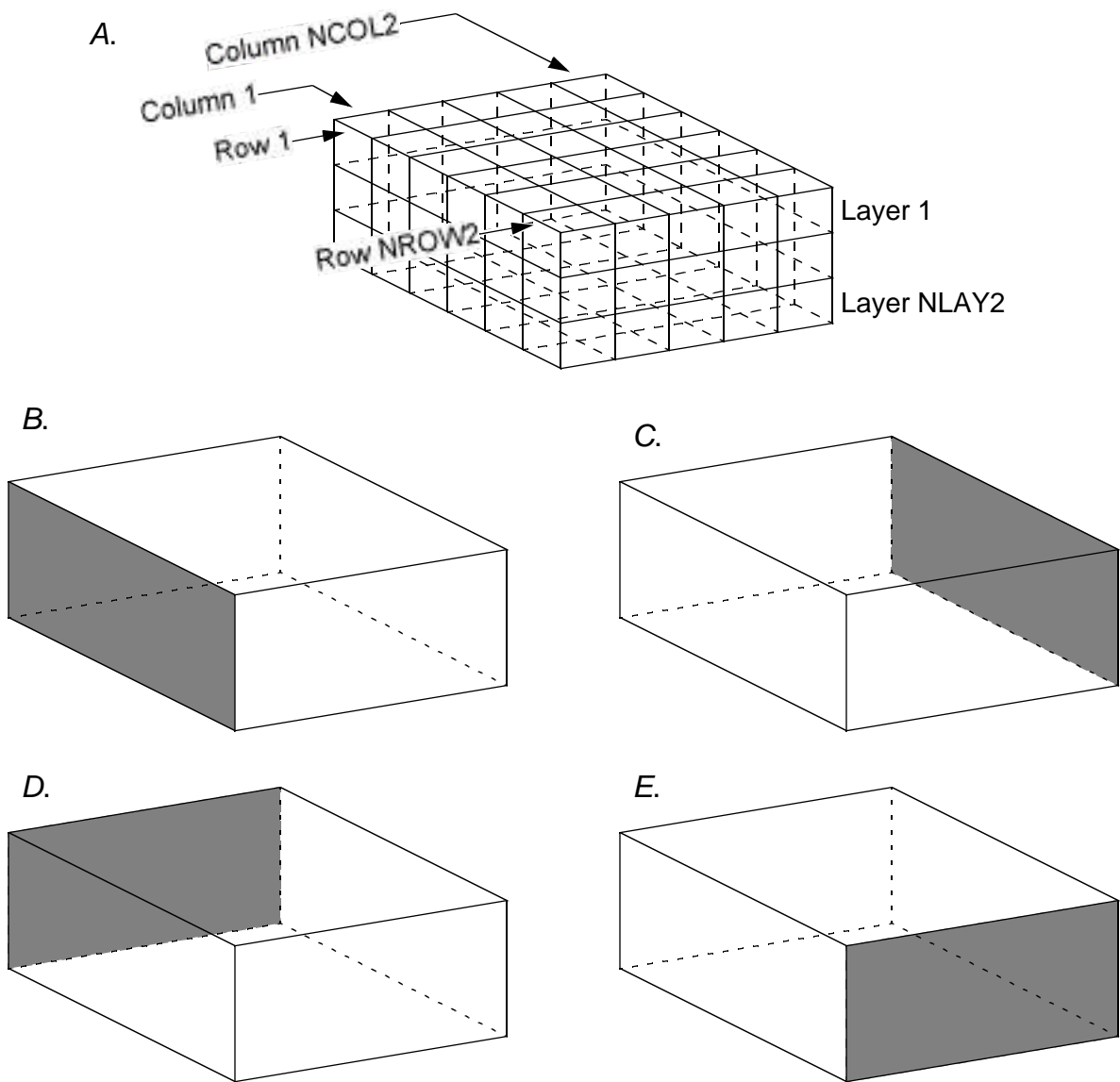
Figure 18. Perimeter vertical surfaces of local model over which boundary-condition types must be specified in arrays. *A*, Example local-model grid. *B*, Surface for specification of boundary condition type in array FCBCT. *C*, Surface for specification of boundary condition type in array LCBCT. *D*, Surface for specification of boundary condition type in array FRBCT. *E*, Surface for specification of boundary condition type in array LRBCT.

NROW2 are the last column and row in the local model, respectively. Columns in arrays FCBCT and LCBCT represent cell locations in local-model rows 1–NROW2 and columns in arrays FRBCT and LRBCT represent cell locations in local-model columns 1–NCOL2. Rows in all four arrays represent model layers in the same way. Each row in an array represents one layer in the regional model. The rows in FCBCT, LCBCT, FRBCT, and LRBCT represent, from uppermost to lowermost, the regional-model layers spanned by the local model (fig. 18). If a regional-model layer is subdivided into two or more local-model layers, one row is used in the boundary-condition type arrays to specify boundary-condition type for each of the subdivided layers.

Local-Model Layers

Local models may include all of the layers in a regional model or a subset of the regional-model layers. Furthermore, regional-model layers with a LAYCON value (Harbaugh and McDonald, 1996) of 0 (zero) may be subdivided into two or more local-model layers. MODTMR does not directly read the number of local-model layers in the input-data set, but rather reads a one-dimensional integer array that specifies the number of local-model layers in each regional-model layer. The array, called the LML or Local-Model-Layer array, includes one element for each layer in the regional model. A value of 0 (zero) means that the corresponding regional-model layer is not included in the local model (fig. 19). A value of 1 means that the regional-model layer is included in the local model and that one local-model layer is used to represent the regional-model layer. A value of $n>1$ means that the regional-model layer is subdivided into n local-model layers.
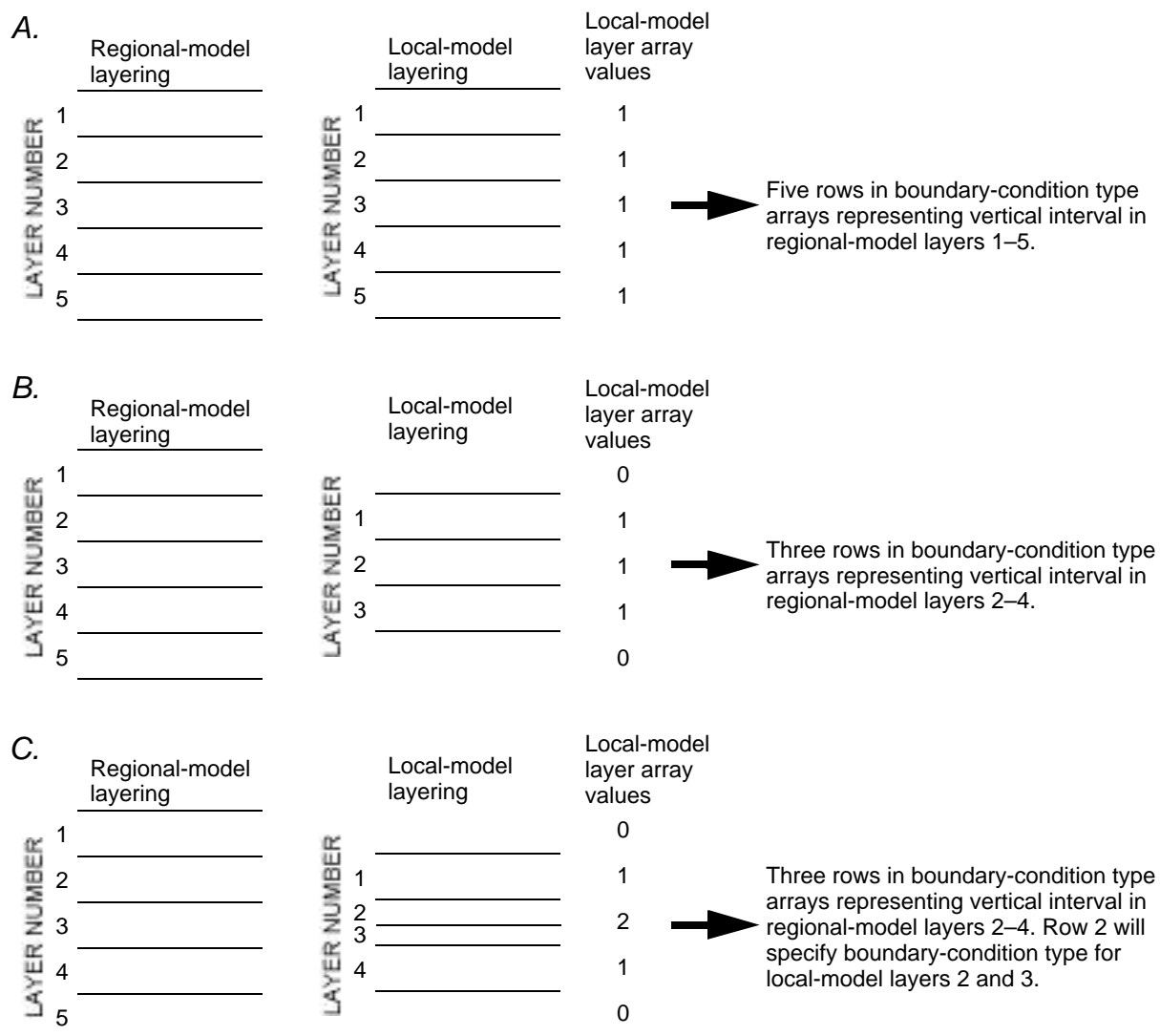
Figure 19. Examples of regional- and local-model layering and corresponding local-model layer values and numbers of rows needed in boundary-condition type arrays. *A*, Example 1, regional and local models have the same layering. *B*, Example 2, local model includes a subset of layers in the regional model. *C*, Example 3, local model includes a subset of layers in the regional model with one regional-model layer subdivided into two local-model layers.

Thickness arrays are needed for each local-model layer that does not span the entire thickness of a regional-model layer. Each thickness array includes one element for each of the NROW2 rows and NCOL2 columns, in the unit of length of the regional model. The thickness values are used in a number of calculations for subdivided layers including calculations of specified-flow perimeter boundary conditions and values of parameters, such as well pumping rate, transmissivity, vertical leakance, and storage coefficient. In the input data set, thickness arrays are read in sequence for layers starting with the uppermost subdivided local-model layer and ending with the lowermost subdivided local-model layer. Thickness arrays are not read for any local-model layer that spans an entire regional-model layer.

Several methods are used for calculating parameters in subdivided layers. Parameters for features and conditions such as recharge, rivers, drains, and evapotranspiration, are assumed to apply only to the uppermost layer of a group of subdivided layers. For integer boundary-array values (IBOUND), starting head, and perimeter specified head, the same horizontally interpolated value for a local-model row and column is used in all subdivided layers at that location. For other parameters, such as transmissivity, storage coefficient, and well pumping rate, horizontally interpolated parameters for a location in the local model are first converted to a value per unit thickness by dividing by the total thickness of subdivided layers at that location. A value at row location i and column location j for subdivided layer m of a regional-model layer subdivided into n sublayers can be calculated as

$$\overline{P}_{i,j,m} = \frac{\overline{P}_{i,j} b_{i,j,m}}{\sum\limits_{k=1}^{n} b_{i,j,k}}, \tag{11}$$

where $\overline{P}_{i,j,m}$ is the interpolated parameter value for the sublayer, $\overline{P}_{i,j}$ is the interpolated parameter for the entire thickness of the regional-model layer, and $b_{i,j,m}$ is the thickness of the mth subdivided layer.

MODFLOW uses vertical leakance (VCONT) read by the Block-Centered Flow Package to calculate vertical conductance between cells. Vertical leakance is K_v/b, where K_v is vertical hydraulic conductivity and b is thickness. For a more detailed discussion of what VCONT can represent, see McDonald and Harbaugh (1988). Subdividing layers in a local model requires creation of new VCONT arrays between subdivided layers and possibly changing the values of VCONT arrays between the outermost subdivided layers in a group of subdivided layers and adjacent layers above and below. Calculation of these arrays rigorously would require additional information on vertical hydraulic conductivity and in some cases on geometry of confining layers not explicitly represented in the regional model. To simplify this procedure, MODTMR constructs VCONT values in a way that makes the local model as much like the regional model as possible. The first part of the approach is to use horizontally interpolated values from regional-model VCONT arrays for the values between the outermost subdivided layers in a group of subdivided layers and adjacent layers above and below (fig. 20). This approach underestimates leakance values because half thicknesses of outermost subdivided layers in the local model are thinner than half thicknesses of the regional-model layers used in the original calculation of leakance. The second part of the approach is to use the assumption that vertical hydraulic conductivity equals horizontal hydraulic conductivity. In reality, vertical hydraulic conductivity is almost always less than horizontal hydraulic conductivity, however, use of a high value of hydraulic conductivity results in less vertical variation in head in the local-model subdivided layers. Using this approach, values in new VCONT arrays within subdivided layers can be computed as K_h/b, where K_h is horizontal hydraulic conductivity, computed as horizontally interpolated transmissivity of the regional-model layer divided by total thickness of all subdivided layers; and b is the sum of the half thickness of two subdivided layers. This approach is meant to be a starting point for construction of a local model. Users of MODTMR should think about what the VCONT arrays represent in the real system and
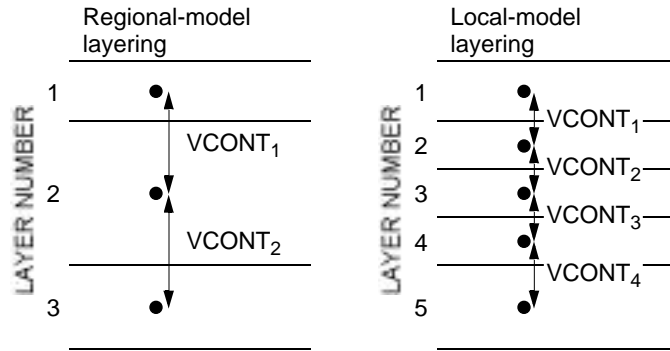
Figure 20. Example of a three-layer regional model with the center layer subdivided into three layers. Two additional VCONT arrays are needed for the interior of the subdivided layer and the outer two VCONT arrays in the local model represent a different vertical interval than in the regional model.

modify the local-model VCONT arrays accordingly, while trying to maintain consistency between the regional and local models.

Construction of Local-Model Input Data Sets for MODFLOW

Perimeter boundary conditions in local models created with the MODTMR are specified using the Flow and Head Boundary Package, version 1, or FHB1 Package (Leake and Lilly, 1997). Execution of MODTMR will create an FHB1 Package data set for the local model regardless of program options. A flag in the MODTMR input data set determines whether the processing will stop after creation of the FHB1 Package data set or processing will continue to construct a MODFLOW-96 name file and input data sets for all packages used by the regional model. Names for files containing local-model data sets are constructed using the local-model data set name, LMDSN, a character-string variable in the MODTMR input. LMDSN contains from one to eight nonblank characters. File names are constructed from LMDSN by appending a period, ".", and a three-character file identifier after the last nonblank character in LMDSN. The three-character identifier for the MODFLOW-96 name file is "nam" and identifiers for all other files are lower case three-character abbreviations for MODFLOW packages (Harbaugh and McDonald, 1996). For example, if LMDSN is "locmdl1", the name of the file for the local-model MODFLOW-96 name file would be "locmdl1.nam" and the name of the file for the local-model Basic Package would be "locmdl1.bas".

MODTMR can create local-model input data sets for all standard packages documented in Harbaugh and McDonald (1996). In addition, MODTMR will create input data sets for several solver packages. The list of supported packages and other input files includes

Input file or Package Name	Reference
MODFLOW-96 name file	Harbaugh and McDonald, 1996
Basic Package, version 5	Harbaugh and McDonald, 1996
Output Control Option file, version 5	Harbaugh and McDonald, 1996
Block-Centered Flow Package, version 5	Harbaugh and McDonald, 1996
River Package, version 5	Harbaugh and McDonald, 1996
Recharge Package, version 5	Harbaugh and McDonald, 1996
Well Package, version 5	Harbaugh and McDonald, 1996
Drain Package, version 5	Harbaugh and McDonald, 1996
Evapotranspiration Package, version 5	Harbaugh and McDonald, 1996
Strongly Implicit Procedure Package, version 5	Harbaugh and McDonald, 1996
Slice-Successive Overrelaxation Package, version 5	Harbaugh and McDonald, 1996
DE4 Package, version 5	Harbaugh, 1995
Preconditioned Conjugant Gradient Package, version 2	Hill, 1990

All input files created by MODTMR, except for the input file for the Preconditioned Conjugant Gradient Package, require the free-format option of MODFLOW-96.

MODFLOW-96 Name File

MODTMR reads the MODFLOW-96 name file for the regional model to get file names and Fortran unit numbers for input data sets. In addition to the restrictions placed on unit numbers by MODFLOW-96, MODTMR requires that no unit numbers for regional-model input and output files be in the range of 90 to 99. The MODFLOW-96 name file for the local model is created if the local-model processing flag, LMPRF, is set to 0 (zero). Unit numbers for individual packages are the same numbers used for each package in the regional model. In general, the local model will include the same package data sets as the regional model. In some cases, however, a feature in a regional model will not be present in a local model because the local-model area does not include the feature. For example, if the regional model includes evapotranspiration, but the area of evapotranspiration does not intersect the area of the local model, there is no need to include evapotranspiration in the local model. If MODTMR can detect that a package for a feature in the regional model is not needed in a local model, the package name will be included in the local-model MODFLOW-96 name file but will be commented out by inserting "# " before the package name.

If the output control for the regional model includes unformatted head and (or) drawdown saved to files, entries for unformatted head and (or) drawdown will be included in the local name file. File names will consist of the local-model data set name followed by the characters ".hed" for the unformatted head file and ".ddn" for the unformatted drawdown file. If the regional model includes cell-by-cell flow terms saved to files, the local-model name file will include entries for one or more cell-by-cell budget files. If cell-by-cell flow terms for all packages are written to the same file, the name of the file will be the local-model data set name followed by the characters ".cbc". If cell-by-cell flow terms for two or more different packages are written to different files, then file names will consist of a three-character package identifier, such as "BCF", followed by the characters ".cbc".

Basic Package

The Basic Package input data set includes two title lines, scalar variables and flags, the boundary array (IBOUND), and starting head (Shead) (Harbaugh and McDonald, 1996). The title lines written to the local-model data set by MODTMR include a message with the name file of the regional-model data set from which the local-model data set was created. Values for flags and variables NPER, IAPART, ISTRT, HNOFLO, NSTP, and TSMULT use values taken directly from the regional-model Basic Package data set. Other variables including NLAY, NROW, NCOL, ITMUNI, and PERLEN have values in the local-model data set that are appropriate for the local model. The keyword "FREE" is put on the options line in the local-model data set. Values in the local-model IBOUND array are computed using interpolation method 1. Values in the Shead array are computed using interpolation method 3.

Output Control

MODTMR requires that the regional-model output-control data set be specified using words instead of numeric values and that one of the specifications for the simulation be "COMPACT BUDGET FILES". MODTMR creates the local-model Output Control data set by copying lines from the regional-model Output Control data set.

Block-Centered Flow Package

The input data set for the Block-Centered Flow Package includes flags, unit numbers, scalar variables, and all or a subset of arrays LAYCON, TRPY, DELR, DELC, Sf1, Tran, HY, BOT, Vcont, Sf2, TOP, and WETDRY. Integer values for LAYCON are taken from the regional-model LAYCON array. Values in the TRPY array depend on the angle of rotation between the regional-model *x* axis and the local-model *x* axis. If the absolute value of the sine of the angle is greater than or equal to 0.999998, a program flag is set to indicate that local-model columns are aligned with regional-model rows. In that case, TRPY values are set to the reciprocals of their values in the regional model and the array multiplier for Tran and HY will include a factor that accounts for the grid rotation. For example, with a TRPY value for a layer of 0.1, and hydraulic-conductivity values along rows (array HY) of 20.0 m/d, MODFLOW will use hydraulic conductivity along columns of 0.1×20.0 or 2.0 m/d in computing conductance along columns. If the rotation angle is 90 degrees (the sine of the angle is 1.0), hydraulic conductivity along local-model rows and columns should be 2.0 and 20.0 m/d, respectively. A TRPY value of 10.0 and an array multiplier of 0.1 results in these correct values. If the absolute value of the sine of the angle of rotation is greater than 0.001745 and is less than 0.999998, then all values in the regional-model TRPY array must be 1.0 and all values in the local-model TRPY array will be set to 1.0. If the absolute value of the sine of the angle is less than or equal to 0.001745, then the regional- and local-model grids are assumed to be aligned and TRPY values in the regional model will be used for corresponding layers in the local model. DELR and DELC values are specified in the MODTMR input data set and those values are written to the local-model Block-Centered Flow Package input data set. Values for local-model cells in arrays Sf1, Tran, HY, Vcont, and Sf2 are computed using interpolation method 2 with equation 5; values for BOT and TOP are computed using interpolation method 3; and values for WETDRY are computed using interpolation method 1. Values of Sf1, Tran, HY, and Sf2 in subdivided layers are computed for individual layers using equation 11.

River Package, Drain Package, and General-Head Boundary Package

The River, Drain, and General-Head Boundary Packages are head-dependent flow-boundary conditions. Each includes a conductance value and a boundary-head value for each location of the boundaries in the model. These features commonly are used to represent linear or point features, such as rivers, streams, drains, and springs. In most cases, these features should be placed in grid cells overlying their correct horizontal location in the local model; however, the only information on locations in the MODFLOW data sets are row and column indices. MODTMR will construct local-model data sets for the River, Drain, and General-Head Boundary Packages by placing these features in any local-model cell that intersects regional-model cells with the features. Conductance for local-model cells are computed using equation 6 and boundary head and other elevation terms are computed using interpolation method 1. Boundary head and other elevation terms include variables Stage and Rbot in the River Package, Elevation in the Drain Package, and Bhead in the General-Head Boundary Package. If a regional-model layer containing a head-dependent flow-boundary condition is subdivided into multiple local-model layers, the head-dependent flow feature will be placed in the uppermost of the multiple local-model layers. Head-dependent flow-boundary conditions are used in some regional models to simulate a laterally distant aquifer boundary. In these cases, the most proper approach in the local model would be to apply the boundary condition to outer cells coincident with the edges of the regional-model cells containing the boundary. Instead, MODTMR will apply the boundary condition in all local-models cells included in or intersected by the regional-model cells containing the boundaries. Solutions to this inappropriate distribution of cells simulating laterally distant boundary conditions are to change the area of extent of the local model so that it does not intersect the regional-model boundary cells or construct the local-model data set for the lateral boundary manually or using another computer program. RIVGRID, which is documented in this report, can be used to construct input data sets for these features in regional and local models using a set of grid-independent data of feature

locations and other properties. If local-model data sets from RIVGRID are used, local-model data sets constructed by MODTMR should be discarded. Regardless of which approach is taken, results of local models should be carefully examined to make sure that flow rates to and from head-dependent boundary features are nearly the same in the local model as in the corresponding area of the regional model. Further calibration of conductance terms in the local model is warranted if flow rates differ significantly.

Recharge Package

The input data set for the Recharge Package includes several integer variables, and for each stress period, an array with recharge rates, and optionally an array with layer numbers. Values in recharge-rate arrays are computed using interpolation method 2, equation 5. Values in the layer-number array are computed using interpolation method 1; however, values are modified appropriately when layer numbers in the local model differ from layer numbers in the regional model. If a layer with recharge in the regional model is subdivided in the local model, recharge is placed in the uppermost subdivided layer.

Well Package

The input data set for the Well Package includes several integer variables; and for each stress period, a list of layer, row, and column indices and corresponding flow rate. MODTMR uses a modified form of interpolation method 2 to compute flow rates for wells in the local model. Instead of computing the area of intersection of a regional-model cell with a well with individual local-model cells, MODTMR computes the intersection of the regional-model cell with the entire domain of the local model. The flow rate used in the local model is the fraction of the regional-model cell within the local model times the pumping rate in the regional model. As discussed previously, this method preserves the consistency between the water budget in the local model and the water budget in the corresponding area of the regional model. Wells in regional-model cells that do not intersect with the local model are not included in the local-model data set. Wells in regional-model cells that do intersect with the local model are placed horizontally in the local-model cell underlying the centroid of the area of intersection. This approach is meant to be an initial step in constructing a local model. Once data sets are constructed using MODTMR, users can reposition wells to their proper locations manually or by use of another computer program. Flow rates for wells in subdivided layers are distributed among sublayers in proportion to sublayer thickness.

Evapotranspiration Package

The input data set for the Evapotranspiration Package includes several integer variables; and for each stress period, arrays specifying elevation of the evapotranspiration (ET) surface, maximum ET flux, ET extinction depth, and layer numbers. Values for all four arrays in the local model are computed using interpolation method 1.

Solver Packages

Solvers packages supported by MODTMR include the Strongly Implicit Procedure Package, Slice-Successive Overrelaxation Package, the Preconditioned Conjugate Gradient Package, version 2, and the DE4 Package. Input for all solver packages includes real- and integer-scalar variables. To construct local-model data sets for these packages, MODTMR copies variables unchanged from the regional-model data set to the local-model data set. The local-model data set for the Preconditioned Conjugate Gradient Package uses formatting from the regional-model data set. Formatting for the other three packages is modified for compatibility with the free-format option used in local-model data sets.

Flow and Head Boundary Package

MODTMR always creates an input data set for the Flow and Head Boundary Package (Leake and Lilly, 1997) in the local model for specified-flow and (or) specified-head boundary conditions derived from a simulation of a regional model. The program, however, will not create an input data set for boundary conditions represented in the regional model with the Flow and Head Boundary Package. The number of simulation times and flow and head values in the data set is given in the NBDTIM variable of the Flow and Head Boundary Package. For steady-state local-model data sets, the value of NBDTIM is 1. MODTMR uses the first cell-wall flow array and (or) the first head array in output files from a steady-state regional-model simulation to construct the boundary conditions for the local model. Values of flow and head are assigned a time value of 0. Construction of a local-model data set for a transient simulation requires output arrays from steady-state and transient regional-model simulations. The steady-state arrays are used in calculating boundary conditions at time 0.0 of the transient simulation and transient-state flow, and head arrays are used in calculating flow and head values at subsequent times. The value of NBDTIM for a transient simulation is 1 plus the number of flow and (or) head arrays saved in the regional-model transient-state simulation. Times for the array values are 0.0 for the steady-state arrays and the regional-model simulation times for each transient-state array. Selections of arrays to be saved for the regional model is carried out by instructions in the Output Control data set. MODTMR requires that arrays be saved using the option "COMPACT BUDGET FILES" because this option saves a simulation time with the array; whereas, the standard option for saving arrays does not. Sizes of local-model data sets for the Flow and Head Boundary Package can be large, particularly if the local model includes many perimeter cells and if many arrays are used to define transient-boundary conditions. In some cases, sizes can be reduced by choosing to save fewer flow arrays and (or) head arrays in the regional-model simulation. This method is reasonable if perimeter flow and (or) head is fairly constant or is changing linearly with time.

MODTMR requires specification of boundary-condition type for all local-model perimeter cells, including those in inactive areas of the regional model. Specified-flow boundaries in inactive areas are constructed using values of 0 (zero) for flow. Specified-head perimeter boundary cells, however, must be surrounded by active regional-model cells. If one or more of the four regional-model cells surrounding a local-model perimeter specified-head cell are not active at the start of a simulation or become inactive during the simulation, MODTMR will not successfully construct the local-model data sets.

The Flow and Head Boundary Package allows for specification of auxiliary variables such as source-fluid concentration or temperature. Data sets created by MODTMR do not include any such auxiliary variables. At each boundary location, however, the Flow and Head Boundary Package specifies a value for an auxiliary integer variable, IAUX. For specified-flow cells, MODTMR sets the value of IAUX to an integer code that specifies the cell face across which boundary flow enters or leaves the local model. The code used is the same as that of the IFACE variable in MODPATH (Pollock, 1994) (fig. 21). For specified-head cells, MODTMR sets the value of IAUX to 0 (zero).
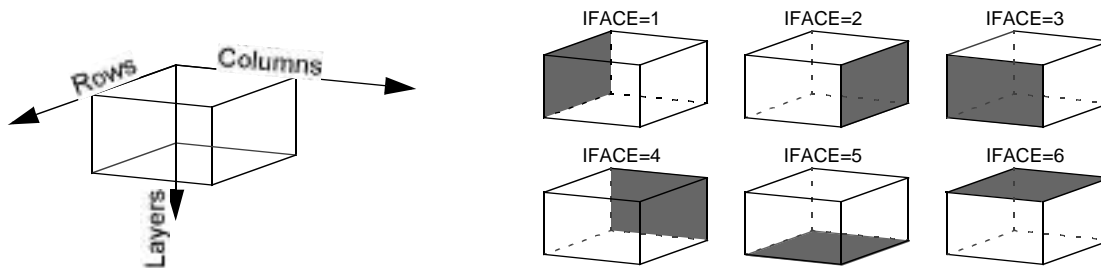
Figure 21. Values of IFACE used by MODPATH (Pollock, 1994) to designate cell faces for flow terms. Program MODTMR inserts values of IFACE in the IAUX parameter for specified-flow cells in the Flow and Head Boundary Package (Leake and Lilly, 1997).

Input Instructions for MODTMR

All input is free format that requires each of the numbered data groups to start on a new input record. Each data group must be within the first 79 spaces on a single line and items within data groups must be separated by at least one space or a comma. Integer data types cannot include a decimal point. Blank spaces are not treated as zeros. For an example annotated input data set, see figure 22.

1. Data: MFNAME LMDSN IFHBUN LMPRF
    Type: Character Character Integer   Integer

2. Data: XYFAC1 ICRNR1 XCRNR1 YCRNR1 ANGL1
    Type: Real   Integer   Real   Real   Real

3. Data: LML
    Type: Integer array

(For data item 3, read NLAY1 integers in free format. Each number in sequence is the number of local-model layers in the corresponding regional-model layer. To omit a regional-model layer from the local model, enter 0 for the corresponding element in the LML array. To subdivide a regional-model layer into more than one local-model layer, enter a number greater than 1.

4. Data: NROW2 NCOL2   ITMUN2
    Type: Integer   Integer   Integer

5. Data: XYFAC2 ICRNR2 XCRNR2 YCRNR2 ANGL2
    Type: Real   Integer   Real   Real   Real

6. Data: DELR2
    Type: Real array

7. Data: DELC2
    Type: Real array

Data items 6 and 7 are arrays describing the grid spacing for the local model along rows and columns, respectively. The arrays are read by MODFLOW utility array reader U1DREL documented by McDonald and Harbaugh (1988) and Harbaugh and McDonald (1996, p. 39–40). For arrays with constant values, use a single record with the word CONSTANT followed by one or more spaces and the constant grid-spacing value. For each array with variable spacing, two or more records are required. The first record includes the word INTERNAL followed by a constant multiplier for each array item, a format for reading array elements, and an integer print code. For detailed instructions on reading arrays, refer to Harbaugh and McDonald (1996, p. 39–40).

8. Data: SSCBC   TRCBC
    Type: Character Character

If the local model does not include any specified-flow cells, include a blank line for data item 8. If local-model simulation is steady state, omit TRCBC.

9. Data: SSHEAD TRHEAD
    Type: Character Character

If the local model does not include any specified-head cells, include a blank line for data item 9. If local-model simulation is steady state, omit TRHEAD.

10. Data: FCBCT

      Type: Integer array

The FCBCT array defines the boundary-condition type for cell faces along column 1 in the local model. Values of 0 (zero) designate specified-flow cells and nonzero values designate specified-head cells. Include a standard header record for U2DINT (Harbaugh and McDonald, 1996, p. 39). List array values along column 1, including one set for each layer in the regional model that is included in the local model.

11. Data: LCBCT

      Type: Integer array

The LCBCT array defines the boundary-condition type for cell faces along column NCOL2 in the local model. Values of 0 (zero) designate specified-flow cells and nonzero values designate specified-head cells. Include a standard header record for U2DINT (Harbaugh and McDonald, 1996, p. 39). List array values along column NCOL2, including one set for each row in the regional model that is included in the local model.

12. Data: FRBCT

      Type: Integer array

The FRBCT array defines the boundary-condition type for cell faces along row 1 in the local model. Values of 0 (zero) designate specified-flow cells and nonzero values designate specified-head cells. Include a standard header record for U2DINT (Harbaugh and McDonald, 1996, p. 39). List array values along row 1, including one set for each row in the regional model that is included in the local model.

13. Data: LRBCT

      Type: Integer array

The LRBCT array defines the boundary-condition type for cell faces along row NROW2 in the local model. Values of 0 (zero) designate specified-flow cells and nonzero values designate specified-head cells. Include a standard header record for U2DINT (Harbaugh and McDonald, 1996, p. 39). List array values along row NROW2, including one set for each row in the regional model that is included in the local model.

14. Data: THK

      Type: Real array

The THK array defines the thickness, in units of length of the regional model, of a local-model layer that does not span the entire thickness of the regional-model layer. The number of thickness arrays needed is the sum of all values in the LML array (See data item 3 and section entitled, "Local-Model Layers") that are greater than 1. Each array includes NROW2 rows and NCOL2 columns of thickness values. Include arrays starting with the uppermost subdivided local-model layer and continuing to the lowermost subdivided layer. Include a standard header record for U2DREL (Harbaugh and McDonald, 1996, p. 39).

```
regss.nam locss 17 0  MFNAME, LMDSN, IFHBUN, LMPRF
1. 4 1000. 1000. 0.  XYFAC1,ICRNR1,XCRNR1,YCRNR1,ANGL1
1 2 0  LML array
16 16 1  NROW2, NCOL2, ITMUN2
0.3048 4 4250. 2500. 30.  XYFAC2,ICRNR2,XCRNR2,YCRNR2,ANGL2
CONSTANT 80  DELR2 array
CONSTANT 80  DELC2 array
regss.cbc  SSCBC
regss.hed  SSHED
INTERNAL 1 (16I1)  8  Header record for FCBCT array
0000000000000001
                    }FCBCT array
0000000000000000
CONSTANT 0  LCBCT array
CONSTANT 0  FRBCT array
CONSTANT 0  LRBCT array
CONSTANT 50.  THK array for local model layer 2
CONSTANT 50.  THK array for local model layer 3
```

Figure 22. Annotated example input data set for program MODTMR. Input data are enclosed in borders.

Explanation of Fields Used in Input Instructions

MFNAME is the name of the file containing the MODFLOW name file for the regional model.

LMDSN is a name for the local-model input-data sets to be created by MODTMR. The name must be eight characters in length or less. For some computer-operating systems, the name must contain only alphabetic characters (upper or lower case) and numbers. MODTMR adds a period to LMDSN and a three-character name extension to construct file names for individual packages being created. For example, if LMDSN is "xyzzy", the name for the MODFLOW name file would be "xyzzy.nam" and the name for the MODFLOW BAS Package would be "xyzzy.bas". Use of spaces, periods, or special characters in LMDSN may cause incompatibility with naming conventions in some operating systems.

IFHBUN is the unit number on which the local-model simulation will read the FHB Package input. Unit numbers for all other MODFLOW packages read by the local model will be the same as the corresponding unit numbers in the regional-model name file.

LMPRF is the local-model processing flag.

   If LMPRF = 0, Local-model data sets will be created for the FHB Package and other packages.

   If LMPRF   0, Local-model data sets will be created only for the FHB Package.

XYFAC1 is a factor for converting units of length in the real-world coordinate system to units of length in the regional-model coordinate system. For models that use the same unit of length as real-world coordinates, use a value of 1.0. For models that use units of feet and real-world coordinates in units of meters, use a value of 3.2808. For models that use units of meters and real-world coordinates in units of feet, use a value of 0.3048.

ICRNR1   is a flag to specify the regional-model corner located at real-world coordinates
XCRNR1,YCRNR1 (see variables X_1 and Y_1 in figure 6).

   If ICRNR1 = 1,   coordinates XCRNR1,YCRNR1 define the location at the intersection of
the outer edges of the first row and first column of the regional-model grid.

   If ICRNR1 = 2,   coordinates XCRNR1,YCRNR1 define the location at the intersection of
the outer edges of the first row and last column of the regional-model grid.

   If ICRNR1 = 3,   coordinates XCRNR1,YCRNR1 define the location at the intersection of
the outer edges of the last row and last column of the regional-model grid.

   If ICRNR1 = 4,   coordinates XCRNR1,YCRNR1 define the location at the intersection of
the outer edges of the last row and first column of the regional-model grid.

XCRNR1   is the distance in the direction of the x axis or ordinate from the real-world coordinate origin to
the regional-model corner defined by variable ICRNR1, in units of the real-world coordinate
system.

YCRNR1   is the distance in the direction of the y axis or abscissa from the real-world coordinate origin to
the regional-model corner defined by variable ICRNR1, in units of the real-world coordinate
system.

ANGL1   is the angle of counter-clockwise rotation measured from the real-world x axis, in degrees, of
the regional-model grid about the corner defined by variable ICRNR1.

LML   is the number of local-model layers in each regional-model layer. Include one 0 (zero) or
positive integer value for each regional-model layer. Values are read free format. For regional-
model layers that are to be excluded from the local model, set the corresponding value of LML
to 0 (zero). For more information on the LML array, see section entitled "Local-Model Layers."

NROW2   is the number of rows in the local-model grid.

NCOL2   is the number of columns in the local-model grid.

ITMUN2   indicates time units of local-model data. The selected time unit is used in calculating parameter
values and boundary-flow values for the local model using information from the regional
model. For additional information, see section entitled "Coordinate Systems and Time Units."

1	seconds	4	days
2	minutes	5	years
3	hours		

XYFAC2   is a factor for converting units of length in the real-world coordinate system to units of length
in the local-model coordinate system. For models that use the same unit of length as real-world
coordinates, use a value of 1.0. For models that use units of feet and real-world coordinates in
units of meters, use a value of 3.2808. For models that use units of meters and real-world
coordinates in units of feet, use a value of 0.3048.

ICRNR2   is a flag to specify the local-model corner located at real-world coordinates
XCRNR2,YCRNR2 (see variables X_2 and Y_2 in figure 6).

   If ICRNR2 = 1,   coordinates XCRNR2,YCRNR2 define the location at the intersection of
the outer edges of the first row and first column of the local-model grid.

   If ICRNR2 = 2,   coordinates XCRNR2,YCRNR2 define the location at the intersection of
the outer edges of the last row and last column of the local-model grid.

If ICRNR2 = 3, coordinates XCRNR2,YCRNR2 define the location at the intersection of the outer edges of the last row and last column of the local-model grid.

If ICRNR2 = 4, coordinates XCRNR2,YCRNR2 define the location at the intersection of the outer edges of the last row and first column of the local-model grid.

XCRNR2 is the distance in the direction of the *x* axis or ordinate from the real-world coordinate origin to the local-model corner defined by variable ICRNR2, in units of the real-world coordinate system.

YCRNR2 is the distance in the direction of the *y* axis or abscissa from the real-world coordinate origin to the local-model corner defined by variable ICRNR2, in units of the real-world coordinate system.

ANGL2 is the angle of counter-clockwise rotation measured from the real-world *x* axis, in degrees, of the local-model grid about the corner defined by variable ICRNR2.

DELR2 is the cell width of the local-model grid along rows. Read one value for each column of the local-model grid. This is a single array with one value for each column.

DELC2 is the cell width of the local-model grid along columns. Read one value for each row of the local-model grid. This is a single array with one value for each row.

SSCBC is the name of the file containing the cell-by-cell flow terms from a steady-state regional-model simulation. Terms must be written using the option of MODFLOW-96 that specifies "COMPACT BUDGET FILES."

TRCBC is the name of the file containing the cell-by-cell flow terms from a transient-state regional-model simulation. Terms must be written using the option of MODFLOW-96 that specifies "COMPACT BUDGET FILES."

SSHEAD is the name of the file containing the unformatted head arrays from a steady-state regional-model simulation.

TRHEAD is the name of the file containing the unformatted head arrays from a transient-state regional-model simulation.

FCBCT is an integer array that specifies the boundary-condition type for cell faces along column 1 in the local model. Array values set to 0 (zero) designate specified-flow cells and array values set to a nonzero value designate specified-head cells. For each layer in the regional model that is included in the local model, include a row with NROW2 integer array values. For more information, see section entitled "Perimeter Boundary Conditions."

LCBCT is an integer array that specifies the boundary-condition type for cell faces along column NCOL2 in the local model. Array values set to 0 (zero) designate specified-flow cells and array values set to a nonzero value designate specified-head cells. For each layer in the regional model that is included in the local model, include a row with NROW2 integer array values. For more information, see section entitled "Perimeter Boundary Conditions."

FRBCT is an integer array that specifies the boundary-condition type for cell faces along row 1 in the local model. Array values set to 0 (zero) designate specified-flow cells and array values set to a nonzero value designate specified-head cells. For each layer in the regional model that is included in the local model, include a row with NCOL2 integer array values. For more information, see section entitled "Perimeter Boundary Conditions."

LRBCT   is an integer array that specifies the boundary-condition type for cell faces along row NROW2 in the local model. Array values set to 0 (zero) designate specified-flow cells and array values set to a nonzero value designate specified-head cells. For each layer in the regional model that is included in the local model, include a row with NCOL2 integer array values. For more information, see section entitled "Perimeter Boundary Conditions."

THK   is an array of thickness values for a subdivided layer, in the units of length of the regional model. Include a standard header record for U2DREL (Harbaugh and McDonald, 1996, p. 39) followed by NROW2 rows and NCOL2 columns of array values. Include one array for each sublayer of subdivided regional-model layers. For more information, see section entitled "Local-Model Layers."

Example Problem

The example problem described in this section illustrates the use of some features of MODTMR to construct local-model data sets from a simple regional model of a hypothetical aquifer system. Features include construction of steady-state and transient data sets, heterogeneous aquifer properties, local-model coordinate-system axes not oriented along with regional-model coordinate-system axes, different units for distance and time in the regional and local models, elimination of a regional-model layer in the local model, and rediscritization of a regional-model layer in the local model.

The domain of the aquifer to be simulated is rectangular with dimensions of 10,000 ft in the direction of the real-world x-coordinate axis and 12,000 ft in the direction of the real-world y-coordinate axis, with a corner of the domain at coordinates x=1,000 ft and y=1,000 ft from the real-world origin (fig. 23). The domain is divided horizontally into two parameter zones and vertically into three layers. Layers are numbered increasing from the uppermost to lowermost layer. Layer 1 contains a water table and a bottom surface that slopes upward linearly in the y direction from an elevation of –120 ft at y=500 ft to an elevation of 120 ft at y=11,500 ft. Layer 1 also includes a constant-head feature at an elevation of 0.0 ft in the location shown in figure 23 and a uniform recharge rate. Layers 2 and 3 are uniform in saturated thickness. Parameters for the two zones in three layers are given in table 1. A well at the location shown in figure 23 begins pumping 20,000 ft^3/d in the year 1985. Prior to that time, flow in the system is assumed to be steady state.

Table 1. Parameter values for layers and zones in example problem for MODTMR

	Layer 1		Layer 2		Layer 3	
	Zone 1	Zone 2	Zone 1	Zone 2	Zone 1	Zone 2
Hydraulic conductivity, in feet per day	3	10	—	—	—	—
Transmissivity, in feet squared per day	—	—	600	2,000	30	100
Vertical leakance, in per day[1]	0.0003	0.001	0.0002	0.00067	—	—
Storage coefficient, dimensionless	0.2	0.3	0.0001	0.0001	0.0001	0.0001
Recharge, in feet per day	0.00556	0.00556	—	—	—	—

[1]Vertical-leakance property applies from center of layer to center of underlying layer.

The regional model for the flow system has 12 rows of a uniform width of 1,000 ft and 13 columns of variable width ranging from 400 to 1,000 ft (fig. 24). The steady-state regional model simulates conditions at the start of 1985 before pumping. Computed head and cell-wall flow terms are saved to files for later use in constructing the steady-state and transient-state local models. The transient model simulates the 15-year period from the start of 1985 through the end of 1999 using 10 time steps of increasing length. Years are treated as having an equal length of 365.25 days, resulting in a total simulation time of 5,478.75 days. Computed head and cell-wall flow terms are saved to files each time step for later use in constructing the transient-state local model.
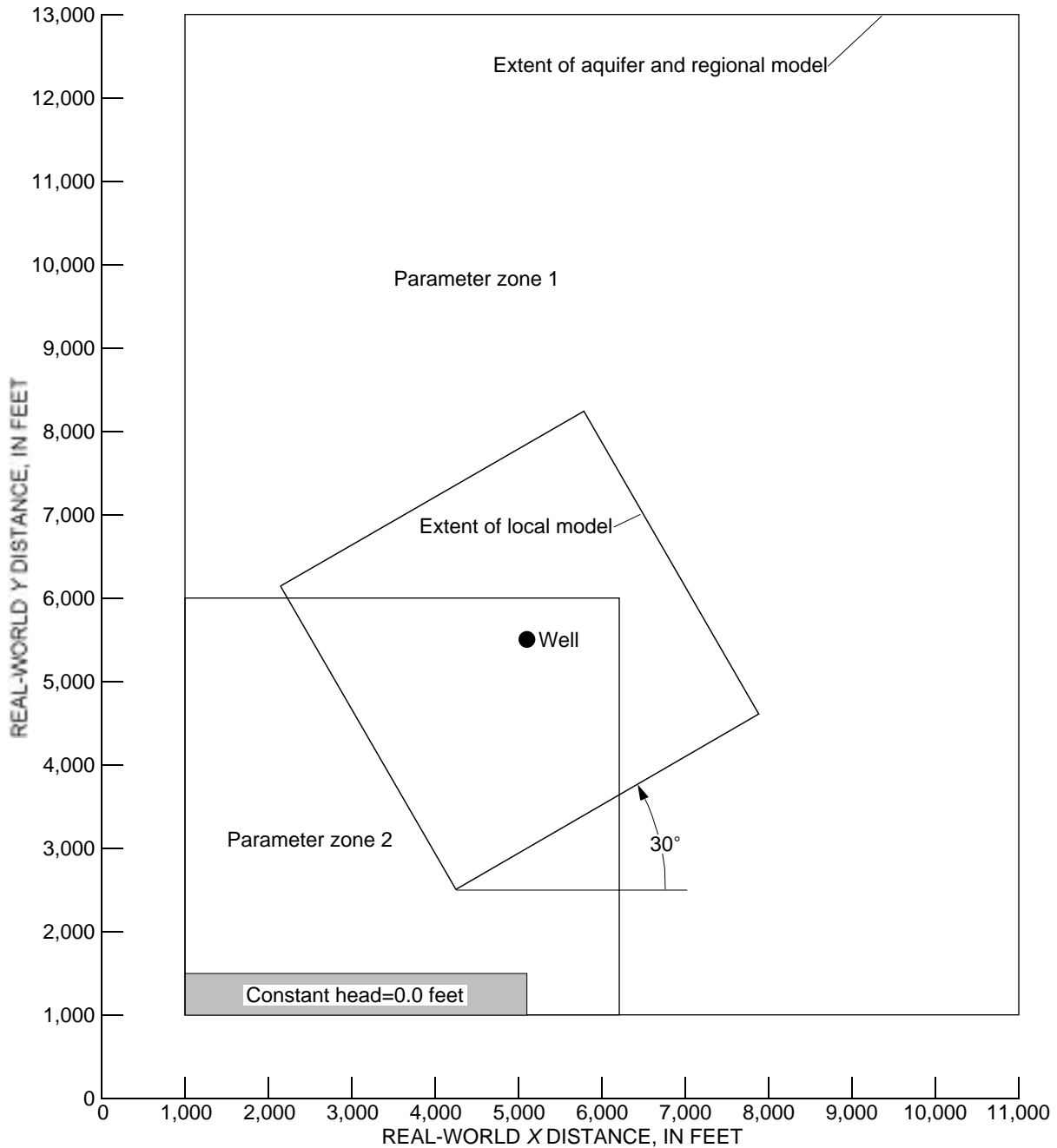
Figure 23. Location of aquifer-system and related features to be simulated in regional-model of example problem for MODTMR.

A local model with the location and orientation shown in figures 23 and 24 is to be constructed by MODTMR to provide more detail in the area of the well. The local-model location, dimensions, orientation, layering, and distance and time units are chosen here to illustrate features of MODTMR and do not necessarily reflect optimum choices for construction of a local model. Units of length for the local model will be meters and units of time for the local model will be seconds. The local model will simulate flow in the upper two layers of the regional model. Furthermore, layer 2 of the regional model will be subdivided into two layers in the local model. The local-model grid will include 16 rows 80 m wide and 16 columns 80 m wide,
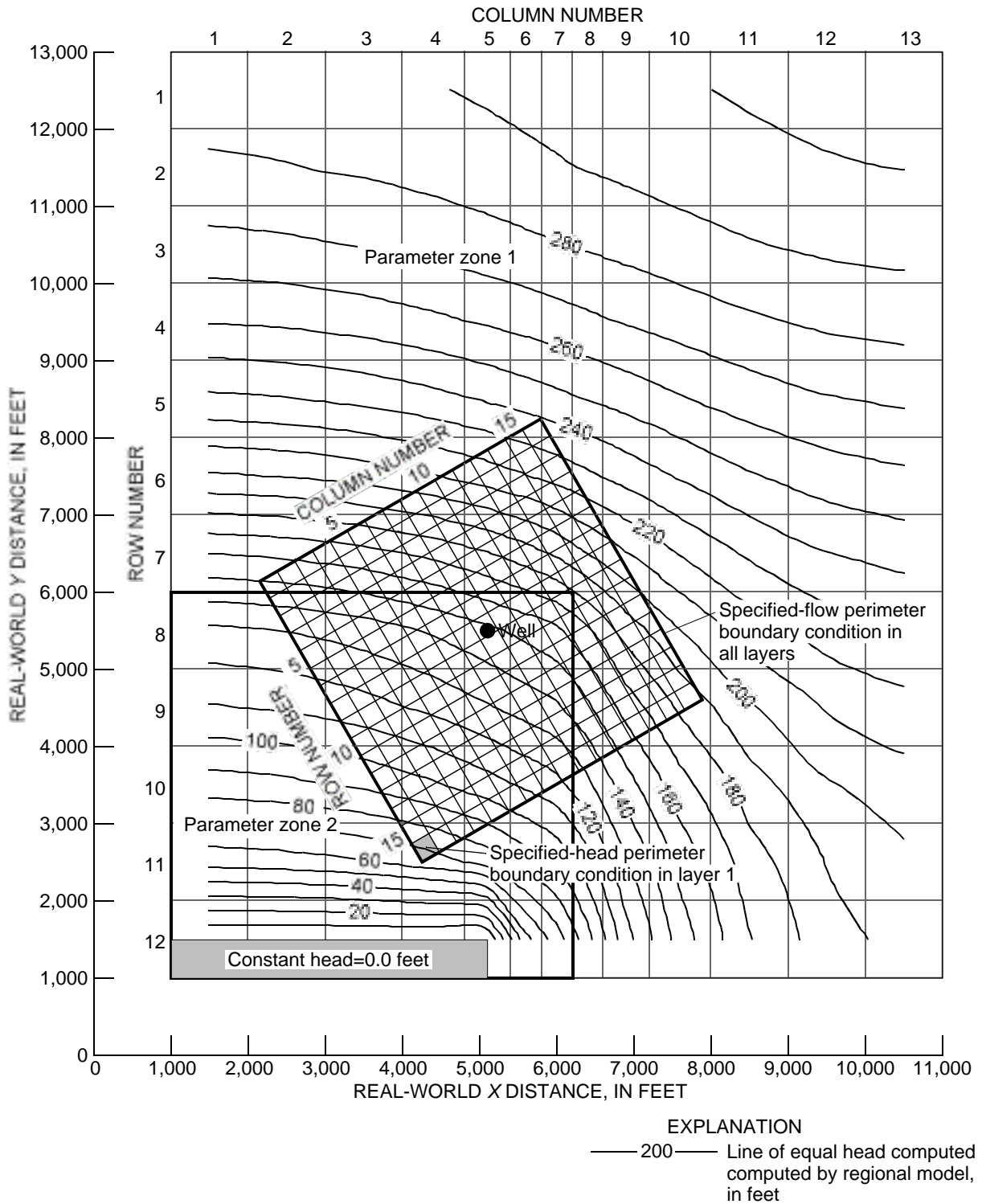
Figure 24. Grids of the regional and local models and head in layer 1 of the regional model.

resulting in a square domain of 1,280 m by 1,280 m. Perimeter boundary conditions of the local model will be specified flow except for the cell in row 16, column 1, layer 1, which will be specified head.

The steps used in the example problem are (1) simulate steady-state flow without pumping in the regional model, (2) construct local-model steady-state data sets using MODTMR and the regional steady-state model input data sets and simulation results, (3) simulate steady-state flow in the local model, (4) compare results of the local model and regional model using TMRDIFF, described in the following section of this report, (5) simulate transient-state flow with pumping in the regional model, (6) construct local-model transient-state data sets using MODTMR and the regional-model input data sets and steady-state and transient-state simulation results, (7) modify the local-model transient-state data sets to use head from steady-state local-model simulation as starting head, (8) simulate transient-state flow in the local model, and (9) compare results of the steady-state local model and regional model using TMRDIFF. Following steps 1 to 3 generates local-model data sets and MODFLOW output for the steady-state regional and local models. Step 2 requires the MODTMR input data set shown in figure 22. Use of TMRDIFF in step 4 allows interpolation of regional-model head at locations of each cell in the local model, allowing a direct comparison of regional-model and local-model results. A comparison of head in layer 1 shows close agreement in computed head in the two models except in the interval between the centers of regional-model rows 7 and 8 where hydraulic properties change abruptly (fig. 25). The differences in this area are related to the ability of the finer grid of the local model to simulate a more detailed head distribution.
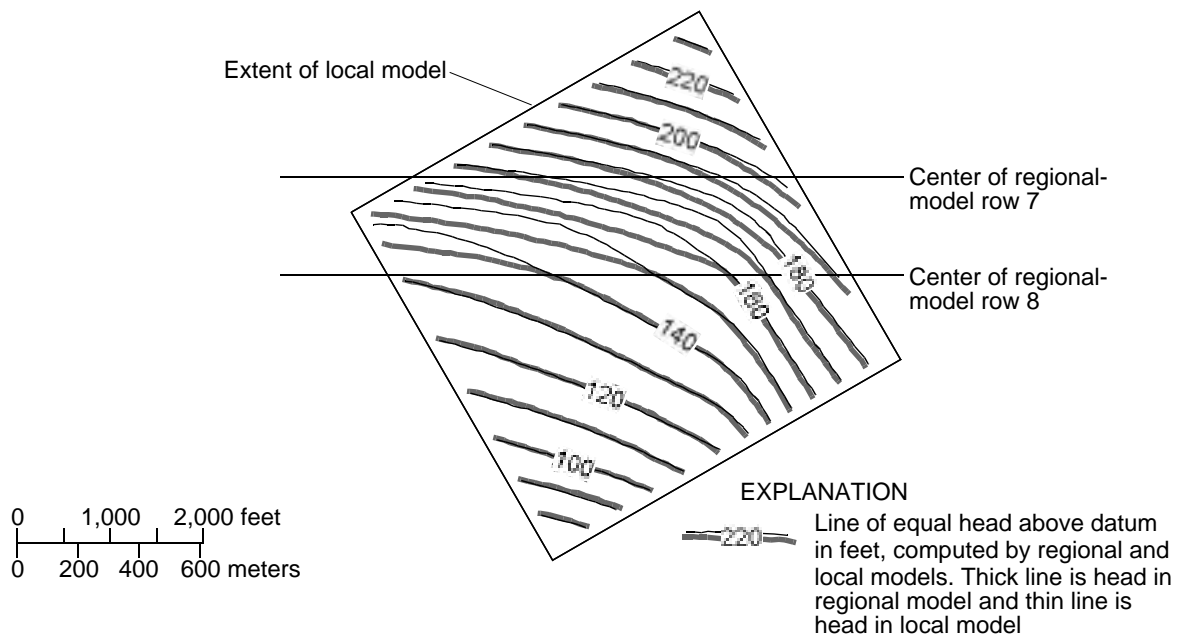
Figure 25. Computed head from example steady-state local model and corresponding part of the regional model.

Construction of the transient local-model data sets in step 6 requires a separate input data set from the one shown in figure 22. The transient MODTMR data set is the same as the steady-state data set (fig. 22) except that variables MFNAME and LMDSN on line 1 must reflect the correct names for transient regional-model and local-model files, and the names of files containing regional-model computed-flow terms and head must be added to the eighth and ninth lines in variables TRCBC and TRHED, respectively. Step 7 should be carried out because of the differences in regional-model and local-model steady-state head in the area of rows 7 and 8 of the regional model. The starting head in the Basic Package data set constructed in step 6 is computed from starting head in the Basic Package of the transient regional model. Because this surface differs in part of the area from the head in the steady-state local model, adjustments in head related to starting conditions will take place in simulations of the transient local model. In this case, the readjustments are of the same order of magnitude as the drawdown from the pumping well. This problem can be

eliminated by using head from the steady-state local model as the starting head for the transient-state local model (fig. 26). For step 8, TMRDIFF was used to generate hydrographs for two locations in the local-model grid. Computed head at corresponding locations in the regional and local models are in close agreement at a distance of more than 1,000 ft from the pumping well, but the head at the location of the pumping well is significantly lower in the local model than in the regional model. The difference in head reflects the ability of the finer grid of the local model to simulate a deeper cone of depression than the coarser grid of the regional model.
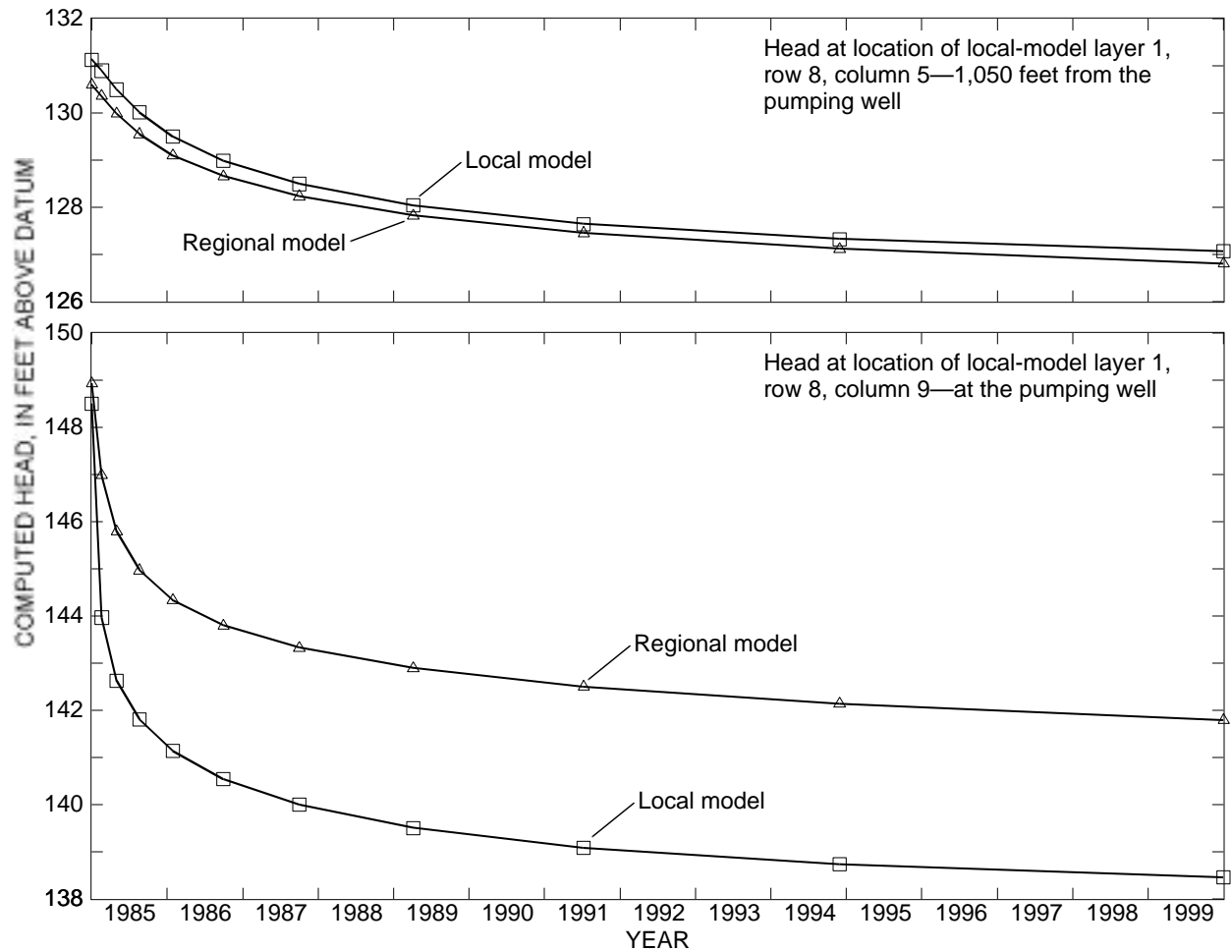
Figure 26. Computed head from two locations in example local model and corresponding locations in regional model.

TMRDIFF

The purpose of TMRDIFF is to provide a means of comparing head and drawdown in a local model with head and drawdown in the corresponding area of the regional model. Because local-model and regional-model grid spacing is different, head and drawdown are not computed at the same locations in the two models. A method is needed to provide output for the two models with head and drawdown calculated at the same points.

Conceptualization of TMRDIFF

TMRDIFF reads computed output from local and regional simulations and writes information on head and drawdown for selected cells in the local-model grid and selected time steps and stress periods. The information written can be (1) arrays of head or drawdown for a specified model layer, stress period, and time step, or (2) time series of head or drawdown for selected local-model layer, row, and column locations for all simulation times at which information was saved from simulations of the local and regional models. For either (1) or (2), output consists of (A) head or drawdown in the local model, (B) head or drawdown in the regional model, and (C) difference between local-model and regional-model head or drawdown.

Because head and drawdown are not computed at the same locations in the local and regional models, a method is needed to compute head in the regional models at locations of local-model cell centers. TMRDIFF uses the bilinear-interpolation method described earlier to compute head or drawdown at desired locations in the regional model. For arrays of head and drawdown, TMRDIFF applies the bilinear interpolation to get values at the location of each local-model cell center for a specific model layer, stress period, and time step. If a point of interest in a model domain is out of the active-flow area, the interpolation procedure returns a value of MODFLOW variable HNOFLO. If a point of interest in a model domain is in a dry cell, the interpolation procedure returns a value of MODFLOW variable HDRY. For information on variables HNOFLO and HDRY, see Harbaugh and McDonald (1996).

TMRDIFF can write arrays of head or drawdown in either the distance units of the local model or in the distance units of the regional model. The program can save array values in the way that MODFLOW normally writes unformatted head or drawdown arrays, or in the way that MODFLOW normally saves formatted head or drawdown arrays. Optionally, arrays can be saved in a format that includes the x and y coordinates of each local-model cell followed by the head or drawdown value for the cell. Coordinates and head or drawdown values for each cell are on a single line, separated by commas. The option of writing head or drawdown arrays always produces three separate files, each containing an array. The name of the first file is the Local-Model Data Set Name (LMDSN in the MODTMR input data set) followed by the characters ".HD1" for a head array or ".DD1" for a drawdown array. The array in the first file is the head or drawdown in the regional model interpolated at cell centers of the local model. The name of the second file is the Local-Model Data Set Name followed by the characters ".HD2" for a head array or ".DD2" for a drawdown array. The array in the second file is the head or drawdown in the local model. The name of the third file is the Local-Model Data Set Name followed by the characters ".HD3" for a head array or ".DD3" for a drawdown array. The array in the third file is head or drawdown in the local model minus the head or drawdown in the regional model interpolated at local-model cell centers.

For hydrographs of head or drawdown, TMRDIFF applies bilinear interpolation to get values for one or more cell locations in the local-model grid. Output from the hydrograph option of TMRDIFF is a file for each point of interest that includes columns of data with times and head or drawdown from the local and regional models. Columns of data may be delimited either by spaces or by commas. Head or drawdown can be given in either the distance units of the local model or in the distance units of the regional model. The times for hydrograph points are the simulation times at which arrays of head and drawdown were saved in the simulations of the local and regional models. Time units for hydrograph points can be seconds, minutes, hours, days, or years, regardless of the time units used by the local and regional models. Furthermore, a starting time can be entered for the hydrograph series, allowing conversion from simulation time to a real-world time. Each hydrograph file contains five columns. The following five lines are from an output hydrograph file, written in comma-separated format:

```
1.968000E+03,6.332048E+03,1.968000E+03,6.333516E+03,1.467285E+00
1.970000E+03,6.332048E+03,1.970000E+03,6.334506E+03,2.458496E+00
1.970400E+03,6.332047E+03,1.970400E+03,6.334702E+03,2.654297E+00
1.971000E+03,6.332047E+03,1.971000E+03,6.334981E+03,2.934082E+00
1.971400E+03,6.332047E+03,1.971400E+03,6.335162E+03,3.114258E+00
```

The first column is simulation time in the regional model, the second column is head or drawdown in the regional model interpolated at the location of the cell of interest in the local model, the third column is simulation time in the local model, the fourth column is head or drawdown at the local-model cell of interest, and the fifth column is head or drawdown in the local model minus head or drawdown interpolated at the same location in the regional model. Local models constructed with MODTMR automatically save head and drawdown files at the same times as corresponding regional models. If output control for either the local or regional model is modified so that arrays are written at different times, output from the hydrograph option of TMRDIFF might not be meaningful.

Files produced by TMRDIFF can be read into contouring programs or spreadsheet programs for graphical display of model results. An example contour plot of output from TMRDIFF is shown in figure 25 and an example hydrograph plot is shown in figure 26.

Input Instructions for TMRDIFF

The main input to TMRDIFF is interactive. To run the program, the following is needed:

1. A MODTMR input data set.
2. MODFLOW data sets for a regional model.
3. MODFLOW data sets for a local model created with MODTMR.
4. Unformatted output of head or drawdown for appropriate simulation times for the local and regional models.

With the proper data sets available, TMRDIFF can be run at a command prompt by typing

```
TMRDIFF
```

followed by the enter key. Following that command, TMRDIFF writes output to the screen asking for specific information on the task to be completed. Most of the user responses are either integer numbers indicating a menu selection or file names. For some operating systems, the user responses can be entered into a file and redirected to TMRDIFF at the command prompt to run the program. For responses that have been entered into a file named "response.txt", the command for this operation is

```
TMRDIFF <response.txt
```

followed by the enter key.

An example of an interactive input session with TMRDIFF follows. In this example, the program is being used to generate arrays of head for a model layer. In this example, program messages and prompts are shown in regular text and user response is shown in bold text.

```
 Enter name of the MODTMR data file:
tmrss
 Select an option for comparing local model and regional model results:
    1 - Write arrays of head or drawdown for a local model layer.
    2 - Write hydrograph data for a local model cell.
    3 - End program.
1
 Select data type:
    1 - Head.
    2 - Drawdown.
1
```

```
 Select distance and time units for output:
   1 - Units from the regional model.
   2 - Units from the local model.
1
 Enter local-model layer number:
1
 Select output type:
   1 - MODFLOW formatted array.
   2 - MODFLOW unformatted array.
   3 - XYZ comma separated.
2
 Enter stress period of array:
1
 Enter time step of array:
1
 Enter name of file with unformatted local model head array:
locss.hed
 Enter name of file with unformatted regional model head array:
regss.hed
 Select an option for comparing local model and regional model results:
   1 - Write arrays of head or drawdown for a local model layer.
   2 - Write hydrograph data for a local model cell.
   3 - End program.
3
```

RIVGRID

The purpose of RIVGRID is to construct MODFLOW data sets for any or all of the following: the River Package, the Drain Package, the General-Head Boundary Package, and the Stream Package. RIVGRID can be used in applications of TMR. MODTMR can construct input data sets for the River, Drain, and General-Head Boundary Packages for local models in applications of TMR; however, these data sets result in the head-dependent-flow boundary features to be spread out over entire areas in local models of regional-model cells containing the features. RIVGRID, on the other hand, allows placement of the features in the correct cells in both regional and local models. Also, RIVGRID can be used independently of the process of TMR to construct data sets for the River, Drain, General-Head Boundary, and Stream Packages for any MODFLOW model under construction.

Conceptualization of RIVGRID

Several key assumptions are inherent in the way that RIVGRID constructs data sets for the River, Drain, General-Head Boundary, and Stream packages. First, the locations of the surface-water features represented by the packages are assumed to be constant for the entire period to be simulated. Second, surface-water features are assumed to be narrow in relation to cell size (fig. 27*A*) so that features can be represented in cells traversed by a trace along the center of the feature. This assumption precludes representing surface-water features that are wide in relation to cell size (fig. 27*B*). For a discussion of representing rivers that are much wider than ground-water model cells, see Leake and Lilly (1995).

For RIVGRID, rivers, drains, general-head boundaries, and streams are assumed to be groups of continuous surface-water features. Within each group, part or all of a continuous feature is represented by series of vertices that define the trace of the feature projected onto a horizontal surface. A series of vertices that represents a single continuous surface-water feature is called a segment. Although only the
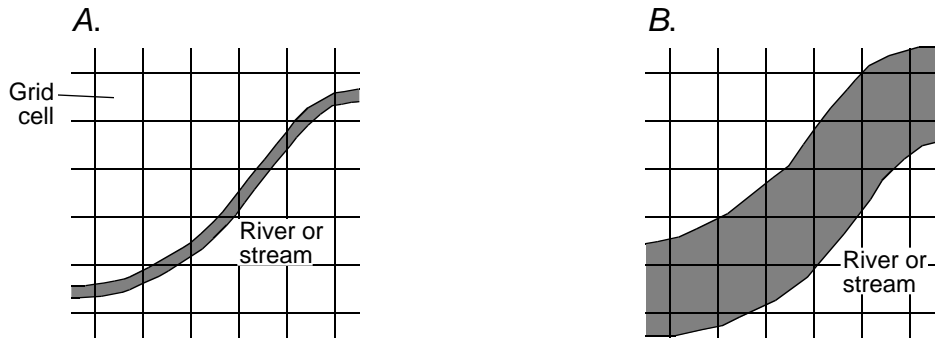
Figure 27. Widths of surface-water features relative to size of ground-water model grid cells. *A*, river or stream is narrow relative to size of grid cells. *B*, river or stream is wide relative to size of grid cells.

MODFLOW Stream Package requires data input by segment, RIVGRID also applies the concept of segments in defining locations of rivers, drains, and general-head boundaries. The example in figure 28 shows one drain segment and three stream segments that intersect a model grid. Segments are defined in terms of a real-world cartesian-coordinate system, such as Universal Transverse Mercator or State-Plane coordinates. Possible sources of vertices for defining segments are data bases in GIS and data digitized from maps. For the Stream Package, vertices must start with the most upstream point on a segment and proceed in a downstream direction. Segments that join at a point as tributaries to a downstream segment should have the same final vertex. Furthermore, the starting vertex of the downstream segment should be the same as the ending vertex of the of the upstream tributary segments (fig. 28). Segments need not be entirely contained within the model grid. Segments that are entirely outside the model grid should not be included in the input to RIVGRID for generating Stream Package data sets.

In addition to having information on locations of segments, RIVGRID must also have information on the model grid. This information includes the location of a selected corner of the grid in the real-world coordinates, the angle of rotation of the grid, the grid spacing, and a factor to convert real-world units of length to the units of length used by the model. Once information on grid locations and segment locations have been read by RIVGRID, the program carries out the following: (1) Rotates, translates, and scales coordinates of all vertices of all segments to a local coordinate system with the origin at the intersection of the outer edges of model row NROW and column 1 (fig. 29). (2) Adds vertices to segments where grid lines (row and column boundaries) cross the segments. (3) For model cells traversed by segments, computes and stores row and column location, length of the segment within the cell, and the distance along the segment from the start of the segment to the midpoint of the piece of the segment within a cell. This information will be used for assigning properties to each cell traversed by a segment. In cells where a segment traverses a corner, the length of the segment in the cell may be small in relation to the size of the cell. For example, the length of stream segment 1 crossing the cell in model row 1 and column 13 in the example in figure 29 is 17.1 m. In cases such as this, RIVGRID adds the small length to the length of the segment in the upgradient or preceding cell and adjusts the distance to the center of segments in that cell. RIVGRID moves pieces of segments to preceding cells when the pieces are less 0.2 times the width of the minimum cell dimension. If a segment in an upgradient or preceding cell does not exist, the short segment length is assigned to the original cell containing the segment. In the example in figure 29, dimensions of the cell at row 1, column 13, are 250 m along rows and 500 m along columns. Pieces of segments in cells that are less than 50 m, therefore, will be grouped with pieces of segments in preceding cells. If a different fraction of the minimum cell dimension is desired, the value of the parameter FMIN in the Main program of RIVGRID must be changed.
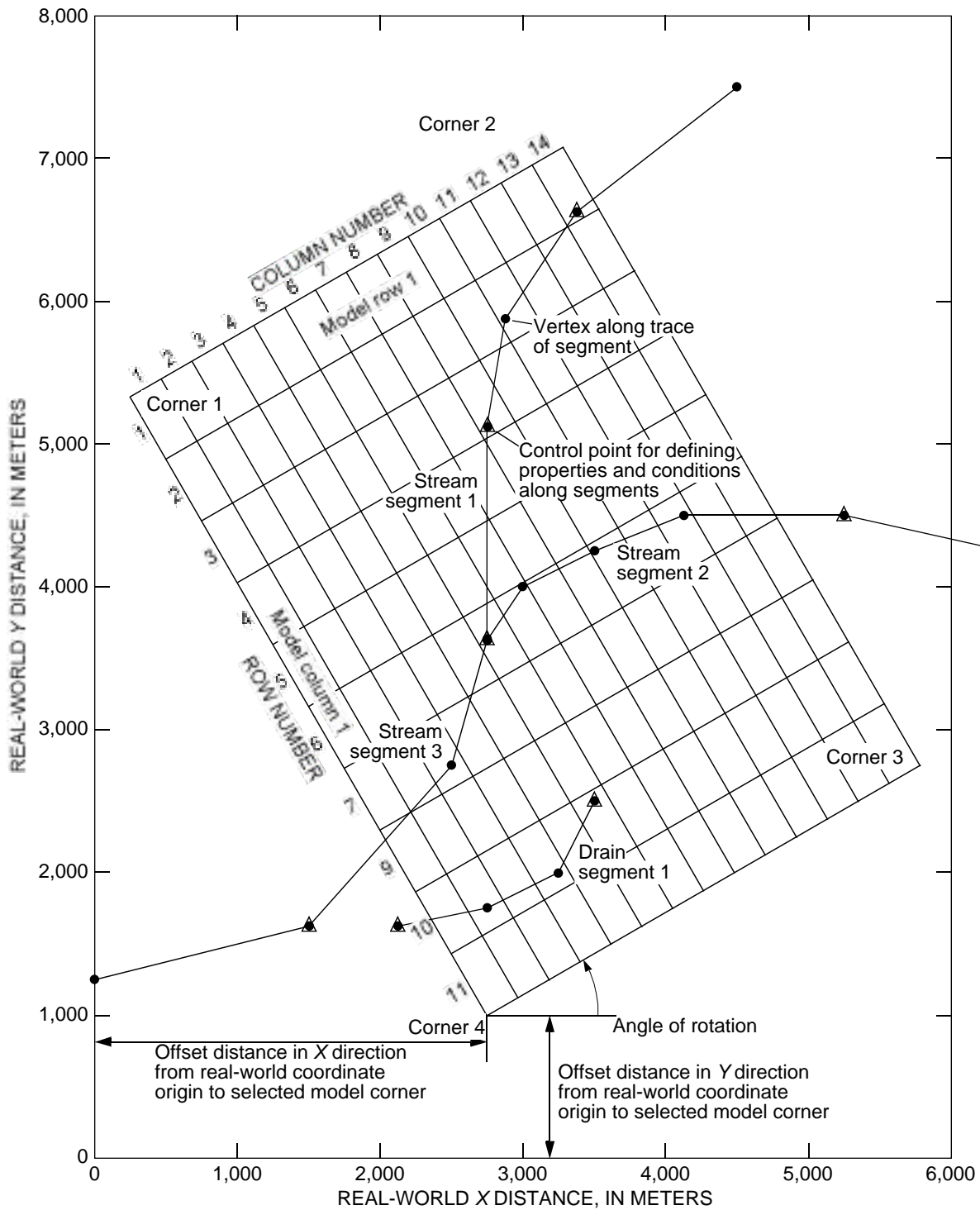
Figure 28. Example model grid, drain segment, and stream segments for an application of RIVGRID.
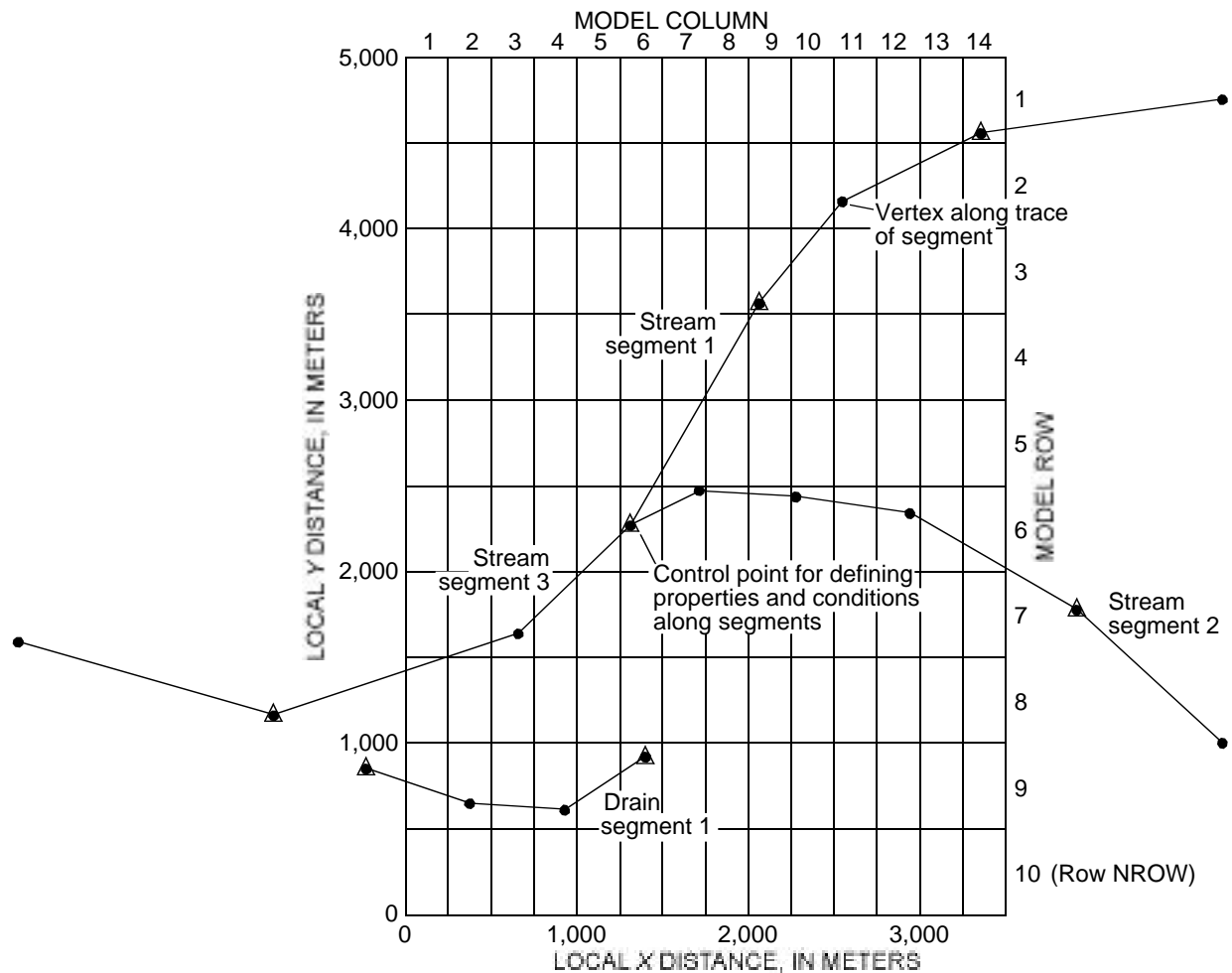
Figure 29. Example model grid with drain and stream segments transformed to local coordinate system.

Within each segment, properties and conditions needed in constructing MODFLOW data sets are entered at control points. Properties needed for MODFLOW packages for cells along traces of segments are interpolated or extrapolated using values of properties at control points. At least two control points are required for each segment. Control-point locations are entered in real-world coordinates. Ideally, control points should be located at a vertex of a segment. If a control-point location does not correspond with the location of a vertex, RIVGRID assigns the control point to the location of the nearest vertex in the segment (including vertices that were originally entered and vertices that were added where grid lines cross the segment). Program output lists the distance from each control point to the nearest vertex. Upon assigning each control point to a vertex along a segment, RIVGRID computes the distance along the segment from the start of the segment to the control point.

Few control points may be needed to define properties and conditions along each segment. Some recommended locations for control points are (1) points where segments cross from one model layer to another, (2) points where slope in stage or bed elevation changes, (3) points where channel width or bed properties abruptly change, and (4) segment endpoints that are coincident with endpoints of adjacent segments.

Properties and conditions that are read for each control point regardless of the package being created are control-point location, model layer, width of channel, vertical hydraulic conductivity of channel bed,

elevation of the top of the channel bed, thickness of the channel bed, and stage. For applications in computing Stream-Package data sets, the flow into the segment is read for all control points but is ignored for all but the most upstream control point in each segment. Furthermore, if the option for the Stream Package to compute stage is being used, channel roughness is read for each control point. Using the channel width, bed thickness, and vertical hydraulic conductivity of the bed, RIVGRID computes the bed conductance per unit channel length for each control point. That value is stored for each control point along with distance from the start of the segment, model layer, bed thickness, and bed top elevation. If the Stream-Package data set is being constructed, flow into each segment is stored and channel width and channel roughness are stored for each control point if stream stage is to be calculated.

Missing information for width of channel, vertical hydraulic conductivity of channel bed, elevation of the top of the channel bed, thickness of the channel bed, stage, and channel roughness may be designated by entering a value of –999. for the missing parameter value. If a parameter value exists at an upstream control point and a value exists at a downstream control point, missing values will be computed using linear interpolation. If two parameter values exist upstream or downstream from the missing value, then misssing values will be computed using linear extrapolation of the two closest values. Parameter values at a minimum of two control points are needed to calculate missing values for a segment. Linear interpolation and extrapolation are discussed in the following paragraphs. Once the missing values have been calculated, RIVGRID can use values at all control points to calculate parameter values for individual reaches. If a value other than –999. is desired to designate missing values, then the value of parameter CTLMIS must be changed in the Main program of RIVGRID to reflect the desired value.

For each cell along the trace of the segment, all properties and conditions needed by model packages except for the model layer are computed by linear interpolation or extrapolation. In this discussion of interpolation and extrapolation, "distance" refers to the distance from the first vertex in a segment to a point on the segment measured along the trace of the segment. For a cell traversed by a segment, suppose that the distance to the center of the fraction of the segment in the cell is d. Furthermore, suppose that d is within the range of the distance to the nth control point, D_n, and the distance to the $n+1$th control point, D_{n+1}. If a property or condition value at the nth control point is P_n and the value at the $n+1$th control point is P_{n+1}, the value, p, at distance d can be computed by linear interpolation as

$$p = P_n + \frac{d - D_n}{D_{n+1} - D_n}(P_{n+1} - P_n) \; . \tag{12}$$

For values of d that are less than the distance to the first control point, D_1, the value of p is extrapolated using $n=1$ in equation 12. Similarly, for values of d that are greater than the distance to the last control point, D_N, the value of p is extrapolated using $n=N-1$ in equation 12, where N is the number of control points for the segment.

Equation 12 is applied to compute properties and conditions such as bed bottom and bed top elevation, water-surface elevation, channel width, and channel roughness for each cell traversed by a segment. Equation 12 also is applied to the property of bed conductance per unit length of channel; however, the interpolated value is then multiplied by the length of the segment in the cell to get the bed conductance for the cell. For values of d within the range of D_n and D_{n+1}, the model-layer number is set to the value at the control point at distance D_n. For values of d less than D_1, the model-layer number is set to the value at the first control point. For values of d greater than D_N, the model-layer number is set to the value at the last control point.

The procedure for reading properties and conditions at control points and computing values at model cells traversed by segments must be carried out once for every model stress period. This allows construction of package data sets with properties and conditions that change with time. Because new lists of control-point information are read for each stress period, numbers and locations of control points can be changed;

however, keeping the same number and location of control points probably is the best approach for most situations. RIVGRID allows reuse of control-point information from a preceding stress period for any or all segments. If control-point information is to be reused for some but not all segments, the interpolation process is carried out as in the preceding stress period to recalculate properties and conditions for individual model cells; however, if control-point information is to be reused for all segments, RIVGRID simply writes a value of –1 for the ITMP variable to signal the River, Drain, General-Head Boundary, or Stream Package that the entire data set will be reused. For more information on the ITMP variable, see package documentation in McDonald and Harbaugh (1988), and Prudic (1989).

Input Instructions for RIVGRID

Input to RIVGRID requires a **Main Input Data Set** that describes the model grid and specifies which MODFLOW Packages are to be constructed. Upon running RIVGRID, the user must enter the name of the file containing the **Main Input Data Set**. For each MODFLOW input package to be constructed, two additional data sets are required, each in a separate file. The first is the **Location Data Set**. This data set consists of lists of coordinate pairs that define horizontal traces of features to be included in the output MODFLOW data set. Lists of coordinates are grouped by segment (see preceding section for a discussion of segments) and for the Stream Package, the coordinates must start at the upstream location and be listed in downstream order. Coordinates for all segments for a package must be included in the **Location Data Set**. The second data set required for each package is the **Control Data Set**. This data set lists information at selected control points along each segment. For a discussion of control points, see previous section.

All input is free format, which requires each of the numbered data groups to start on a new input record. Each data group must be within the first 79 spaces on a single line and items within data groups must be separated by at least one space or a comma. Integer-data types cannot include a decimal point. Blank spaces are not treated as 0 (zero). For an example annotated input-data set, refer to figure 30.

Main Input Data Set

1. Data: NROW   NCOL   NPER
   Type: Integer   Integer   Integer

2. Data: XYFAC   ICRNR   XCRNR   YCRNR   ANGL
   Type: Real   Integer   Real   Real   Real

3. Data: RIVLDS   RIVCDS   RIVODS   IRIVCB
   Type: Character   Character   Character   Integer

If a River Package Data set is not being constructed, include a blank line for data item 3.

4. Data: DRNLDS   DRNCDS   DRNODS   IDRNCB
   Type: Character   Character   Character   Integer

If a Drain Package data set is not being constructed, include a blank line for data item 4.

5. Data: GHBLDS   GHBCDS   GHBODS   IGHBCB
   Type: Character   Character   Character   Integer

If a General-Head Boundary Package data set is not being constructed, include a blank line for data item 5.

6. Data: STRLDS   STRCDS   STRODS   ICALC   CONST   ISTCB1   ISTCB2
   Type: Character   Character   Character   Integer   Real   Integer   Integer

If a Stream Package data set is not being constructed, include a blank line for data item 6.

7. Data: DELR
    Type: Real array

8. Data: DELC
    Type: Real array

Data items 7 and 8 are arrays describing the grid spacing along rows and columns, respectively. The arrays are read by MODFLOW utility array reader U1DREL documented by McDonald and Harbaugh (1988) and Harbaugh and McDonald (1996, p. 39–40). For arrays with constant values, use a single record with the word CONSTANT followed by one or more spaces and the constant grid-spacing value. For each array with variable spacing, two or more records are required. The first record includes the word INTERNAL followed by a constant multiplier for each array item, a format for reading array elements, and an integer print code. For detailed instructions on reading arrays, refer to Harbaugh and McDonald (1996, p. 39–40). For an example, see figure 30.

Location Data Set

1. Data: XLOC   YLOC
    Type: Real   Real

Include data item 1 for each vertex along the trace of a segment. At least two vertices must be included for each segment. For the Stream Package, order vertices in the downstream direction.

2. Data: ENDFLG   IUPSEG   NTRB   ITRIB(1) ITRIB(2) ...   ITRIB(NTRB)
    Type: Character Real   Integer   Integer   Integer   ...   Integer

Include sets of data items 1 and 2 for each segment. For the Stream Package, sets of data items 1 and 2 must follow in sequential order from upstream to downstream. Location data sets must be in separate files for each MODFLOW package.

Control Data Set

1. Data: NCTL
    Type: Integer

2. Data: XCP   YCP   ILAY   RW   VHCOND   RBE   RBTHK   STAGE   Q   ROUGH
    Type: Real   Real   Integer   Real   Real   Real   Real   Real   Real   Real

Omit data item 2 if NCTL is less than one. Otherwise, include NCTL sets of data item 2. Repeat data items 1 and 2 for each segment. Include sets of data items 1 and 2 for all segments for each of the NPER stress periods.

Main Input Data Set

```
        10          14           2   | NROW, NCOL, NPER
       1.0           4       2750.       1000.        30.   | XYFAC,ICRNR,XCRNR,YCRNR,ANGL
                                (Blank line indicating River Package data set will not be created)
drnseg3.dat drnctl.dat test1.drn 0   | DRNLDS,DRNCDS,DRNODS,IDRNCB
                                (Blank line indicating General-Head Boundary Package data set will not be created)
strseg3.dat strctl.dat test1.str 1 1.0   23    0 | STRLDS,STRCDS,STRODS,ICALC,ICONST,ISTCB1,ISTCB2
INTERNAL 1.0 (14F4.0) 0 | Header record for array DELR
 250 250 250 250 250 250 250 250 250 250 250 250 250 250 | DELR—Cell widths for each column
CONSTANT 500. | DELC—All values are set to 500.
```

Location Data Set for Drain Package (file drnseg3.dat)

```
 3500.   2500.
 3250.   2000.   } XLOC,YLOC for each of four vertices that define trace of drain segment 1
 2750.   1750.
 2125.   1625.
END | ENDFLG for drain segment 1
```

Control Data Set for Drain Package (file drnctl.dat)

```
        2 | NCTL for segment 1, stress period 1
  3500.      2500.      1       7.  0.04 507. 1.0 507. } XCP,YCP,ILAY,RW,VHCOND,RBE,RBTHK,STAGE
  2125.      1625.      1       7.  0.04 505. 1.0 507. } for all control points in segment 1
       -1 | NCTL for segment 1, stress period 2 (reuse control information from last stress period)
```

Location Data Set for Stream Package (file strseg3.dat)

```
 4500.   7500.
 3375.   6625.
 2875.   5875.   } XLOC,YLOC for each of five vertices that define trace of stream segment 1
 2750.   5125.
 2750.   3625.
END   0   0 | ENDFLG, IUPSEG,NTRB for stream segment 1
 6375.   4250.
 5250.   4500.
 4125.   4500.   } XLOC,YLOC for each of six vertices that define trace of stream segment 2
 3500.   4250.
 3000.   4000.
 2750.   3625.
END   0   0 | ENDFLG, IUPSEG,NTRB for stream segment 2
 2750.   3625.
 2500.   2750.   } XLOC,YLOC for each of four vertices that define trace of stream segment 3
 1500.   1625.
    0.   1250.
END   0   2   1   2 | ENDFLG, IUPSEG,NTRB, ITRIB(1), ITRIB(2) for stream segment 3
```

Control Data Set for Stream Package (file strctl.dat)

```
        3 | NCTL for segment 1, stress period 1
 3376.   6624. 1 10. 0.01 525. 1.0 526. 425.00 .02377 } XCP,YCP,ILAY,RW,VHCOND,RBE,RBTHK,STAGE,
 2750.   5125. 1 11. 0.01 514. 1.0 515.    0.0 .02377 } Q, ROUGH for all control points in segment 1
 2750.   3625. 1 12. 0.01 509. 1.0 510.    0.0 .02377
        2 | NCTL for segment 2, stress period 1
 5250.   4500. 1  4. 0.01 522. 1.0 523. 610.00 .02377 } XCP,YCP,ILAY,RW,VHCOND,RBE,RBTHK,STAGE,
 2750.   3625. 1  6. 0.01 509. 1.0 510.    0.0 .02377 } Q, ROUGH for all control points in segment 2
        2 | NCTL for segment 3, stress period 1
 2750.   3625. 1 17. 0.01 509. 1.0 510.   -1.  .01500 } XCP,YCP,ILAY,RW,VHCOND,RBE,RBTHK,STAGE,
 1500.   1625. 1 20. 0.01 504. 1.0 505.    0.0 .01100 } Q, ROUGH for all control points in segment 3
       -1 | NCTL for segment 1, stress period 2 (reuse control information from last stress period)
       -1 | NCTL for segment 2, stress period 2 (reuse control information from last stress period)
       -1 | NCTL for segment 3, stress period 2 (reuse control information from last stress period)
```

Figure 30. Annotated example input data sets for RIVGRID. Input data are enclosed in borders.

Explanation of Fields Used in Input Instructions

Input Variables in the Main Input Data Set

NROW   is the number of model rows.

NCOL   is the number of model columns.

NPER   is the number of model stress periods.

XYFAC   is a factor for converting units of length in the real-world coordinate system to units of length in the model coordinate system. For models that use the same unit of length as real-world coordinates defining traces of features such as rivers, drains, and springs, use a value of 1.0. For models that use units of feet and real-world coordinates in units of meters, use a value of 3.2808. For models that use units of meters and real-world coordinates in units of feet, use a value of 0.3048.

ICRNR   is a flag to specify the model corner located at real-world coordinates XCRNR,YCRNR (fig. 28).

        If ICRNR = 1,   coordinates XCRNR,YCRNR define the location at the intersection of the outer edges of model row 1 and column 1.

        If ICRNR = 2,   coordinates XCRNR,YCRNR define the location at the intersection of the outer edges of model row 1 and column NCOL.

        If ICRNR = 3,   coordinates XCRNR,YCRNR define the location at the intersection of the outer edges of model row NROW and column NCOL.

        If ICRNR = 4,   coordinates XCRNR,YCRNR define the location at the intersection of the outer edges of model row NROW and column 1.

XCRNR   is the distance in the direction of the *x* axis or ordinate from the real-world coordinate origin to the model corner defined by variable ICRNR.

YCRNR   is the distance in the direction of the *y* axis or abscissa from the real-world coordinate origin to the model corner defined by variable ICRNR.

ANGL   is the angle of counter-clockwise rotation measured from the real-world *x* axis, in degrees, of the grid about the corner defined by variable ICRNR.

RIVLDS   is the name of the file containing the Location Data Set for constructing a River Package data set for MODFLOW.

RIVCDS   is the name of the file containing the Control Data Set for constructing a River Package data set for MODFLOW.

RIVODS   is the name of the file in which the River Package data set will be written.

IRIVCB   is a flag and a unit number associated with cell-by-cell flow terms in the MODFLOW River Package. The value entered here will be written to the River Package data set constructed by RIVGRID. For more information, see Harbaugh and McDonald (1996, p. 26).

DRNLDS   is the name of the file containing the Location Data Set for constructing a Drain Package data set for MODFLOW.

DRNCDS   is the name of the file containing the Control Data Set for constructing a Drain Package data set for MODFLOW.

DRNODS   is the name of the file in which the Drain Package data set will be written.

IDRNCB   is a flag and a unit number associated with cell-by-cell flow terms in the MODFLOW Drain Package. The value entered here will be written to the Drain Package data set constructed by RIVGRID. For more information, see Harbaugh and McDonald (1996, p. 31).

GHBLDS   is the name of the file containing the Location Data Set for constructing a General-Head Boundary Package data set for MODFLOW.

GHBCDS   is the name of the file containing the Control Data Set for constructing a General-Head Boundary Package data set for MODFLOW.

GHBODS   is the name of the file in which the General-Head Boundary Package data set will be written.

IGHBCB   is a flag and a unit number associated with cell-by-cell flow terms in the MODFLOW General-Head Boundary Package. The value entered here will be written to the General-Head Boundary Package data set constructed by RIVGRID. For more information, see Harbaugh and McDonald (1996, p. 35).

STRLDS   is the name of the file containing the Location Data Set for constructing a Stream Package data set for MODFLOW.

STRCDS   is the name of the file containing the Control Data Set for constructing a Stream Package data set for MODFLOW.

STRODS   is the name of the file in which the Stream Package data set will be written.

ICALC   is a flag, which when positive, specifies that stream stages in reaches are to be calculated by the Stream Package. For more information, see Prudic (1989, p. 29).

CONST   is a constant value used in calculating stream stages by the Stream Package. For more information, see Prudic (1989, p. 29).

ISTCB1   is a flag and a unit number associated with cell-by-cell flow terms in the MODFLOW Stream Package. The value entered here will be written to the Stream Package data set constructed by RIVGRID. For more information, see Prudic (1989, p. 29).

ISTCB2   is a flag and a unit number associated with storing streamflow values in the MODFLOW Stream Package. The value entered here will be written to the Stream Package data set constructed by RIVGRID. For more information, see Prudic (1989, p. 29).

DELR   is the cell width along rows. Read one value for each of the NCOL columns. This is a single array with one value for each column.

DELC   is the cell width along columns. Read one value for each of the NROW rows. This is a single array with one value for each row.

XLOC   is the distance, in real-world measurement units, in the direction of the *x* axis or ordinate from the real-world coordinate origin to a vertex along the trace of a river, drain, general-head boundary, or stream.

YLOC   is the distance, in real-world measurement units, in the direction of the *y* axis or abscissa from the real-world coordinate origin to a vertex along the trace of a river, drain, general-head boundary, or stream.

ENDFLG is a keyword on a line following lines of data with values of XLOC and YLOC. The keyword entered in the data file as END signals the program that the last vertex for a segment has been read.

IUPSEG is the upstream segment number from which flow for the segment terminated by ENDFLG will be diverted. Include this variable only in Location Data Sets for the Stream Package. For a segment that is not a diversion, IUPSEG must be specified as 0 (zero).

NTRB   is the number of tributary segments to the segment terminated by ENDFLG. Include this variable only in Location Data Sets for the Stream Package.

ITRIB(*n*) is the number of the *n*th tributary segment to the segment terminated by ENDFLG. Include this variable only in Location Data Sets for the Stream Package. Include NTRB values to define all tributaries to segment.

Input Variables in the Control Data Set

NCTL   is a flag and a counter.

        If NCTL > 1,   it is the number of control points used for a segment to define properties and conditions for a stress period. At least two separate control points must be used for each segment.

        If NCTL   0,   control-point information from the last stress period will be reused.

XCP   is the distance, in real-world measurement units, in the direction of the *x* axis or ordinate from the real-world coordinate origin to a control point along a river, drain, general-head boundary, or stream.

YCP   is the distance, in real-world measurement units, in the direction of the *y* axis or abscissa from the real-world coordinate origin to a control point along a river, drain, general-head boundary, or stream.

ILAY   is the layer number to which the river, drain, general-head boundary, or stream is connected at the control point. The value of ILAY defines the model layer number for all model cells downstream from a control point (or sequentially after the location of a control point) until the next control point redefines the model layer. For segments all in one model layer, use the same value of ILAY for all control points.

RW   is the width of the river, drain, general-head boundary, or stream at the control point. If width is to be calculated from width at other control points, enter a value of –999.

VHCOND is the vertical hydraulic conductivity of the bed of the river, drain, general-head boundary, or stream at the control point. If vertical hydraulic conductivity is unknown and is to be calculated from vertical hydraulic conductivity at other control points, enter a value of –999.

RBE is the elevation of the top of the bed of the river, drain, general-head boundary, or stream at the control point. If bed elevation is unknown and is to be calculated from bed elevation at other control points, enter a value of –999.

RBTHK is the thickness of the bed of the river, drain, general-head boundary, or stream at the control point. If bed thickness is unknown and is to be calculated from bed thickness at other control points, enter a value of –999.

STAGE is the stage of water in the river, drain, general-head boundary, or stream at the control point. For constructing Stream Package data sets with ICALC greater than 0 (zero), stage is read, but not used. If stage is unknown and is to be calculated from stage at other control points, enter a value of –999.

Q is the Stream Package inflow value for each segment. Omit Q when constructing data sets for the River, Drain, General-Head Boundary Packages. Values are read for all control points, but are ignored for all but the most upstream control point. That value is written in the Stream Package data set as the variable "Flow." For more information on that variable, see Prudic (1989, p. 30).

ROUGH the Manning's roughness coefficient at the control point. Omit ROUGH when constructing data sets for the River, Drain, General-Head Boundary Packages. Omit ROUGH when constructing data sets for the Stream Package with ICALC set to a value less than 1. If roughness coefficient is unknown and is to be calculated from roughness coefficient at other control points, enter a value of –999.

REFERENCES CITED

Buxton, H., and Reilly,T.E., 1986, A technique for analysis of ground-water systems of regional and subregional scales applied to Long Island, New York: U.S. Geological Survey Water-Supply Paper 2310, p. 129–142.

Goode, D.J., 1990, Particle velocity interpolation in block-centered finite difference groundwater flow models: Water Resources Research, v. 26, no. 5, p. 925–940.

Harbaugh, A.W., 1995, Direct solution package based on alternating diagonal ordering for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 95–288, 46 p.

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.

Hill, M.C., 1990, Preconditioned conjugate-gradient 2 (PCG2), a computer program for solving ground-water flow equations: U.S. Geological Survey Water-Resources Investigations Report 90–4048, 43 p.

Konikow, L.F., Goode, D.J., and Hornberger, G.Z., 1996, A three-dimensional method-of-characteristics solute-transport model (MOC3D): U.S. Geological Survey Water-Resources Investigations Report 96–4267, 87 p.

Leake, S.A., 1998, Considerations for assigning parameters in embedded ground water flow models, *in* Poeter, E., Zheng, C., and Hill, M., eds., Modflow 98, Proceedings, October 4–8, 1998, Golden Colorado: Colorado School of Mines, p. 491–496.

Leake, S.A., and Lilly, M.R., 1995, Simulation of the interaction between ground water in an alluvial aquifer and surface water in a large braided river, *in* Charbeneau, R.J., ed., Groundwater Management, proceedings of the First International Conference on Water Resources Engineering, August 14–16, 1995, San Antonio, Texas: American Society of Civil Engineers, p. 325–330.

_____ 1997, Documentation of a computer program (FHB1) for assignment of transient specified-flow and specified-head boundaries in applications of the Modular Finite-Difference Ground-Water Flow Model (MODFLOW): U.S. Geological Survey Open-File Report 97–571, 50 p.

Leake, S.A., Lawson, P.W., Lilly, M.R., and Claar, D.V., 1998, Assignment of boundary conditions in embedded ground water flow models: Ground Water, v. 36, no. 4, p. 621–625.

McDonald, M.G., and Harbaugh, A.W., 1988, A modular three-dimensional finite-difference ground-water flow model: U.S. Geological Survey Techniques of Water-Resources Investigations, book 6, chap. A1, 586 p.

Miller, R.T., and Voss, C.I., 1987, Finite-difference grid for a doublet well in an anisotropic aquifer: Ground Water, v. 24, no. 4, p. 490–496.

Pollock, D.W., 1994, User's guide for MODPATH/MODPATH-PLOT, version 3: a particle tracking post-processing package for MODFLOW, the U.S. Geological Survey finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 94–464, 234 p.

Prudic, D.E., 1989, Documentation of a computer program to simulate stream-aquifer relations using a modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 88–729, 113 p.

Townley, L.R., and Wilson, J.L., 1980, Description of and users's manual for a finite-element aquifer flow model AQUIFEM-1: Ralph M. Parsons Laboratory Technology Adaptation Program Report 79–3, Cambridge Massachusetts.

Ward, D.S., Buss, D.R., Mercer, J.W., and Hughes, S.S., 1987, Evaluation of a groundwater corrective action at the Chem-Dyne hazardous waste site using a telescope mesh refinement modeling approach: Water Resources Research, v. 23, no. 4, p. 603–617.