AnAqSim User Guide

Analytic Aquifer Simulator

© 2022 Fitts Geosolutions, LLC

Table of Contents

1   Introduction

AnAqSim

Analytic Aquifer Simulator

AnAqSim is analytic element software for simulating groundwater flow. It uses subdomains as described in Fitts (2010), which gives it strong capabilities with respect to heterogeneity and anisotropy. It also employs high-order line elements, spatially-variable area sinks, and finite-difference time steps to allow multi-level aquifer systems and wide-ranging transient flow simulations. AnAqSim is a product of Fitts Geosolutions, LLC and was coded in C#.

1.1   AnAqSim Versions

There is one version of AnAqSim, which is a 64-bit Windows application. By default, AnAqSim is unlicensed, but a license can be activated which allows larger models to be solved. The installed software for unlicensed and licensed AnAqSim is one and the same. The differences between the two are described below and hinge on whether you have a valid activated license.

Before release 2021-1, there were 3 versions of AnAqSim: 64-bit, 32-bit, and the free AnAqSimEDU (32 bit). With 2021-1, unlicensed AnAqSim replaces AnAqSimEDU. Unlicensed AnAqSim is still free but has much more generous constraints than AnAqSimEDU allowing larger, more complex problems to be solved.

Unlicensed AnAqSim is free.   It allows modeling of a wide array of situations both steady and transient and for many users this may be all that is needed. When solving, Its limits are as follows:

- Allows multi-level (3D) simulations with up to three aquifer levels.

- It allows fully transient simulations with up to 2 different time periods and up to 5 time steps in each period.

- The number of equations in the model's system of equations is limited to 2000. This allows fairly complex models, but users needing greater complexity should consider AnAqSim.

- AnAqSim can run all of the tutorial models on the website.

These constraints only apply to the solve step. With unlicensed AnAqSim, you may examine, make plots, and use analysis features on models that are larger than these constraints by opening the model file and then loading a saved solution. This way, it is possible to allow others to review any AnAqSim model with the free, unlicensed

AnAqSim.   If you need to solve more complex models, you can purchase license(s) for AnAqSim and increase capabilities as listed below.

Licensed AnAqSim is licensed using one-year activations. See http://www.fittsgeosolutions.com for license and pricing details. Licensed AnAqSim can solve larger, more complex models with these capabilities:

• Allows multi-level (3D) simulations with up to 15 aquifer levels.

• It allows fully transient simulations with any number time periods and up to 20 time steps in each period.

• The number of equations in the model's system of equations is not limited, but somewhere around 40,000 to 50,000 equations is a practical upper limit based on computer memory and the time required to solve.

1.2   System Requirements

• 64-bit Windows operating system.
• At least 350 MB of available disk space.
• At least 1 GB memory. More memory is needed for larger problems and most users have 8+ Gb.
• Microsoft .NET v 4.6.1 or higher.

1.3   Installing AnAqSim

AnAqSim can be downloaded from the Fitts Geosolutions, LLC website. The downloaded file is an executable installation file. To start the installer, run or double-click on the downloaded file. The installer guides you through a typical installation process. The installer checks to see if the right version of the Microsoft .net runtime libraries are installed on your computer (these are used by .net programs like AnAqSim). If they are not there, they will be downloaded (free) from a Microsoft web site and installed before AnAqSim is installed.

Once installed, the program will be included in the Programs part of your Start Menu. The AnAqSim entry has two sub-entries, one starts AnAqSim and the other opens the user guide. AnAqSim is typically installed in this directory on your hard drive:

Program Files/ Fitts Geosolutions/ AnAqSim.

When you first start AnAqSim it is unlicensed; it will run but will have constraints when solving the system of equations. Activating a license will lift these constraints. If you have purchased activations, see License Menu/Activating a License for instructions on how to activate your license.

1.4   AnAqSim License

You may purchase time-limited licenses for AnAqSim, which increase its capabilities. Licenses are node-locked to a specific computer.

To learn about license activation or other license operations, see the License Menu topics. The method of license activation changed in February 2020 to a simpler model that allows for autonomous license activation right after purchasing AnAqSim. Before Feb 2020, license activation required email communication between the user and Fitts Geosolutions.

1.5   Release Version and History

The current AnAqSim release number and a history of AnAqSim releases including lists of changes from one release to another are posted at the website.   The release numbering scheme gives the year followed by the release number in that year. New releases are posted to the web site and may be downloaded, installed, and run by anyone with a valid license.   See the next topic about how to update to a newer release.

You can see the release number of your installed AnAqSim by selecting About AnAqSim on the main menu.

1.6   Updating to a Newer Release

As long as you are within the term of your AnAqSim license, you may upgrade AnAqSim to the current release. To update, first check that there is a newer release available from Fitts Geosolutions, LLC. If there is and you want to update:

1. Download the newer release installation file from the Fitts Geosolutions, LLC web site to your computer,

2. Uninstall the older release from your computer using the Windows Control Panel -> Add/Remove Programs dialog,

3. Install the newer release as described in the *Installing AnAqSim* section.

Updating will not affect your license, as the anaqsim license file is not removed by the uninstall operation.

1.7   Help and Documentation

Help is available in many forms: this User Guide, and instructional videos, tutorials, and example models available at the website.

The AnAqSim User Guide can be accessed by selecting Help from AnAqSim's main menu or from the computer's start menu under AnAqSim. There are two forms of the help file that are loaded into the AnAqSim program directory at installation time:

1. The file AnAqSimUserguide.chm is a Windows help file. This is what is opened from AnAqSim or the start menu.

2. The file AnAqSimUserguide.pdf, which is readable if your computer has the free Adobe Reader software installed.

The Windows help file is the better version to use on the computer, since it contains clearer navigation tools and nice indexing and searching features. The pdf file is the one to use if you want to print a paper copy of the User Guide, since it is laid out in typical 8.5 x 11 paper size with page numbers.

1.8   Tutorials and Videos

The website has three tutorials that walk the viewer through construction of AnAqSim models of increasing complexity. A quick way to get familiar with AnAqSim is to work through these tutorials in sequence, building your own AnAqSim models like the ones shown in the tutorials. The tutorials lead you through creating these models step-by-step. The tutorials are pdf files with with outlines at the first page that contain clickable links, so you can easily jump forward and backward to review, if needed. The completed tutorial models are included in the Documentation directory of the AnAqSim software installation:

- tutor1.anaq is the input file for simple one-level steady model with irregular boundary conditions, a recharge basin, a river, and a well, and anisotropic K.

- tutor2.anaq is the input file for the steady model with heterogeneity and a 3D area with multiple levels.

- tutor3.anaq is the input file for a transient dewatering simulation model within a 3D area.

All three tutorials may be done with either the unlicensed or licensed AnAqSim.

The website also has several short videos that give instruction about different aspects of the user interface (e.g. the *Plot* Tab or Digitizing) and some example models. The User Guide, tutorials, and videos are the primary instruction and support resources.

1.9   Contact and Support

Support is available for licensed AnAqSim users to make sure that AnAqSim is functioning properly on their computer. Please first check the AnAqSim User Guide (Help) and the tutorials and videos at the web site to see if your question may be answered there.

The contact information for support is:

Fitts Geosolutions, LLC
http://www.fittsgeosolutions.com
(207) 510-7650
cfitts@fittsgeosolutions.com

2   AnAqSim Modeling Concepts

This section does not repeat details that are published elsewhere in books and articles, but instead gives a quick outline of the techniques used in AnAqSim and cites the appropriate references for those interested in the details.

AnAqSim employs the analytic element method (AEM), which superposes analytic solutions to yield a composite solution consisting of equations for head and discharge as functions of location and time. The AEM is described in detail in books by Strack (1989) and Haitjema (1995). Shorter summaries of the method may be found in Fitts (2012) and Strack (2003). The AEM is fundamentally different than numerical methods like finite elements and finite differences, where the domain is broken into small blocks or elements and simple head distributions (e.g. linear) are assumed within these blocks or elements. In the AEM, boundaries of the domain are discretized, but the domain itself is not.

AnAqSim uses a variation of the AEM that divides the modeled region into subdomains, each with its own definition of aquifer parameters and its own separate AEM model (Fitts, 2010). The model for a given subdomain (called a *domain* in AnAqSim) includes contributions from elements inside and on the external boundary of the subdomain; elements beyond the subdomain do not contribute. Each subdomain model is written in terms of two-dimensional functions, but three-dimensional flow may be simulated using multiple levels in a model. In multi-level models, the resistance to vertical flow is accounted for in the vertical leakage between levels.

This subdomain approach allows for a high degree of flexibility with respect to a model's heterogeneity, anisotropy, and layering. For example, it is possible for a subdomain that is anisotropic to be adjacent to another subdomain that is isotropic or anisotropic with a different direction and ratio. The subdomain approach allows mixed layering schemes. For example, an area with multiple levels (subdomains stacked vertically and leaking vertically to each other) can abut an area with subdomains in just a single level. This allows the model to focus layering and computational effort in the area of interest, with a simpler single-level model for distant areas.

Another key aspect of AnAqSim is that it allows complete transient simulation capabilities by using finite difference time steps as suggested by Haitjema and Strack (1985). The transient term in the flow equations is handled in essentially the same manner as it is in finite difference programs like MODFLOW.

2.1   Subdomains and Model Levels

AnAqSim uses one separate two-dimensional model for each subdomain. In AnAqSim, subdomain input is entered in the Model Input / Domains data tables (*Domain* in AnAqSim's menu system is short for *subdomain*). In each subdomain model, the resistance to vertical flow is neglected and the head is independent of elevation within the subdomain (Dupuit assumption). Resistance to vertical flow and three-dimensional

flow are modeled by using multiple levels with vertical leakage between levels. To illustrate how subdomains and model levels are implemented in AnAqSim, we will go through several examples, starting from simple and working toward complex.

The simplest model would be one level (two-dimensional), and only one subdomain (homogeneous). A plan view of such a simple model is show below.
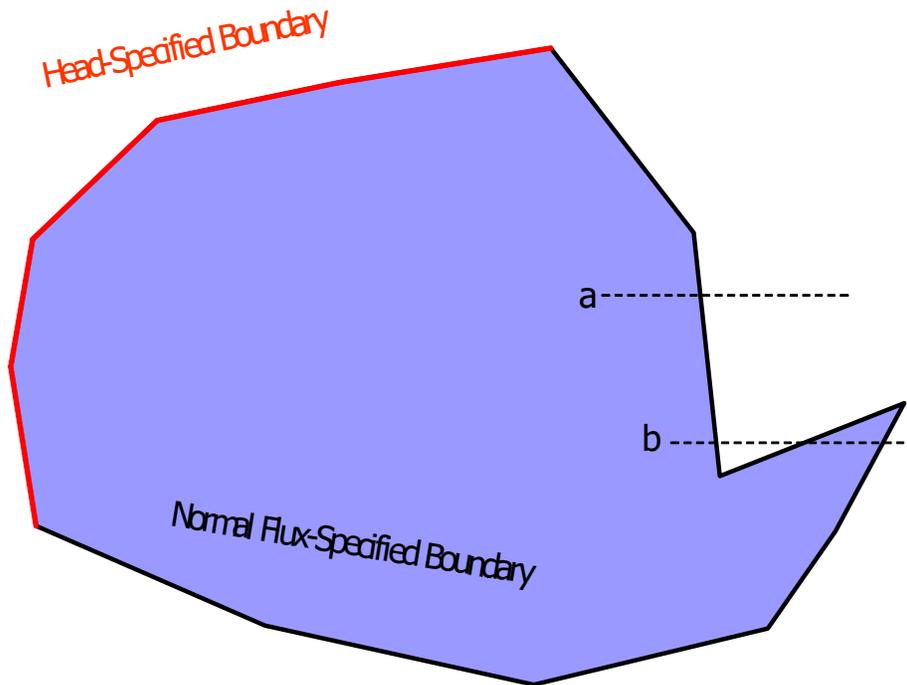
The properties of the subdomain (hydraulic conductivity, porosity, elevations, etc) are defined in the Model Input / Domains menu. The spatial extent of the subdomain (blue) is defined by the line boundaries that are listed as being on the external perimeter of the subdomain. This is different from TWODAN or other analytic element programs where one domain is the infinite "background" domain, and a heterogeneity (domain) lies inside a polygon dedicated to defining the heterogeneity boundary. The scheme in AnAqSim allows different kinds of boundaries to define the limits of a domain, which is much more flexible. In the simple model shown above, there are two line boundaries that form the external boundary of the domain: one that is head-specified (red) and one that is normal-flux specified (black).

The algorithm that determines what points lie inside the domain works as follows: a point is inside the subdomain if going in the positive x direction (right) from the point, you cross a boundary of the domain an odd number of times. For most points inside the domain, there is only one boundary crossing in the positive x direction (like "a" above), but it is possible to have 3 or more crossings (like "b" above).

In general, subdomain boundaries should combine to form a perfectly closed polygon, with exactly matching points where the line boundary end/start points meet. If the line boundaries defining the boundary of the domain have a gap, it can lead to erroneous

definitions of domain areas and odd model results. The boundary gap in the example below allows a strip to the left of the gap (white) that is technically not inside that domain (blue); points in this strip have zero boundary crossings to the right. A similar result happens if there is overlap in the location of two end/start points; where they overlap, the algorithm sees two crossings in the positive x direction.
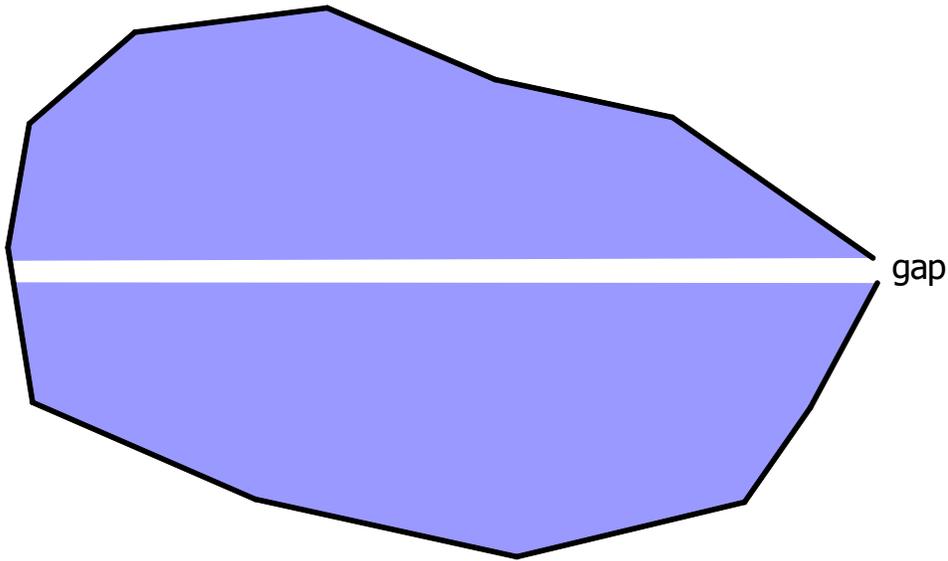
Now consider a model that is still one level (two-dimensional), but is heterogeneous with three subdomains, blue, yellow, and green (below). This example has three types of line boundaries which are labeled: head-specified (hs), normal flux-specified (nfs), and inter-domain (id). Each different line boundary is shown in a different color and is labeled hs, nfs, or id. At all the common intersections where two or more line boundaries meet, the coordinates of the end points must match exactly, so that each subdomain region is defined without   gaps or errors.
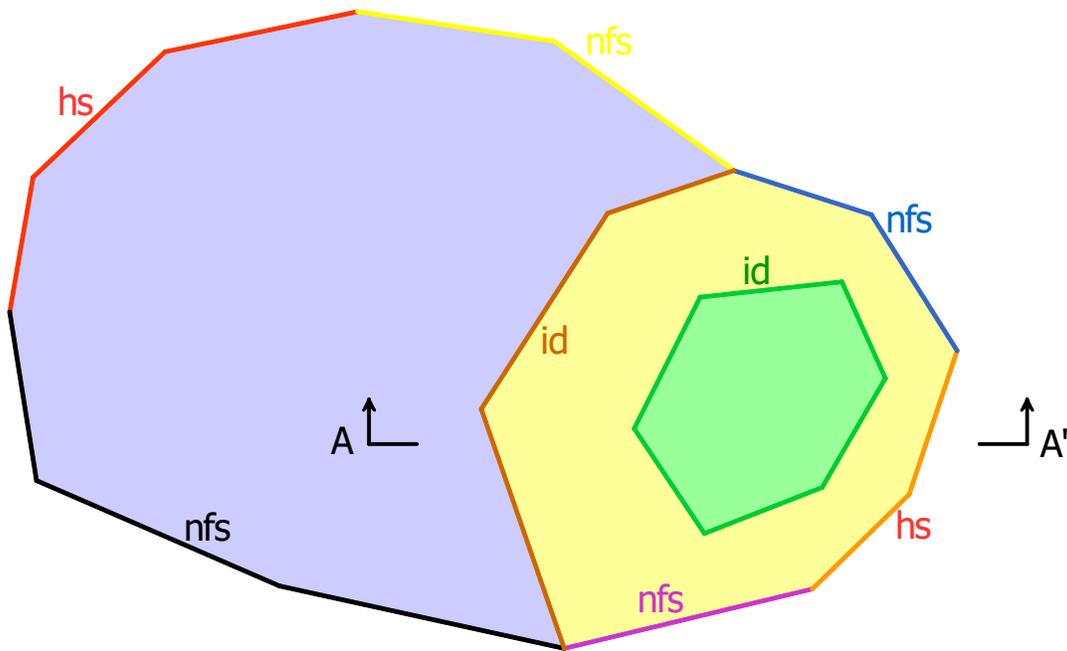
Along inter-domain boundaries, you specify which domain lies to the left and which domain lies to the right. For example, if the coordinates of the inter-domain boundary between blue and yellow are listed in order from bottom to top, blue is to the left and yellow is to the right. If the coordinates of the boundary between yellow and green is defined in clockwise order, yellow is to the left and green is to the right.

Now imagine that the model shown above is a single level in the blue and yellow areas (two-dimensional), but has four levels in green area (three-dimensional). A vertical cross section along A-A' for such a model is shown below.
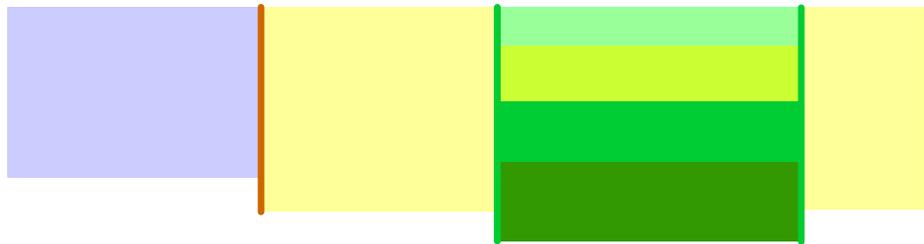
The convention for naming levels in AnAqSim is to start at level 1 at the top and increase the level numbers with depth in a multi-level stack. In the green area, level 1 is at the top and level 4 is at the bottom. AnAqSim assumes that there is vertical leakage between subdomains that exist in the same area but at different vertical levels. It is possible to skip a level number in a stack of subdomains; for example, the green area in the above section could have domains with levels 2,3,5, and 6 rather than levels 1,2,3, and 4. AnAqSim finds the next domains above and below, no matter what the level numbers are and even if there are gaps in level numbers.   This is helpful in cases with complex layering schemes such as where a domain has limited extent compared to the domains above

and/or below it. The resistance to vertical flow between levels is computed based on the vertical hydraulic conductivities and thicknesses specified for the domains involved.

The brown inter-domain boundary separating the blue and yellow subdomains would have one subdomain on the left (blue) and one on the right (yellow). The green inter-domain boundary separating the yellow and green subdomains would have one subdomain on the left (yellow) and four on the right (shades of green), assuming that boundary is defined with coordinates in clockwise order. Across inter-domain boundaries, there is approximate continuity head and approximate continuity of the normal component of discharge. This approximation is discussed more in the *Line Boundary Conditions* section. Interdomain boundaries can connect subdomains in different levels (e.g. level 1 on one side and levels 2 and 3 on the other).

2.2   Left / Right with Respect to Line Boundaries

With interdomain line boundaries and with normal flux-specified line boundaries, it is important to specify boundaries to the left and/or to the right of the line boundary. To explain what this means, refer to the following figure.
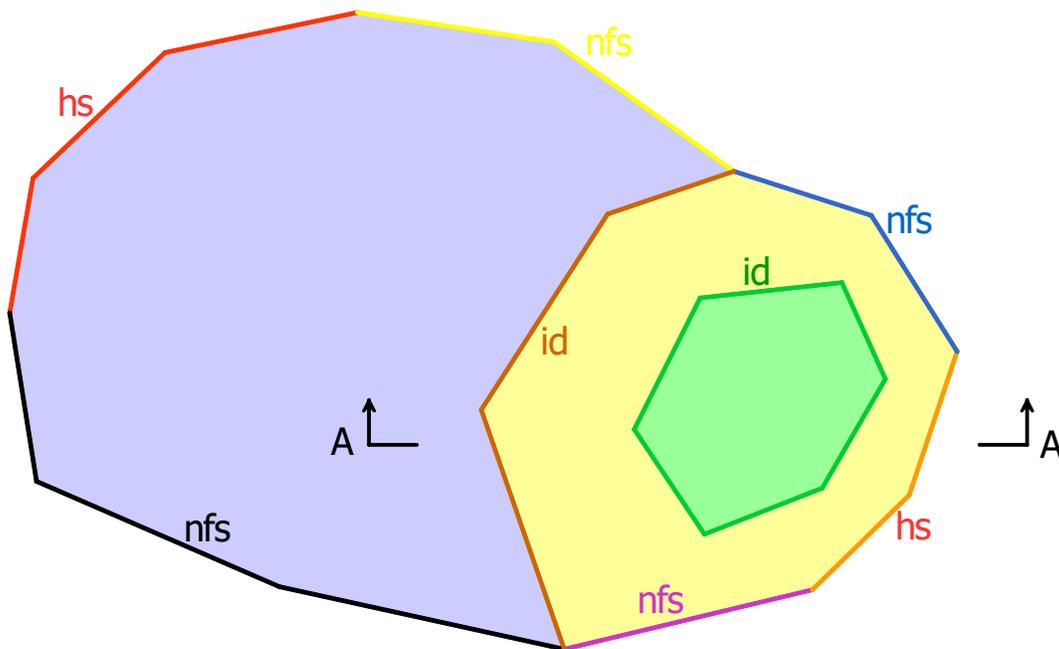
Assume that north is up and south is down in this map-view plot of an AnAqSim model. The different colored areas represent different subdomains, which are bounded by various line boundaries. Consider the red interdomain line boundary (id) that separates the blue and yellow subdomains. If the coordinates for this line boundary are in order from south to north, then the blue subdomain would be to the left and the yellow subdomain would be to the right. Think of it as though you were walking along the line boundary from the first vertex towards the last vertex in the coordinates. The "left" side

is to your left as you walk along the boundary and the "right" side is to the right. Alternatively, if the coordinates of this interdomain boundary are listed in order from north to south, then the yellow domain is to the left and the blue domain is to the right.

Now consider the green interdomain boundary separating the green and yellow subdomains. If the coordinates of that boundary are specified in clockwise order, the yellow domain is to the left and the green domain is to the right. Alternatively, if the coordinates of that boundary are specified in counterclockwise order, the green domain is to the left and the yellow domain is to the right.

When a normal flux-specified boundary is external or with a head-dependent normal flux boundary, the vertex coordinates must be specified in counterclockwise order, with the domain to the left of the boundary as you proceed along it. Consider the purple normal flux-specified boundary (nfs) in the figure. It's coordinates must be specified in counterclockwise order (west to east) with the yellow domain to the left. Likewise, for the blue normal flux-specified boundary, the coordinates must be specified counterclockwise from southeast to northwest, keeping the yellow domain to the left. The normal flux is positive for flow across the boundary from left to right as you proceed from the start toward the end (flow out of subdomain). Normal flux is negative for flow from right to left as you proceed from the start toward the end (flow into subdomain).

2.3   Recharge, Leakage and Transient Storage

Like in any flow model, the flow equation in AnAqSim is based on Darcy's Law and conservation of mass (and volume, with constant density). The conservation equation, in its simplest is:

$- \nabla Q = \gamma = L_t + L_b + S \, \partial h / \partial t$

where ∇Q is the divergence of the two-dimensional aquifer discharge vector field and γ is the net extraction per area (sink term, units of L/T). The sink term γ may have contributions from leakage out the top of the subdomain (L_t), leakage out the bottom of the subdomain (L_b), and transient discharge/area into storage ($S \, \partial h / \partial t$). See equations 4-6 of [Fitts (2010)](#).

The vertical leakages L_t and L_b are specific discharges proportional to the head difference between the domain and the head above or below (could be specified head or another domain at a different level), and proportional to the equivalent vertical hydraulic conductivity K_e that is based on the vertical conductivities and the saturated thicknesses at the average heads specified for the domains involved. If there is leakage between two domains at different levels, the equation used for computing K_e is

$K_e = (b1 + b2) / (b1/K1 + b2/K2)$

where b1 and b2 are the average saturated thicknesses of the two layers and K1 and K2 are the vertical hydraulic conductivities of the two layers. In AnAqSim, K_e is assumed to be constant and independent of head and actual saturated thickness.

Under restricted circumstances (uniform recharge, steady flow, and a single-level model), γ is constant and independent of location. In such cases, the uniform γ distribution may be modeled exactly using a uniform area sink.

In many practical cases, the model needs spatially-variable extraction (γ varies with x, y) due to spatially-variable vertical leakage and/or spatially-variable storage changes. When that is the case, the model needs spatially-variable area sinks to approximate the proper distribution of γ. The spatially-variable area sink functions in AnAqSim create a smooth, continuous, irregular γ surface within a subdomain. The model is using equation 13 of Fitts (2010) as a model of the distribution of γ, with γ equal to the perfect extraction/area (computed by equation 6 of Fitts (2010)) at each basis point, but approximating equation 6 of Fitts (2010) between basis points. The modeled distribution of γ satisfies the flow equation perfectly at basis points and approximates the flow equation between basis points. This approximation is more accurate if the basis point spacing is smaller.

To check the accuracy of this approximation, AnAqSim provides an analysis tool (*Analysis Menu / Graph Conditions Along a Line*). If checks with this tool reveal a poor approximation, the basis point spacing is too large. If the approximation is fine, you may be able to decrease the basis point spacing and save some computation.

Spatially-variable area sinks can rack up a large number of equations and computational burden on the system, so use them sparingly. Where possible (often the far-field), use a single level in the model, which if it is steady means you can use the very efficient uniform area sinks instead of spatially-variable area sinks. Use the special well basis point spacings around wells to gain accuracy with minimal computation.

2.4   Elements

The analytic elements used in AnAqSim include the elements described in the following references:

- The standard well element described by *Strack* [1989] is used in isotropic subdomains and the well element of *Fitts* [2006] is used in anisotropic subdomains.
- Line boundaries are represented by high-order line elements similar to those described by *Jankovic and Barnes* [1999]. Line elements are either linesinks, line dipoles, or line doublets. Line elements may have up to 10 unknown parameters (degrees of freedom). For example, a linesink with one parameter has a constant discharge/length along its length, a linesink with two parameters has discharge/length that varies linearly along its length, and a linesink with three parameters has discharge/length that varies parabolically along its length. *Strack* [1989] gives a good overview of line elements.

- Uniform area source/sinks (constant over a subdomain) are modeled with the uniform recharge functions described by *Strack* [1989]. For these, the extraction/area is uniform over the entire subdomain. This is appropriate in cases with a single level, steady flow, and a uniform recharge rate or zero recharge.
- Spatially-variable area source/sinks are modeled with the multi-quadric radial basis function elements described by *Strack and Jankovic* [1999]. With these, the extraction/area varies with location. Spatially-variable extraction occurs when the extraction = recharge + leakage + transient storage flux is spatially variable. This is generally the case in multi-level models where spatially-variable leakage occurs and in transient models where spatially-variable storage flux occurs. See the previous section and *Fitts* (2010) for details of now these elements are used to represent recharge, leakage and storage flux.

2.5   Wells

There are three possible kinds of pumping wells in AnAqSim:

1. Discharge-specified wells in just one subdomain. With this type of well, you specify the known discharge of the well.

2. Discharge-specified wells spanning multiple subdomains. This is for wells that are screened across multiple levels and subdomains in a multi-level part of a model. For example, a well could be screened across the two lowest levels in a 4-level part of a model. You specify the known discharge of the well, and AnAqSim imposes these conditions: a) the heads at the well radii in each screened subdomain are identical, and b) the total discharge of the well elements in the screened subdomains equals the specified discharge of the well.

3. Head--specified wells in just one subdomain. With this type of well, you specify the known head at the well radius. AnAqSim determines the discharge needed to meet this condition.

2.6   Interdomain Line Boundary Conditions

At the inter-domain boundaries, two different boundary conditions are enforced by AnAqSim. First, heads are matched across the boundary at specific points in all subdomains that intersect the boundary. For example with the green inter-domain boundary shown in the figures below, heads are matched in the yellow domain and the four green domains at control points along the boundary.
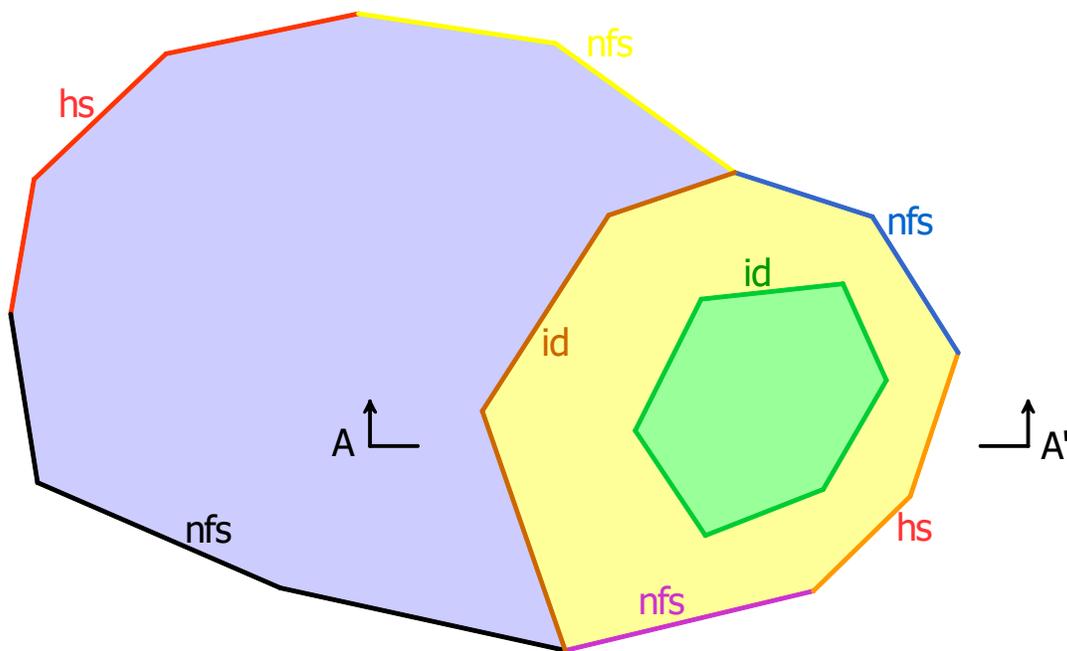
Plan view (above) and section view (below) of a multi-domain model.
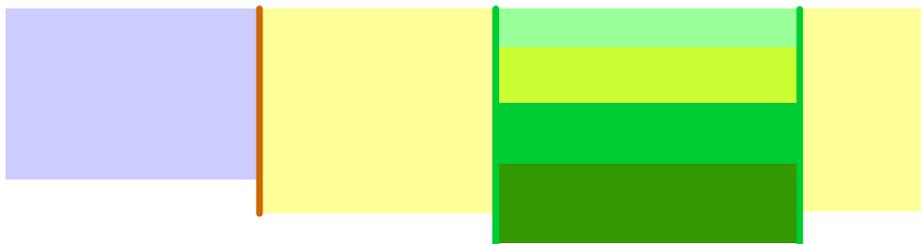
Secondly the normal component of discharge is matched across line segments on the inter-domain boundary. For the green inter-domain boundary, this means that the sum of the normal components of discharge in the four green subdomains must equal the normal component of discharge in the yellow subdomain. This condition is enforced across intervals on a line segment. If each line segment on the inter-domain boundary has n parameters, then the condition is enforced across n intervals per line segment.

A good way to check that you have enough parameters specified on any line boundary condition is to use *right-click/ Check Line Boundary Conditions* which creates a plot of the boundary condition accuracy along a line segment of the line boundary. If you don't have satisfactory accuracy, you can either (1) go under Line Boundaries and specify more parameters (control points) per boundary line segment, and/or (2) break long boundary line segment into more, shorter segments.

One consequence of the head-matching condition is that there will be no vertical hydraulic gradient between different levels on the multi-level side of an inter-domain boundary. This is consistent with the fact that there is no vertical resistance to flow accounted for on the single-level side of an inter-domain boundary. Because of this, it is

not appropriate to use one inter-domain boundary with multiple levels on both sides of the boundary; doing so would remove vertical head gradients at the boundary, which defeats the purpose of having multiple levels.

With inter-domain boundaries, there should generally be just one domain (level) on one of the two sides of the boundary. AnAqSim checks the input and gives a warning if there are two or more levels on both sides of an inter-domain boundary. To explain why, consider the case illustrated below, with two levels on the yellow side of the inter-domain boundary(s) and four on the green side. If there was just one inter-domain boundary there, it would be as though an infinitely conductive thin vertical boundary were inserted between the yellow side and the green side, and heads would match in all six domains that meet there, meaning there can be no vertical head gradient on either side of the boundary. The total normal component of discharge would match (total normal discharge in the two yellow domains would match the total normal discharge of the four green domains), but that normal discharge may be distributed oddly between different levels (e.g. a lot of flow from the lower yellow level going into the uppermost green level). A better solution would be to use two inter-domain boundaries:

1. one with the lower yellow domain on one side and the two lowest green domains on the other, and

2. one with the upper yellow domain on one side and the upper one or two green domains on the other. This way, there can be vertical gradients between at the yellow/green interface, and upper-level flows are matched with each other and lower-level flows are matched with each other.
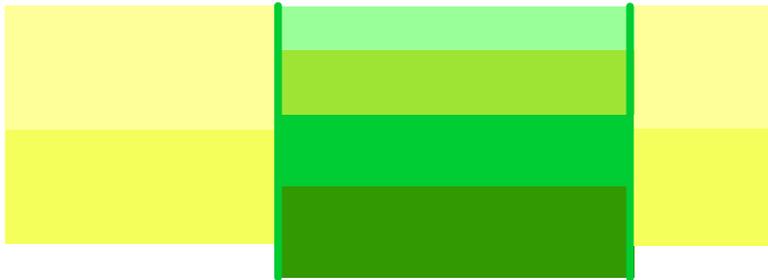
2.7   Pathline Tracing

Pathlines are traced in the horizontal plane using the aquifer discharge vector function in the domain and a numerical tracing procedure outlined in section 26 of (Strack, 1989). Three-dimensional pathline tracing is done using a finite-difference form of equation 24 in the paper by Strack (1984), which uses three-dimensional flow continuity to approximate the vertical component of pathlines. For transient models, pathlines are traced through a transient flow field that evolves through the duration of the transient simulation (this is new in release 2016-2; earlier releases traced pathlines through the flow field of the last time step in a transient model). From 2016-2 on, pathlines in transient models start at the specified start time, and can be traced backward to the

beginning of the simulation or forward until the end of the simulation. In steady models, pathlines are traced forward or backward until they either exit the model or until the specified total time is reached.

Where pathlines cross inter-domain boundaries, the elevation of the pathline is adjusted so that the fraction of the normal component (perpendicular to the boundary segment) of flow above and below the pathline on each side of the inter-domain boundary match. For example, say you have an inter-domain boundary with one domain on the left and three on the right. If the pathline comes to the boundary from the left with 45% of the normal component of discharge above the pathline and 55% below it, the domain and elevation of the pathline side will be determined so that this 45/55 ratio is preserved on the right side as well. There is typically a jump in the elevation of the pathline as it crosses an inter-domain boundary for the following reasons:

- The elevations of the domains on left and right do not necessarily match

- Where there are multiple levels on one side of the boundary, the distributions of the normal components of flow vary from one level to another due to K and flow field variations

- The total normal components of flow match only approximately from left to right side (see Fitts (2010)). AnAqSim is not matching these normal components of flow perfectly at every point on the boundary - it is matching the total normal discharge across segments of the boundary perfectly.

When the normal discharge components across an inter-domain boundary are in different directions in different levels (usually where flow is nearly stagnant), this algorithm breaks down, and under such conditions pathlines may terminate at the inter-domain boundary.

When pathlines intersect an internal line-sink boundary such as a river boundary, there are two possibilities: 1) the pathline could be consumed by the linesink and terminate, or 2) the pathline could continue on underneath the linesink. Whether option 1 or 2 occurs is determined based in the elevation of the pathline as it approaches the linesink, the discharge/length of the linesink at the intersection, and the normal component of the domain discharge at the intersection. It is assumed that the linesink is at the top of the domain and draws its water from the upper portion of the domain discharge. If the result is case 2 with the pathline continuing under the linesink, its elevation will generally jump as it crosses under the linesink, gaining elevation if the linesink is extracting water and losing elevation if the linesink is injecting water.

In multi-level areas of models, pathlines can cross from one level to another vertically.

Where pathlines exit the bottom or top of the model due to recharge or leakage (e.g. at the water table for pathlines traced upstream), AnAqSim draws a circle at the end of the pathline and if you move the cursor over the circle, it will display elapsed time, domain, elevation, etc.

3   Computational Limits and Efficiency

There are limits imposed by the amount of memory that a computer can address and these can limit the maximum size of an AnAqSim model. The following graph shows data from numerical experiments with 64-bit AnAqSim on a number of different computers. These data are specific to these machines and each computer will behave differently depending on what other software is running and how the computer is configured, but this at least gives rough guidance. The vertical axis shows the RAM used by the AnAqSim task - you will need more than this amount of memory to handle the operating system and possibly other tasks. Thanks to staff at McLane Environmental for providing several of the data points in this graph. When the number of equations approaches the upper limit posed by memory on a given computer, the solve times increase significantly.   If, while running a model, all available memory (RAM) is used by AnAqSim and other tasks running, the windows operating system and .net will swap pages of memory to/from disk, which is a much slower than using RAM. Symptoms of this condition are a dramatic slowing of all tasks and lots of hard disk activity; this is generally a condition to avoid. It is best to increase your computer's memory or reduce the complexity of the model so that you can make runs entirely within RAM.
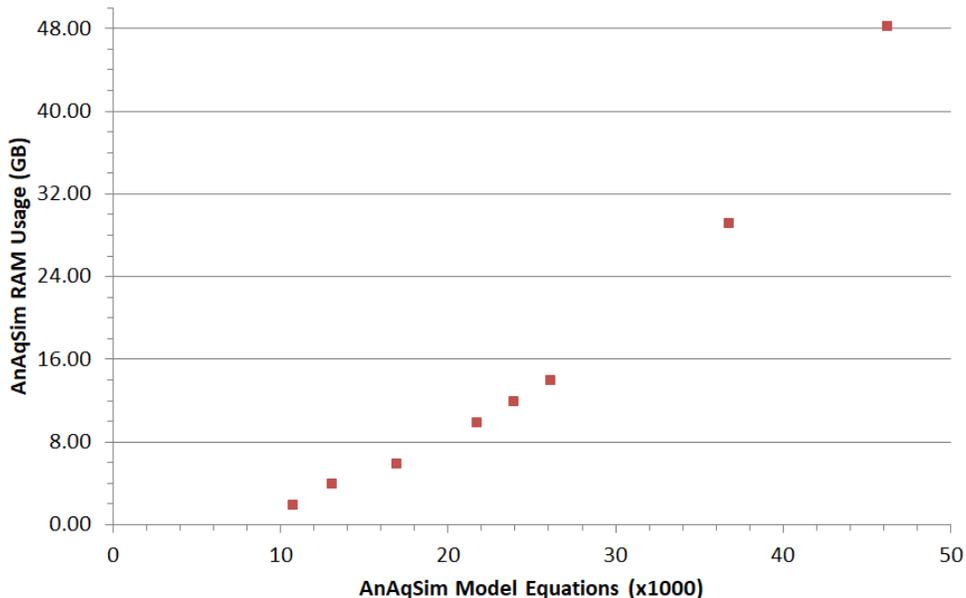
Another potential limit to the application of AnAqSim is where the system of equations gets so large that it takes too long to solve in a reasonable time.   That limit on present standard personal computers is probably in the range of a several tens of thousands of equations. AnAqSim can be the right tool for problems of moderate complexity, but not extreme complexity. Computing power evolves rapidly, and these limits will migrate

upward with time. For situations with extreme complexity, a numerical method like finite differences or finite elements is probably a better choice.

Using multi-level areas in a model adds rapidly to the number of equations in the system. To keep your problem within a manageable number of equations, use multiple levels sparingly, just in your area of interest and transition to a single level in the far-field.

Some additional suggestions for efficient modeling and limiting computation time are given in the *Solve* section.

4   Parallel Processing for Plot Computations

Starting with release 2019-1, parallel processing was implemented for these computations that occur in plot-making:

• Computing values (e.g. head) that are the basis for contours

• Computing values (e.g. velocity) that are the basis for vectors

• Tracing multiple pathlines from lines, areas, wells, and circles

This can be a time-saver for complex models and plots, if your computer's CPU has multiple cores. Experiments indicate that making a plot with hundreds of area pathlines is quicker by a factor approaching 4 on a 4-core CPU, or approaching a factor of 6 on a 6-core CPU.

5   Automating AnAqSim Runs from the Command Line

In addition to running AnAqSim interactively through the graphic user interface, you can make automated runs using the Windows command prompt. This may be useful for running parameter estimation software such as PEST or for executing a batch of unattended runs. The command prompt is accessed in Windows 7 by clicking Start Button -> All Programs -> Accessories -> Command Prompt. Once in the Command Prompt, the command syntax is:

path_to_AnAqSim   path_to_input_file   path_to_run_control_file

There are two ways to approach this:

1. Type in the full path for every input/output file; or

2. Change the current directory of the command prompt at the start, then just type in the file names (no path needed) for every input/output file, which should be in the current directory.

The second approach is a bit cleaner and simpler.

Assuming you've changed the current directory in the command prompt to the directory with all input/output files, the following is an example command that would execute a run of AnAqSim:

"c:\program files\fitts geosolutions\anaqsim\anaqsim.exe" run23.anaq run23.txt

It is necessary to put quotes around the path when the path contains names with spaces such as "program files" or "my documents". In the above example, AnAqSim is launched, *run23.anaq* is opened from the command prompt current directory, then execution and outputs are controlled by lines in the run control file *run23.txt,* also in the command prompt current directory.

The run control file is a text file that you create with a text editor like the *Notepad* accessory that comes with Windows operating systems. The following is an example of the contents of a run control file:

> *initialheadsfile run23.hds*
> *outputfile run23.out*
> *headspecifiedwells*
> *dischargespecifiedwells*
> *headspecifiedlines*
> *rivers*
> *calibration*
> *externalboundaries*
> *verticalleakage*
> *capturedpathlines*
> *headoutputfile run23outputheads.csv*
> *exit*

The first line *initialheadsfile run23.hds* tells AnAqSim to use file *run23.hds* as the initial heads for a transient run. As written with no path, *run23.hds* must be located in the command prompt current directory. If it's located elsewhere, you must write the entire *path+filename* here instead of just *run23.hds*. The two parameters on this line must be separated by any combination of spaces and/or tab characters. This line is only needed for transient runs where the starting heads are to be read from a file.

The second line *outputfile run23.out* tells AnAqSim to write text outputs from the automated run to a new text file named *run23.out*. As written with no path, *run23.out* will be written to the command prompt current directory. If you want it written elsewhere, you must enter the entire *path+filename* here. Again the two parameters on this line must be separated by any combination of spaces and/or tab characters and the file name should include the path to the file.

The 3rd through 7th lines direct AnAqSim to write certain outputs to the output file after having solved the model:

headspecifiedwells directs AnAqSim to write the discharges of head-specified wells to the output file. If this line is omitted from the run control file, these will not be written to the output file.

dischargespecifiedwells directs AnAqSim to write the heads at discharge-specified wells to the output file, and if there are multi-level discharge-specified wells in the model, the breakdown of discharges by domain for those multi-domain wells is also written to the output file. If this line is omitted from the run control file, this information will not be written.

headspecifiedlines directs AnAqSim to write the discharges of internal head-specified line boundaries to the output file. If this line is omitted from the run control file, these will not be written.

rivers directs AnAqSim to write the discharges of river line boundaries to the output file. If this line is omitted from the run control file, these will not be written.

calibration directs AnAqSim to write the calibration to the output file. If this line is omitted from the run control file, this will not be written. All types of calibration information is written.

external boundaries directs AnAqSim to write the discharges across external boundaries to the output file. If this line is omitted from the run control file, this will not be written.

verticalleakage directs AnAqSim to write the vertical leakages over polygon areas to the output file. If this line is omitted from the run control file, this will not be written.

capturedpathlines directs AnAqSim to write the number of displayed pathlines to the output file. With an automated run, the plot is not actually drawn, but the pathlines are traced in memory, and the number that would have been displayed is totaled.   This is

useful for running parameter optimization problems where you want to maximize or minimize the number of pathlines captured by specified wells or line boundaries.   Use this with capture constrain in [pathline settings](#).

headoutputfile run23outputheads.csv directs AnAqSim to evaluate heads according to the current plot settings (*Plot Input / What to Plot / Window* and *Plot Input / Contours Settings / Points_Evaluate*) in each level and at each time step if it is a transient simulation. These heads are written to a comma-delimited text file named *run23outputheads.csv which* will be written to the command prompt current directory.   If you want it written elsewhere, you must write the entire *path+filename* here. This file contains header lines that list the *level*, and for a transient run the *time period, time step, time*. The data following each header appears as lines listing: *x, y, head*.

exit directs AnAqSim to close after having solved and written the specified outputs. If a line with "exit" is not in the control file, AnAqSim will remain open so that the user could continue with other ways of analyzing the solution or making plots.

All of the commands listed in the run control file are lower case (case sensitive). The lines may appear in any order.

6   User Interface

The user interface consists of one main menu plus three different tabs:

- Plot: for graphically displaying map-based inputs and outputs

- Data: for editing input data. The data view is automatically displayed when you chose to edit something under the *Model Input, Plot Input*, or *Analysis Input* menus.

- Log: for displaying text outputs. The run log displays information when files are opened or saved, when the *Solve* menu is executed, and in response to a number of choices under the *Analysis* menu.

More information about each of these tabs is given in the following sections. Change from one tab to another by clicking on the tab at the left. The current active tab is highlighted in blue. In the following image, the plot tab is selected.
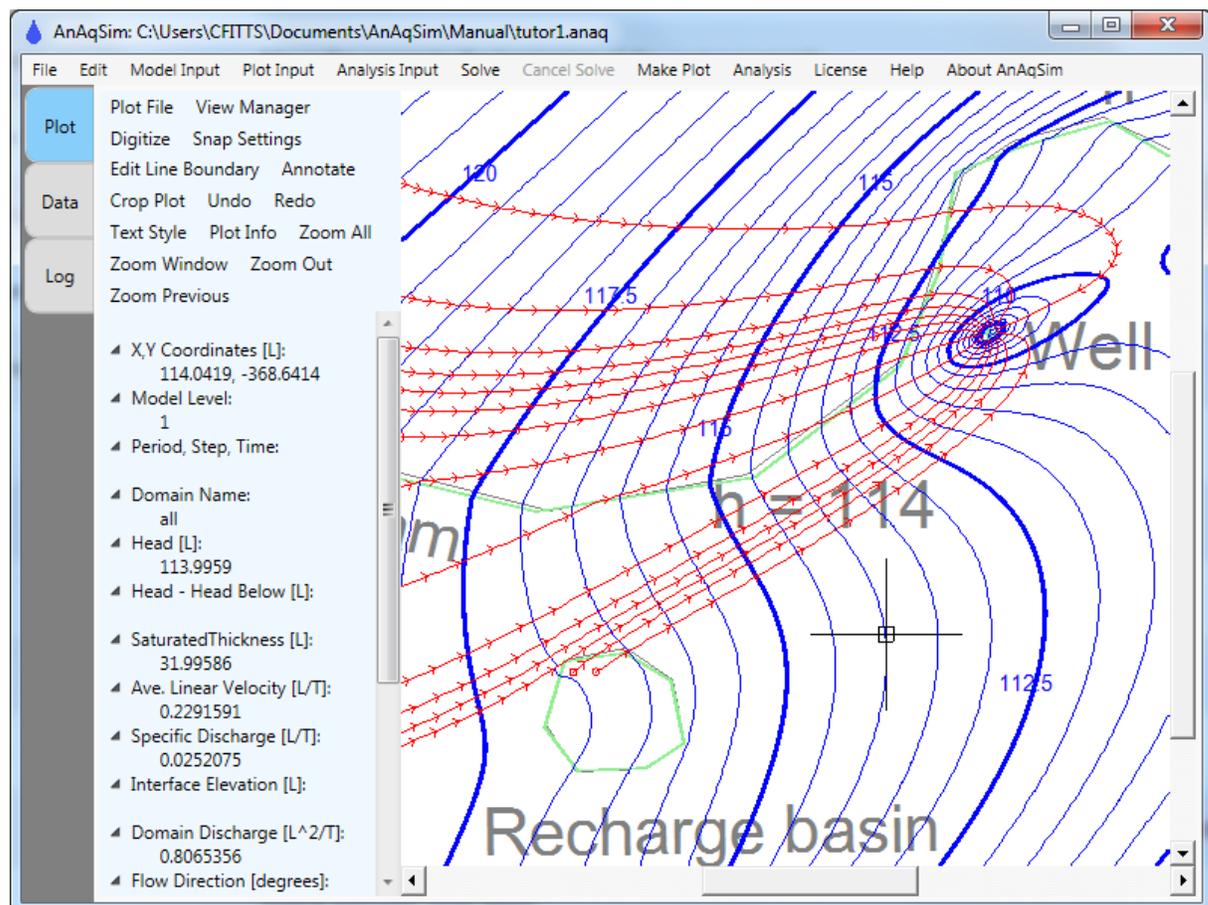
6.1   Plot Tab

The plot tab shows the model inputs and results plotted in map view. This view is automatically shown after opening an existing model, and after making a plot under the

Make Plot menu. Most of the plot view is a map view of the model that can display a basemap, model elements, simulated heads, flow vectors, pathlines, etc. The plot area has scroll bars that allow you to shift the view left/right and up/down while retaining the same scale.   The scroll wheel on most mouse devices will cause the view to zoom in and out. Pressing down on the scroll wheel and moving the mouse will allow you to pan the view with most mouse devices.
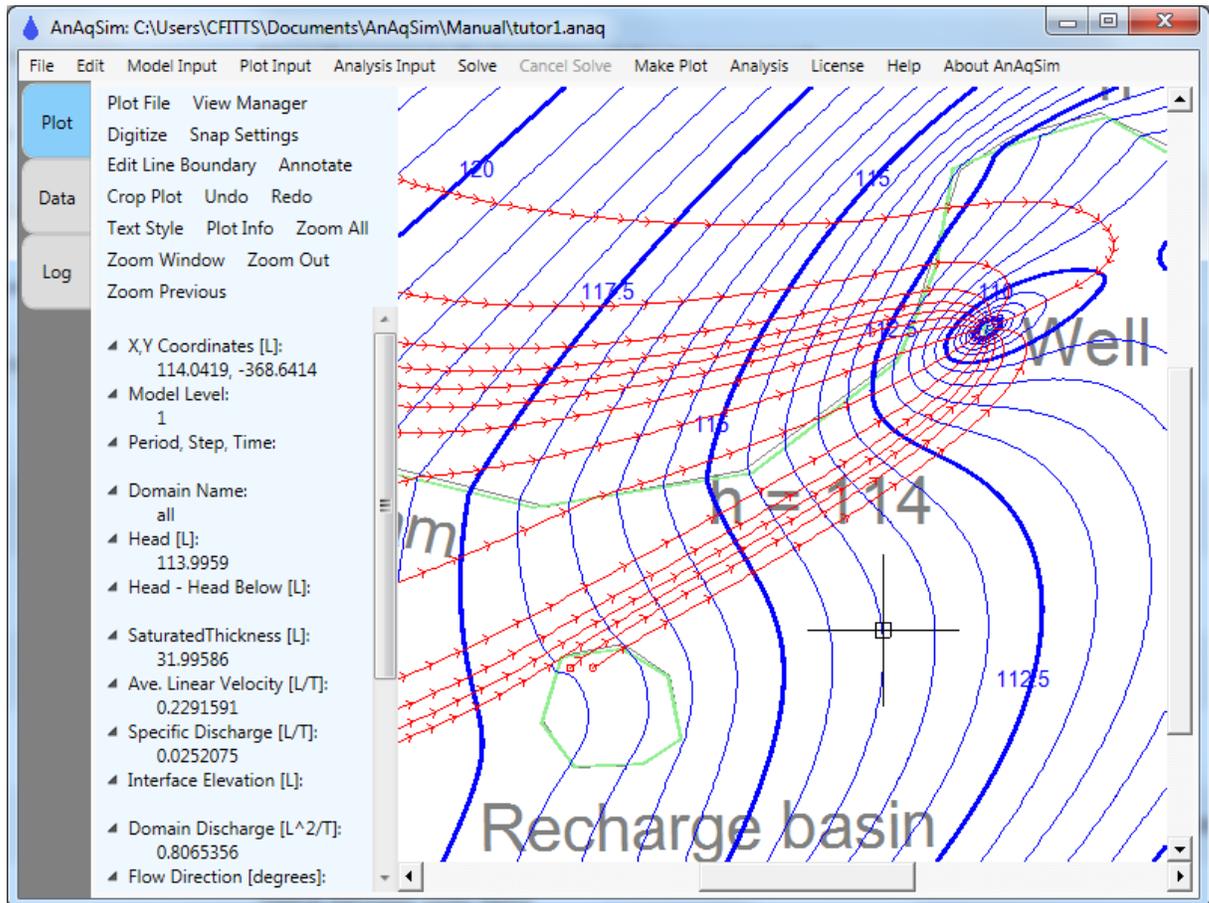
On the upper left is a separate plot view menu that applies just to the plot (Plot File, View Manager,...Zoom Previous). Choices in this menu allow you to zoom to a different view, save or print the plot, digitize and edit coordinates, or add annotations to the plot. See the tutorial videos at the website to see the various functions in the plot view menu explained and demonstrated. Right-clicking your mouse over the plot brings up a context menu with many choices for digitizing, editing line boundaries, and generating outputs related to the cursor location.

6.1.1 Context Menu (right click over plot)

When the cursor is over the plot, you may bring up a context menu by clicking your right mouse button. This menu allows the following options, which are handy short cuts while building, editing, and analyzing a model:

- *Edit Domain Properties* - switches to the data view with a domain data table displayed and one highlighted row: the row of the domain where the cursor is located.

- *Edit Nearest Well* - switches to the data view with a well data table displayed and one highlighted row: the row of the well nearest the cursor location.

- *Edit Nearest Line Boundary* - switches to the data view with a line boundary data table displayed and one highlighted row: the row of the line boundary nearest the cursor location.

- *Edit Area Sink that Applies Here* - switches to the data view with an area sink data table displayed and one highlighted row: the row of the area sink that applies at the cursor location.

- *Digitize* - includes the most commonly used commands (Point, Polyline, Clear Digitizing Marks) from the Plot View/Digitize menu, but without the pop-up instructional windows. These are efficient for experienced users.

- *Edit Line Boundary* - allows the same operations (Insert Vertex, Delete Vertex) as are found in the Plot View/Digitize menu, but without the pop-up instructional windows. These are efficient for experienced users. One line boundary, spatially-variable area sink (SVAS) polygon boundary, or vertical leakage polygon must be selected before executing either of these menu items.

- *Set Plot Window to Current View* - this is the same as Plot Input/Set Plot Window to Current View, setting the window for subsequent plots to the current view.

- *Set Plot Window to Entire Model* - this is the same as Plot Input/Set Plot Window to Entire Model, setting the window for subsequent plots to the entire model.

- *Check Nearest Well Head and Discharge* - this writes the head and discharge of the nearest well to the Log tab.

- *Check Nearest Line Boundary Condition* - This allows you to check the accuracy of the approximation of boundary conditions along the particular segment of a line boundary that is closest to the cursor when this is selected. This causes a graph to be made of the conditions on that boundary segment. The graphs vary depending on the type of line boundary. You may export a bitmap graphic of the graph or the underlying data (see the *Exporting X-Y Graphs* topic). Prior to release 2016-2, this feature was under the Analysis menu.

- *Graph Head Hydrograph(s) Here, All Levels* - for a transient simulation, this causes hydrographs (graphs of head vs. time) to be made that show heads in all model levels at

the location of the cursor. These graphs do not contain the initial heads (time=0) at these locations. If you want hydrographs that include initial heads, use Analysis Input/Hydrograph Points and Analysis/Graph Head Hydrographs and Analysis/Graph Drawdown Hydrographs.

6.1.2   Data at Cursor Location

To the left of the plot and below the plot menu is an area that displays the coordinates of the cursor and model results (X,Y Coordinates, Model Level, Domain Name, Head,...) at the location of the cursor. As you move the cursor, these data are updated for the cursor location. Each item may be hidden or visible; to toggle this, click on the triangle to the left of the label. If the window is not tall enough to display all items, use the scrollbar immediately to the right of this area. In transient models, the values reflect the time step of the selected period, which ends at the time listed. Head and interface values are at the end of the time step at the time listed. Discharge-related values apply over the duration of the time step listed.

- *Head Above - Head* is the difference in head from the level above to the level of the plot.

- *Head - Head Below* is the difference in head from the level of the plot to the level below.

- *Interface Elevation* is the elevation of the fresh-salt interface (only applies in models with fresh-salt interface domains).

- *Domain Discharge* is the specific discharge times the saturated thickness, which equals the discharge in the domain per unit length normal to the discharge.

- *Flow Direction* is the direction of the specific discharge or average linear velocity vector, assuming the positive x axis is zero degrees.

- *Top of Model Condition* is the condition at the top of the topmost domain of the model at this x,y, defined by the area sink input.

- *Bottom of Model Condition* is the condition at the bottom of the bottommost domain of the model at this x,y, defined by the area sink input.

- *Modeled Extraction* is the extraction per area γ [L/T] that AnAqSim is modeling at the point, using the spatially-variable area sink functions. This quantity is defined by equation 13 of Fitts (2010).

- *Extraction from Heads* is defined as $L_t + L_b + S\,\partial h / \partial t$, where L_t is leakage out the top of the subdomain computed using head differences and vertical hydraulic conductivities, L_b is the leakage out the bottom of the subdomain computed similarly, and $S\,\partial h / \partial t$ is the transient discharge/area into storage computed using the head change over a time step and storativity. At spatially-variable basis points, this equals the modeled extraction, but between basis points, the two are unequal, but close if the basis point spacing is small enough. This quantity is defined by equation 6 of Fitts (2010). In a

transient simulation, this cannot be computed for the first time step, since the initial head at time zero at the cursor location is not known and therefore ∂ h can't be computed.

- *Leakage out Top* is the leakage out the top of the domain, L_t, as defined above [L/T].

- *Leakage out Bottom* is the leakage out the bottom of the domain, L_b, as defined above [L/T].

- *Storage Flux* is the flux into storage in a transient model, S ∂ h / ∂ t, as defined above [L/T].

- *Leakage Factor Above* is the leakage factor (LF) computed for upward leakage from the subdomain. If the subdomain is at the top of the model with no overlying level, the leakage factor is computed as LF = (T b / (2 * K_v))$^{1/2}$, where T is the transmissivity of the subdomain, b is the saturated thickness of the subdomain, K_v is the vertical hydraulic conductivity of the upper half of the subdomain. If the subdomain has another subdomain overlying it, the leakage factor is computed with the following equation where T_a is the transmissivity of the subdomain above, b_a is the saturated thickness of the subdomain above, and K_{va} is the vertical hydraulic conductivity of the lower half of the subdomain above. Leakage factors are used as guidance for determining appropriate basis point spacing for spatially-variable area sinks. See the discussion under spatially-variable area sinks.

$$LF = \{ T T_a [b / (2 * K_v) + b_a / (2 * K_{va})] / (T + T_a) \}^{1/2}$$

- *Leakage Factor Below* is the leakage factor (LF) computed for downward leakage from the subdomain. It is computed in a manner analogous to that described above for *Leakage Factor Above*.

6.2   Data Tab

The data tab allows you to edit model inputs, plot inputs, and analysis inputs with a data grid component that displays data from the underlying data tables. To display and edit a table, make a selection under the *Model Input, Plot Input*, or *Analysis Input* menus.
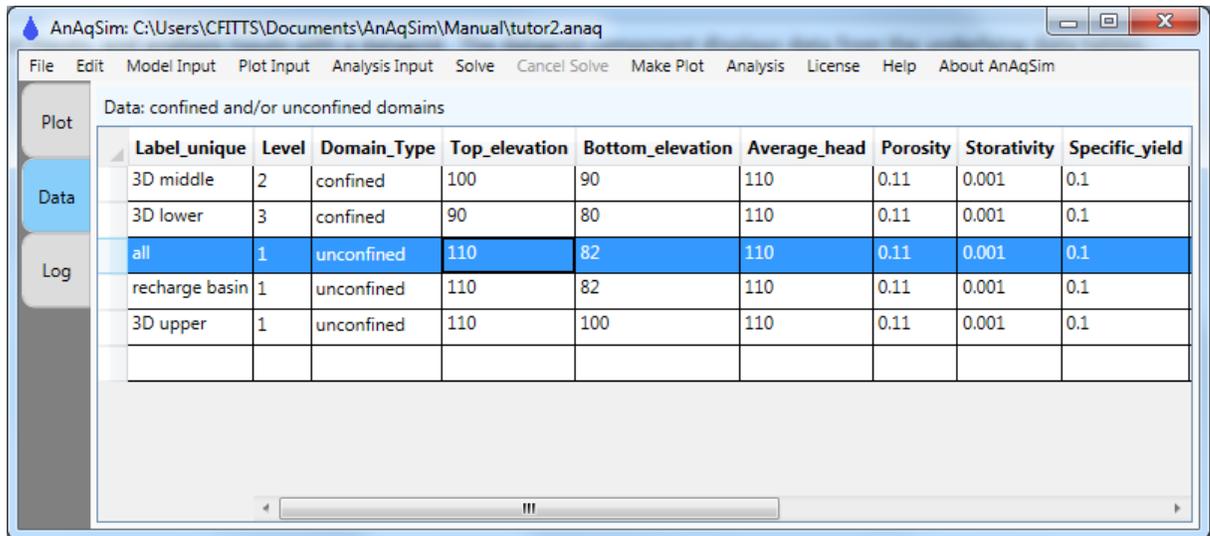
There is a context menu that pops up when you right-click over the grid. Options in this menu include

- Paste New Rows - this pastes in new rows of data that are tab-delimited between columns. This format allows you to paste data copied from spreadsheets like Excel.

- Copy Selected Rows - this copies the selected rows to the system clipboard, which can then be pasted into spreadsheets like Excel.

- Copy All Rows - this copies all rows in the spreadsheet to the system clipboard, which can then be pasted into spreadsheets like Excel.

6.2.1   Using the Data Grid

A data table is displayed in the grid when you select an item under the _Model Input_, _Plot Input_, or _Analysis Input_ menus. The displayed data is linked to one of several database tables, and when you edit the displayed data, the underlying data table is updated.

The table of data is displayed with headers that define each column, such as _Label, Domain, Parameters_per_line, ..._ as shown below.
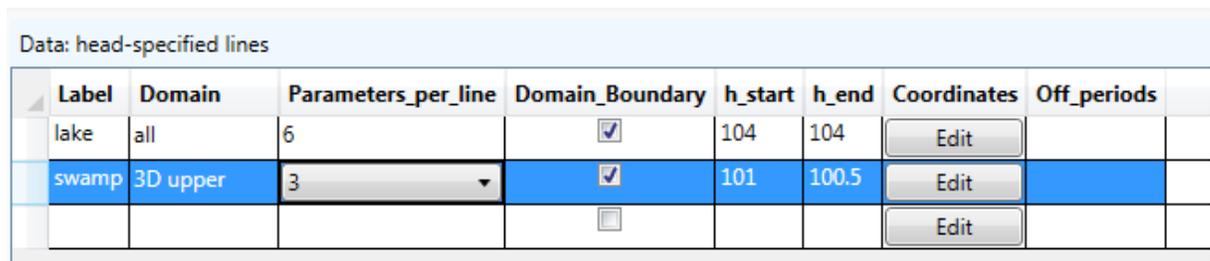
You can move from cell to cell with the arrow keys or using the mouse. The current row is highlighted blue. You can enter new values by navigating to a cell and typing a new entry, or you can double-click on a cell to edit cell contents with a text editor, as shown below, where the contents of the cell with "101" is being edited.

omain_Boundary	h_start	h_end	Coordinates	C
☑	104	104	Edit	
☑	101	100.5	Edit	
☐				

When you enter a value in a grid cell, the underlying database is updated when you press enter or move to a different cell. At this point the cell value is checked to make sure it is compatible (e.g. a positive real number for hydraulic conductivity). If the value is incompatible, an error message is displayed and you must correct the cell entry. Be sure to remember to press enter or move to another cell after editing the value in a cell, otherwise the value will not be changed in the database.

New rows are created by editing the blank row at the bottom of the table. A new row of data is entered into the underlying data table only when you press enter after editing the row, at which point a new blank line appears below the line just entered. The following two screen shots shows a new 2nd row before it has been entered in the database (no blank row shows below it), and after (blank row below 2nd row).

Data: head-specified lines

Label	Domain	Parameters_per_line	Domain_Boundary	h_start	h_end	Coordinates	Off_periods
lake	all	6	☑	104	104	Edit	
swamp	3D upper	3	☑	101	100.5	Edit	

Data: head-specified lines

Label	Domain	Parameters_per_line	Domain_Boundary	h_start	h_end	Coordinates	Off_periods
lake	all	6	☑	104	104	Edit	
swamp	3D upper	3	☑	101	100.5	Edit	
			☐			Edit	

In data tables that contain multiple rows, the leftmost field is often called *Label,* and it is always displayed even if you scroll far to the right.   No entry is required in this field, and it accepts any text. It is wise to fill in a text label in this field (e.g. "PW-103" for a pumping well). The label will help you know which feature this row represents, and many analysis outputs make use of this label. Also you can sort the data based on entries in this column to easily find the row you want. The contents of the table can be sorted by clicking on the column header. Clicking a second time reverses the sort order. It is a good idea to choose labels that easily allow you to sort features. For example, if you want to easily find a group of wells on property A, you could give them labels such as

"A_MW102", "A_MW105", "A_MW113"... so they would be grouped together after sorting by the label column.

Column widths are automatically adjusted to fit the contents. You can increase or shrink column widths by dragging the left or right the vertical line that separates columns in the header (top) row. Double-clicking on this vertical line automatically resizes the column width to fit the contents.

Some columns, like the *Parameters_per_line* column in the table shown above, are edited using a drop-down list of choices. To see the list, double-click the cell, then select the item you want.

Other columns, like the *Coordinates* column, contain buttons to edit or select data; these are edited by clicking on the button.

6.2.2   Number Formats

All data grid cells that expect numerical input have common format constraints. You can input real numbers with formats such as the following:

- 1256.82

- 0.004

- 1.4e-2

The last one is scientific notation for 1.4×10^{-2}.

You should not insert commas to mark thousands, millions, (e.g. 1,200,000) as the comma may be interpreted as a decimal mark. In North America, the convention is to use a period for the decimal marker. In Europe, the comma is often used as a decimal marker. There is a Windows operating system setting to switch between these modes. Often, European users need to adjust these settings to use AnAqSim.

6.2.3   Editing Coordinates

In many of the data input tables, there are columns and cells that display an "Edit" button in the *Coordinates* column. When you click the button, a text box window pops up and you enter coordinate data there:
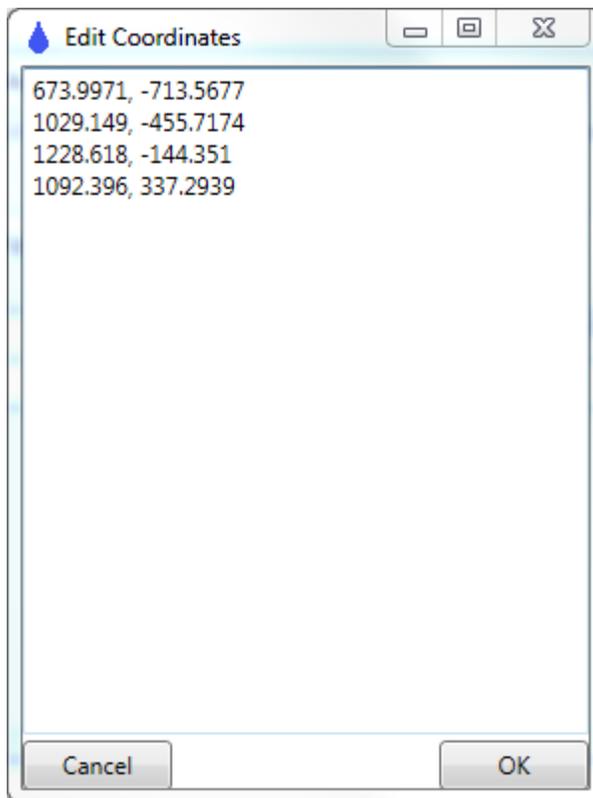
Often, you will digitize the coordinates in the plot tab and then paste the coordinates into this text box window. Alternately, you can just type coordinates in. The OK button records the edited coordinates and the Cancel button does not.

Once input, coordinates can be edited graphically by selecting the line boundary and then moving the vertexes or inserting or deleting vertexes.

6.2.4   Deleting Data Rows

Delete one or more rows of data in the data table by selecting rows and then pressing the Delete key.   Row(s) are selected by clicking (and dragging for multiple rows) in the leftmost column of the grid. A dialog will ask you if you really want to delete those records from the data table.

6.2.5   Importing and Exporting Data

To import data from Excel into a data table, highlight a block of data in an Excel sheet that corresponds to row(s) of data in a data table, copy that block in Excel, then right-click over the data grid and select *Paste New Rows*. This will add these copied rows to the data table. Make sure that the columns in the copied block match the columns in the data table. Data in *Coordinates* columns cannot be be pasted in due to their multi-line structure, but all other columns can be pasted in. In the case of a *Coordinates* column, a

paste operation leaves that blank and you must enter the coordinates by clicking on the *Edit* button in that column.

To export rows of data to Excel or a text file, select rows of data (see section above) and then right-click over the data grid and select *Copy Selected Rows*. After doing this the rows of data are in the computer's clipboard as tab-delimited data, which can then be pasted into Excel or into text files.

6.2.6   The Run Log

The lower part of the data view holds the run log, which is an area that displays text output from the program. The run log continues to accrue more text as you execute various tasks such as updating license information, opening a file, solving the system of equations, checking boundary conditions, checking calibration results, or closing a file. If the text in the run log gets long enough, a scroll bar will appear to let you scroll through the entire log. You can select all or a portion of the text in the run log and cut, copy, and paste this text. This is an easy way to move text results to another document.

6.3   Log Tab

The Log tab holds the run log, which is an area that displays text output from the program. The run log continues to accrue more text as you execute various tasks such as updating license information, opening a file, solving the system of equations, checking boundary conditions, checking calibration results, or closing a file. If the text in the run log gets long enough, a scroll bar will appear to let you scroll through the entire log, as shown below. You can select all or a portion of the text in the run log and cut, copy, and paste this text. This is an easy way to move text results to another document.
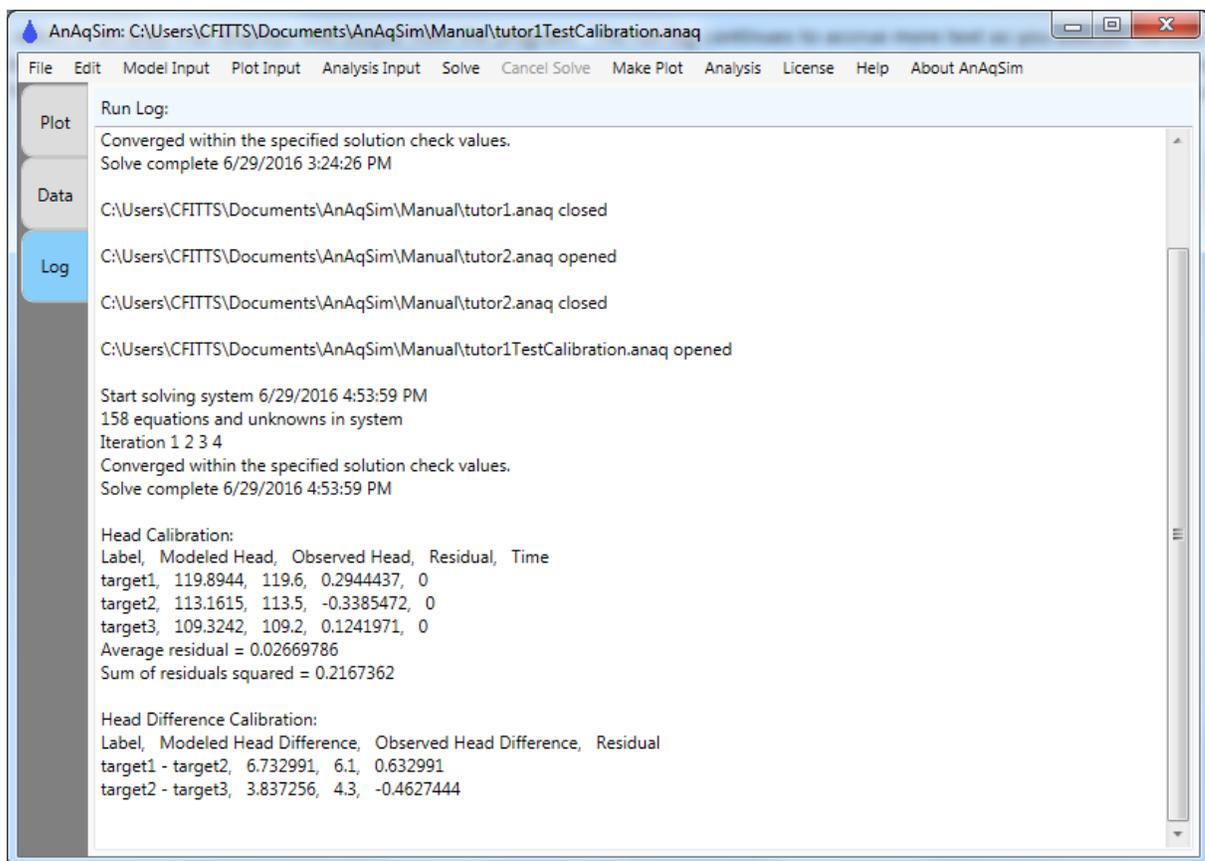
6.4   General Modeling Sequence

Creating a model follows this general sequence:

1. You can start a new model either right after starting the program or after selecting *File/Close* which closes the current input and begins a new input data set. Once either of these steps is taken, you may edit data tables under the *Model Input, Plot Input, or Analysis Input* menus.

2. Create model input using the *Model Input* menu. **Make sure to define *Domains* before adding *Well, Line Boundary,* or *Area Source/*Sink elements.** This sequence is necessary because the input for the elements includes specification of the domain(s) they are in. When adding elements, it helps a lot to use a basemap and digitize coordinates on top of the basemap.

3. Define what you want displayed in plots with the *Plot Input* menu.

4. Define what analysis features you want with the *Analysis Input* menu.

5. Save your model frequently as you build your input.

6. When the model input is complete, select *Solve* to solve the system of equations. This is required after making any model input changes and before making output plots or using the *Analysis* menu.

7. View plots of the model results with *Make Plot*.

8. Examine model results with the *Analysis* menu.

9. Loop back through steps 2-8 to revise the model, re-solving the system after revision and before examining results.

6.5   Menu Keyboard Shortcuts

You can access common menu items with keyboard shortcuts by pressing the key sequences as listed below.   Many are standard Windows shortcuts.

Ctrl-O   File/Open

Ctrl-S   File/Save

F12   File/SaveAs

Ctrl-W   File/Close

Alt-FE   File/Exit

Alt-S   Solve

Alt-PA   Make Plot/All Selected
            Features

Alt-PE   Make Plot/Elements Only

Alt-V   Switch View

Alt-H   Help

7   File Menu

This menu allows you to open and save input files and create files that are used for creating initial head conditions for transient models.

7.1   Open

This selection opens a dialog that allows you to find and open existing input files (**.anaq** extension). These files store the data you edit under the *[Model Input](#), [Plot Input](#),* or *[Analysis Input](#)* menus in XML file format. XML is a common ASCII database file format. You could edit these directly with a text or XML editor, but that is not recommended since it risks corrupting input with improper values or format. When you open a file, the layout of elements in the model is drawn to the plot view.

If you want to be able to open .anaq files by double-clicking on them in Windows Explorer, in addition to opening them from the *File/Open* menu, the Windows operating system must associate .anaq files with AnAqSim. In case this association was not established during installation you can manually do it with Windows Explorer. To do this, locate a .anaq file in Windows Explorer. Right click on the file and then select *Open with,* then select *Chose default program.* In the dialog that pops up, check the box next to *Always use the selected program to open this kind of file* and then select the *Browse* button and browse to find the AnAqSim.exe file in the Program Files / Fitts Geosolutions / AnAqSim software directory. Now Windows will associate .anaq files with AnAqSim.exe, and you can open any .anaq file directly from Windows Explorer by double-clicking on it.

7.2   Save, Save As

The *Save As* option brings up a dialog that allows you to save your input to a file with a new name. Using *Save* saves the input to the same file name. If you have yet to save input and have no filename, it will function like *Save As*. When you save, you save the input data tables to an XML format database file with the **.anaq** extension.

7.3   Close

This closes the input you are working on and clears all the associated data tables in memory. After selecting *Close*, you may begin editing a new model.

7.4   Save Locations for Initial Transient Heads

A transient model needs initial heads so it can compute the head change that occurs during the first time step. These values are needed at the location of each basis point in each spatially-variable area sink, which account for storage fluxes. The initial head values come from a pre-existing model, which could be steady or transient. Initial heads are also

retrieved for discharge-specified wells, hydrograph points, and transient line conditions (see *Analysis Input Menu* for the last two items).

The initial conditions model must have the same number and extent of layers as the transient model, at least in areas with basis points, wells, hydrograph points, or transient line condition lines. When the initial head locations are written, each location is identified by its x,y coordinates and its layer. This is new in release 2020-1; prior releases wrote the x,y coordinates and the internal domain number. The change made in release 2020-1 allows different domain configurations between the initial and the transient models, which can be helpful. Because of this change, you must not mix initial head location or initial head files created prior to release 2020-1 with a release 2020-1 or later model. To avoid incompatibility when you switch to release 2020-1, re-create the initial head location file and the initial heads file as outlined below using release 2020-1.

To create a transient model that has a proper set of initial heads, these steps are necessary:

1) Make sure the transient model you begin to create has been saved to it's own unique file name, different from the file that contains the input for the model that will provide the initial condition heads.

2) Set up the transient model (i.e. uncheck Steady under *Model Input/General*, establish the time step sequence under *Model Input/Time Steps*, adjust input for the transient case, set up all spatially-variable area sinks, make the appropriate settings under *Analysis Input/Hydrograph Points* and *Analysis Input/Transient Line Conditions*, etc.

3) Select *File/Save Locations for Initial Transient Heads*, which saves the locations for transient starting heads from the transient model (these are the locations of all basis points associated with spatially-variable area sinks in the transient model and locations of hydrograph points, wells, and transient line conditions. This saves the level and the coordinates of each of these to a binary file with the .ihl extension.

4) Close the transient model.

5) Open the initial conditions model (the one that represents conditions at the start of the transient run). Solve it. *Select File/Write Initial Transient Heads,* click on the initial heads file you would like to use for this simulation, and click Open. This reads in the locations saved from the .ihl file saved in step 3.

6) A dialog opens asking you to name the binary initial heads file. The default name is the same as the initial conditions input file but with the .hds ending. AnAqSim then writes the initial heads at the locations in the .ihl file to a binary file with the .hds file extension.

7) Close the initial conditions model.

8) Open the transient model. After checking that all model parameters are set correctly for the transient run, select Solve. At this point AnAqSim will ask you to select the .hds file containing the initial heads created in step 6.

When you solve the transient model, the heads are read in and used to determine the head change in the first time step at each basis point.

7.5   Write Initial Transient Heads

See the discussion under *Save Locations for Initial Transient Heads* for an overview of setting up initial heads for transient models.

7.6   Save Solution

This allows you to save the model solution after you have solved. Later, you can open the model input file, then load the saved solution, and avoid the "Solve" step. This is particularly handy for large models that have longer solve times, and allows you to save your solution and come back to it later for making plots or doing analysis of the solution. All model objects with their strengths are saved in a binary format, to a file that has the same name as the input file, but with the ".solu" filename extension instead of ".anaq".

7.7   Load Saved Solution

This allows you to load in a previously saved model solution. To make use of this, first open the model input file for the model, then load the saved solution, which avoids the need for the "Solve" step. This is particularly handy for large models that have longer solve times, and allows you to save your solution and come back to it later for making plots or doing analysis of the solution. All model objects and their strengths are read in from a binary file that has the same name as the input file, but with the ".solu" filename extension instead of ".anaq".

7.8   Export Input Data to Excel File

This causes the entire input database to be written to one excel file in the same directory as the input file (*.anaq), written to a file with the same name but the excel suffix (*.xlsx). The Excel file has multiple sheets, one for each data table. Each sheet contains the same headers as the data tables plus all rows of input. This is a handy way to document model inputs all in one readable file.

7.9   Exit

This exits AnAqSim. A dialog asks if you want to save the current input before exiting. The same is achieved by clicking on the red "x" at the upper right corner of the AnAqSim window.

8   Edit Menu

This is like edit menus in most other Windows applications with *Cut, Copy*, and *Paste* menu choices. These functions are also available with the usual keyboard shortcuts: control-x (cut) control -c (copy) control -v (paste).

9   Model Input Menu

This menu allows you to edit data that defines the aquifer flow model. Input may be done in any order except that domains must be defined before adding wells, line boundaries or area source/sinks, since the latter are associated with a particular domain or domains.

9.1   General

This item has only one line of input. The first item is checked if the model is steady-state and not checked if the model is transient. To simulate storage fluxes in transient models, spatially-variable area sinks (SVAS) are required. If you try to solve a transient model without any SVAS, AnAqSim gives an error message.

The other three items are text values that document the length and time units used in the model, and provide comments to document the run. The model uses consistent length and time units throughout. For example, if you chose meters and days, then hydraulic conductivity, specific discharge, and average linear velocity are in m/day, well discharges are in m^3/day, and time markers on pathlines are in days.

9.2   Solution

The two data tables under this menu define settings involved in solving the system of equations in your model.   The first defines parameters involved in solver iterations, and the second lays out the solution accuracy needed before iteration ceases.

9.2.1   Solve Settings

- *Underrelaxation* is a parameter that governs how unknown strength parameters are updated at each iteration. If underrelaxation is 1.0, the new strength parameter equals the parameter from the most recent iteration. If this value is 0.7, the new strength parameter is weighted 70% by the parameter at the most recent iteration and 30% by the parameter at the previous iteration. Lower values help damp out oscillations in parameter values from iteration to iteration and may improve convergence in some situations. Higher values close to 1.0 speed convergence when oscillation is not a problem. A good range for most cases is 0.9 to 1.0.

- *Maximum Iterations*. When solving, iteration continues until the tolerances specified in *Check Settings* are met at all boundary condition control points or basis points, or if those criteria are not met, iteration ceases when this maximum number of iterations is reached.

- *Starting Heads*. For transient runs, the first time step needs initial heads from some source. There are two possibilities here:

1. Assign a constant value of head for each domain. The starting head at a point is set to the domain's average head, which is specified under *Domains* input. In the case of a *Discharge-Specified (Multi-Domain)* well, the starting head is set to the average head of the first domain listed. This option should only be used for very simple transient runs where the initial condition is a uniform, flat, constant head.

2. Read initial heads from a file. This file is written by the initial (time zero) model, which is another AnAqSim model of the same area that may be steady or transient (see discussion of this file under the *File* menu).

- *Almost_dry_fraction*. This parameter affects the solution in cases with unconfined, confined/unconfined, unconfined interface, and confined interface domains, where the heads can drop to low enough levels that the freshwater saturated thickness of the domain approaches zero. Instead of letting the domain have zero or negative saturated thickness, AnAqSim has a minimum saturated thickness. In an unconfined, confined/unconfined, or unconfined interface domain, the minimum saturated thickness = (average head - base elevation) * *Almost_dry_fraction.* In a confined interface domain, the minimum saturated thickness = (top elevation - bottom elevation) * *Almost_dry_fraction.* When heads drop to or below a level that would create this minimum saturated thickness, the aquifer is treated like a confined aquifer with this fixed minimum saturated thickness. This prevents domains from actually going completely dry and helps models converge despite portions of some domains approaching "dry" conditions. Setting *Almost_dry_fraction* to a low value means that very little simulated horizontal discharge will occur in "'dry" portions of the domain. Heads in these "dry"' areas will drop to unrealistically low levels (below the base of an unconfined domain, for example), and have little real meaning. When contour plots are made of head, these unrealistically low values are neglected. These low heads do appear in other outputs such as the panel to the left of the plot and in profiles. Using higher values of *Almost_dry_fraction* may help convergence by limiting the magnitude of head gradients in "dry" areas, at the expense of allowing more actual discharge in "dry" areas. You can check the amount of discharge in a "dry" area by using the *Analysis/Graph Conditions Along a Line* and making a profile of domain discharge along or across a line. Chose a low enough value of *Almost_dry_fraction* so the discharge in these "dry" areas is acceptably small.

- *Interface_leakage_option*. This specifies one of two options for computing vertical leakage at a spatially-variable area sink basis point where a fresh/salt interface is present. If unchecked (default), the vertical leakage is computed just like it is in all other cases: the vertical leakage rate is proportional to the difference in head from one level to the next. If this column is checked and an interface is present in the overlying layer, the head difference is computed by using the freshwater head that is at pressure equilibrium with static salt water at the base of the overlying layer; the head in the overlying layer is not used to compute the head difference. The head that is used to represent the overlying layer is the same as the head at the toe of the interface in the overlying layer. The unchecked mode is appropriate when the resistance to vertical

flow between levels is due to the K3 of the domains themselves. The checked mode is appropriate when the resistance to vertical flow between levels is due to an aquitard between the levels that is not explicitly modeled (in this case the resistance of the aquitard must be incorporated into the K3 values specified in the upper and/or lower domain). See Fitts et al (2015) for comparisons of these methods and discussion.

9.2.2   Check Settings

These settings define the accuracy of boundary conditions required of the solution; iteration continues until these conditions are met.   If when Solve is pressed the solution converged before reaching the maximum number of iterations, all boundary conditions were met within the tolerances specified here.

These settings also affect the function of *Analysis/Check Boundary Conditions at Control Points*, which is used to check how well the solution meets specified boundary conditions. Such conditions include heads at head-specified wells and linesinks, extraction at spatially-variable area sink basis points, etc. When you select *Analysis/Check Boundary Conditions at Control Points,* each boundary condition is checked, and if the discrepancy between the specified condition and the model-simulated condition is greater than a threshold you specify here, the program prints the discrepancy to the run log. If the discrepancy is less than the threshold, nothing is printed. In cases where the solution did not converge to within these tolerances during the *Solve* process, the offending boundary conditions are listed along with their accuracy. This can be a help to home in on which boundary conditions are holding up the *Solve* process.

Four kinds of boundary condition tolerances are defined as follows.

- *Head_check_tolerance* is the threshold for the magnitude of discrepancy between specified and modeled heads, used at head-specified wells and line boundaries.

- *Qn_check_tolerance* is the threshold for discrepancies in the computed discharge per length along River line elements. It units are discharge/length [(L^3/T)/L = L^2/T].

- *Q_check_tolerance* is the threshold for discrepancies in discharge, which are used along inter-domain line boundaries and normal-flux specified boundaries. In both cases, the condition being checked is the total discharge across a segment along the line [L^3/T]. For inter-domain boundaries, it is the comparison of discharges on opposite sides of the boundary. For normal-flux specified boundaries it is the difference between modeled and specified discharges based on the specified discharge per length times the length of the segment.

- *Extraction_check_tolerance* is the threshold for discrepancies in the modeled extraction (Equation 13 of Fitts (2010)) and the extraction computed from head

values (Equation 6 of Fitts (2010)). The units of extraction and this threshold are discharge/area $[(L^3/T)/L^2 = L/T]$.

9.3   Time Steps

This input is used only for transient models. If you are doing a transient model, make sure you uncheck *Steady* under *Model Input/General*. Each row of input in the Time Steps table defines a time period, during which all boundary conditions are constant. For example, a model could have three time periods with different recharge rates, river stages, or well discharge rates in each of the three periods, but within each period the values remain constant.

For each time period, you specify the total length of the time period (*Period_Length*), the number of time steps the period is divided into (*Steps_in_period*), and the time step multiplier (*Step_multiplier*). The multiplier causes the length of successive time steps to grow by a factor equal to the time step multiplier. The following table illustrates the lengths of time four time steps for a period that is 100 time units long, using various time step multipliers.

	Length of Time		
Time Step	Multiplier = 1.0	Multiplier = 1.5	Multiplier = 2.0
1	25.00	12.31	6.67
2	25.00	18.46	13.33
3	25.00	27.69	26.67
4	25.00	41.54	53.33
Total Time	100.00	100.00	100.00

In all cases, the total time of the period is 100, but the lengths of the four steps change as the time step multiplier changes. This scheme is the same as employed in MODFLOW. Using a multiplier larger than 1.0 helps concentrate computing power early in the time period when there is more transient change occurring. Transient storage fluxes, which are part of spatially-variable extraction, are computed for each time step using a finite-difference approximation of the governing equation (Equation 6 of Fitts, 2010).

9.4   Domains

The properties of each domain (called a *subdomain* in Fitts, 2010) are set with data tables under this menu. Different tables define the properties of different kinds of domains. A domain is a polygonal region of the model in a certain model level. Inside a domain, the aquifer properties (hydraulic conductivities, base elevation, storativity, porosity, etc) are homogeneous.

The boundary of a particular domain is defined by a combination of line boundaries that, in their input, are listed as external to the domain. Head-specified, normal flux-specified, and inter-domain boundaries can be external boundaries for domains. Other line boundaries like river and discharge-specified line boundaries are internal to domains and do not define domain boundaries. See the discussion under *Subdomains and Model Levels* for more detail and some examples.

All domain input data tables may be accessed through the main menu or by using a pop-up context menu when the cursor is over the plot.

9.4.1   Boundaries of Domains

The geometry of the boundary of each domain is not specified in the domain data tables, but is determined by the distribution line boundaries that define the external domain boundaries. Line boundaries that can be external domain boundaries are *head-specified, normal flux-specified*, and *inter-domain*. The data that is input for these types of line boundaries include information about which domain(s) that they bound. All domains should be completely bounded by such line boundaries, so that their geometry is unambiguous. See the discussion under *Subdomains and Model Levels* for more detail and some examples.

For the best accuracy, make sure the coordinates of the starting and ending points of adjacent external line boundaries match exactly (copy them). For example, if a domain boundary has two line boundaries defining it - one head-specified and the other inter-domain, make sure that the start/end points where these two line boundaries join have the exact same coordinates.

9.4.2   Input Common to All Domains

Several parameters are common and required input for any type of domain:

- The *label_unique* allows you and the model keep track of which domain is which. When adding wells, line boundaries, etc. to the model, this label defines which domain these features are in. These labels are required and must be unique (no two domains should have the same label). Once the labels have been declared and other features have been added to the model, do not change the labels because doing so would require changing the domain label of each well, line boundary, and area source/sink that is in that domain.

- The *level* of a domain refers to where this domain fits in the vertical sequence of model levels. In a multi-layered part of a model, the level begins at 1 at the uppermost level and increases in deeper levels. The *level* may be from 1 to a maximum of 15. If there are vertically stacked domains in a multi-level part of the model, vertical leakage is assumed to occur between domains that are at different levels but occur at the same x,y coordinate. For example, if three domains in a three-level part of the model were assigned levels 1, 2, and 4, there could be

vertical leakage between levels 1 and 2 and between levels 2 and 4, if this area of the model has spatially-variable area source/sinks. When making plots of model results, plots show one level at a time.

- For *Average_head*, list your best estimate of the average head in this domain for your simulation. This then defines a constant that is added to the discharge potential for this domain. More details are given about this in the next topic.

- The value of *Porosity* is used in computing average linear velocity and advection travel times along pathlines. The average linear velocity = specific discharge/porosity. For a solute that adsorbs to the porous medium, the solute travel time will be be longer than the pure water advective travel time. If you want to plot travel times that factor in retardation of a solute, you may input *Retardation Factor * Porosity* instead of *Porosity* in this column and the travel times will reflect travel of a retarded solute plume.

- *K1_horizontal* is the horizontal hydraulic conductivity. In a domain that is anisotropic in the horizontal plane, K1 differs from K2, and these represent the principle hydraulic conductivities in the horizontal plane. K1 > K2, K1 < K2, and K1 = K2 are all possible.

- *K2_horizontal* is the horizontal hydraulic conductivity. In a domain that is anisotropic in the horizontal plane, K2 is in the direction normal to the *K1_horizontal* direction. To simplify inputs, you can enter "=K1" if K2=K1 and the domain is isotropic in the plane of the domain. If you want a fixed ratio of anisotropy, you can enter "=K1*D" where D is a real number. Using "=K1" or "=K1*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated.

- *Angle_K1_to_x* is the angle, in degrees, between the *x* axis and the direction of *K1_horizontal*. Positive angles are measured counter-clockwise from the *x* axis.

- *K3_vertical_top* defines the vertical hydraulic conductivity of the upper half of the domain. This parameter is only used if there is vertical leakage with spatially-variable area source/sinks. To simplify inputs, you can enter "=K1" if K3=K1 and the domain is isotropic normal to the plane of the domain. If you want a fixed ratio of anisotropy, you can enter "=K1*D" where D is a real number less than or equal to 1.0. For example, make K1/K3 = 10 by entering "=K1*0.1". Using "=K1" or "=K1*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated.

- *K3_vertical_bottom* defines the vertical hydraulic conductivity of the lower half of the domain. This parameter is only used if there is vertical leakage with spatially-variable area source/sinks. To simplify inputs, you can enter "=K1" if K3=K1 and the domain is isotropic normal to the plane of the domain. If you want a fixed

ratio of anisotropy, you can enter "=K1*D" where D is a real number less than or equal to 1.0. For example, make K1/K3 = 10 by entering "=K1*0.1". Using "=K1" or "=K1*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated.

9.4.3   Details about Average_head

In other two-dimensional AEM programs such as TWODAN, the flow region is open to infinity, and one unknown that needs to be solved for is the amount of flow that goes between the modeled area and infinity. In these programs, to generate an equation to solve for that additional unknown, you specify a head at one location ("reference head" in TWODAN).

In AnAqSim, each domain model is closed and finite, so there is not that extra unknown. You specify the average head in each domain. which in turn defines a constant that is added to the potential for that domain. Since there are linesinks that bound each subdomain, the flow field outside those linesinks does not matter (the flow to/from infinity doesn't affect the solution inside the domain boundary). You could specify a variety of different average head values, within a reasonable range (close to the actual average), and get essentially identical results.

Say you have a simple AnAqSim one-domain model that has head-specified boundaries all around the external boundary, with h=100. There is zero recharge, so h should be 100 everywhere inside the domain. If you specify the average domain h=100, the program adds a constant to the potential that is the potential corresponding to h=100. On solving, it will turn out that the boundary conditions are met perfectly everywhere on the boundary and the boundary linesinks all have zero discharge; the analytical model will boil down to the simple equation h(x,y)=100. With zero discharges in the boundary linesinks, there is no flow to or from infinity to the model boundary from the outside (even though you never see this part of a domain model, it exists).

Now imagine that instead you set the average domain head to 110, which adds a larger constant to the potential for this domain. Now, to achieve the boundary h=100, the boundary linesinks need to extract water to pull the head surface down. In this case the solution on and inside the boundary will still be approximately h=100, but there will be flow to the outside of the boundary linesinks from infinity. Likewise, if you set the average domain head to 90, the solution on and inside the boundary will be approximately h=100, but there will be flow from the outside of the boundary linesinks to infinity. When you change the average head for a domain, it changes the part of the domain solution that you never see - the part that lies outside the external boundary of the domain.

If you use long line elements with few parameters and the average head is not close to the actual average, the differences in the external, unseen part of the model may have some visible impact on the model within the domain boundary. The most likely manifestation will be some lumpiness in the head surface near those boundary elements.

 Correct this by choosing a more representative average head and/or shortening line boundary elements and increasing the number of parameters per line.

9.4.4   Confined and/or Unconfined

Starting with release 2015-1, confined, unconfined, and confined/unconfined domains are in the same data table, which allows the user to quickly switch between these domain types. For confined and/or unconfined domains, these parameters are needed in addition to those that are common to all domains:

1. *Domain_Type* determines which type of domain is to be modeled. *Confined*, *Unconfined*, and *Confined/Unconfined* are the three options. For *Confined*, the domain is always a fixed saturated thickness equal to the top elevation minus the bottom elevation, and the domain's transmissivity is independent of head. Confined domains generate linear equations, while the other two options can generate nonlinear equations. It is often wise to begin your modeling with confined domains which tend to converge faster and be less prone to numerical issues associated with nonlinearity and drying up.   Later, it is easy to switch to unconfined or confined/unconfined domain types. With the *Unconfined* domain type, the domain is always unconfined and the *Top_elevation* is not used (although some value must be input). The *Confined/Unconfined* domain type behaves like a confined aquifer when the head equals or exceeds the top elevation, but like an unconfined aquifer where the saturated thickness depends on head, when head drops below the top elevation (see Strack, 1989, section 8; Haitjema, 1995, section 3.1.3; Strack, 2003, equation 3).

2. *Top_elevation* defines the elevation of the top of the domain. Not used for unconfined domain type.

3. *Bottom_elevation* defines the elevation of the bottom of the domain.

4. *Storativity (S)* is the dimensionless elastic storage parameter. S normally is the saturated thickness times specific storage (Ss). Starting with release 2021-1, there is an option to input the value of specific storage instead of storativity. This can simplify input by avoiding the step of multiplying by saturated thickness. To input specific storage, enter "Ss:" in front of the value. For example, if you entered "Ss: 0.0037", the value of storativity in this confined domain would be set to 0.0037 * b, where b is the saturated thickness (top elevation - bottom elevation). See *Storage Parameter Details* for more on how different storage parameters apply for different domain types.

5. *Specific_yield (Sy)* is the dimensionless storage parameter for the unconfined domain type. See *Storage Parameter Details* for more on how these parameters apply for different domain types.

For numerical stability where the saturated thickness of an unconfined or confined/unconfined domain approaches zero, AnAqSim imposes a minimum saturated thickness. When heads drop near or below the bottom, the domain reverts to a

confined-type domain with a fixed minimum saturated thickness. This facet of AnAqSim is governed by a parameter called *Almost_dry_fraction* under [Solution/Solve Settings](#).

9.4.5   Confined Interface

For confined interface domains, these parameters are needed in addition to those that are common to all domains:

1. *Top_elevation* defines the elevation of the top of the domain.

2. *Bottom_elevation* defines the elevation of the bottom of the domain. The saturated thickness is the difference between the top and bottom elevations.

3. *Storativity* is the dimensionless storage parameter that applies in the confined portion of this type of aquifer, equal to saturated thickness times specific storage. As described under *[Confined and/or Unconfined](#)* domains, you may enter specific storage values here instead of storativity values.

4. *Salt_elevation* defines the elevation of the surface of the salt water, which is assumed to be static. Typically this is about the elevation of sea level.

5. *DensityRatio* is the ratio of the salt water density to fresh water density. This varies from place to place, but is often near 1.025.

Interface domains in AnAqSim are based on the Ghyben-Herzberg approximation:

- The salt water is hydrostatic - pressure is proportional to depth below *Salt_elevation*. This assumption is reasonable when the flow is roughly steady.

- The fresh/salt water interface is sharp, with no mixing.

- Fresh water within a domain is hydrostatic (Dupuit approximation); there is no vertical resistance to flow.

The confined interface domains are based on the techniques presented by Strack (1989) on pages 101-106 and in Fitts et al (2015). These domains are confined with fresh water from top to bottom when heads are high enough that there is no interface, or they are confined with an interface and some salt water if heads are low enough. Confined interface domains would go to zero fresh water saturated thickness when the fresh water head drops to a level where the fresh water pressure at the top of the domain equals the salt water pressure at that elevation. This occurs where the freshwater head = *Top_elevation + (Salt_elevation - Top_elevation) * DensityRatio*. For numerical stability where the fresh water saturated thickness approaches zero, AnAqSim imposes a minimum saturated thickness. When heads are low enough, the domain reverts to a confined-type domain with this minimum saturated thickness. This facet of AnAqSim is governed by a parameter called *Almost_dry_fraction* under [Solution/Solve Settings](#).

Generally transient simulations with interface domains will be inaccurate because it is assumed that the salt water has a hydrostatic distribution of pressure on the interface. In

most transient situations, the salt water is moving, and when that movement has a vertical component, the hydrostatic pressure assumption is violated. If you feel the hydrostatic salt water assumption is still reasonable, you may proceed with a transient simulation but AnAqSim will issue a warning. See *Storage Parameter Details* for more on how storage parameters apply to this domain type*.

9.4.6   Unconfined Interface

For unconfined interface domains, these parameters are needed in addition to those that are common to all domains:

1. *Bottom_elevation* defines the elevation of the bottom of the domain. The saturated thickness is the difference between the head and the bottom elevation.

2. *Specific_yield* is the dimensionless storage parameter for an unconfined aquifer.

3. *Salt_elevation* defines the elevation of the surface of the salt water, which is assumed to be static. Typically this is about the elevation of sea level.

4. *DensityRatio* is the ratio of the salt water density to fresh water density. This varies from place to place, but is often near 1.025.

Interface domains in AnAqSim are based on the Ghyben-Herzberg approximation:

• The salt water is hydrostatic - pressure is proportional to depth below *Salt_elevation*.

• The fresh/salt water interface is sharp, with no mixing.

• Fresh water within a domain is hydrostatic (Dupuit approximation); there is no vertical resistance to flow.

The unconfined interface domains are based on the techniques presented by Strack (1989) on pages 108-111 and in Fitts et al (2015). These domains are unconfined with fresh water from the water table to the bottom when heads are high enough that there is no interface, or they are unconfined with an interface and some salt water if heads are low enough. Unconfined interface domains would go to zero fresh water saturated thickness when the fresh water head drops to *Salt_elevation*. To avoid dry conditions, keep all heads in the domain above *Salt_elevation*. For numerical stability where the fresh water saturated thickness approaches zero, AnAqSim imposes a minimum saturated thickness. When heads are low enough, the domain reverts to a confined-type domain with this minimum saturated thickness. This facet of AnAqSim is governed by a parameter called *Almost_dry_fraction* under Solution/Solve Settings.

Generally transient simulations with interface domains will be inaccurate because it is assumed that the salt water has a hydrostatic distribution of pressure on the interface. In most transient situations, the salt water is moving, and when that movement has a vertical component, the hydrostatic pressure assumption is violated. If you feel the hydrostatic salt water assumption is still reasonable, you may proceed with a transient

simulation but AnAqSim will issue a warning. See *Storage Parameter Details* for more on how storage parameters apply to this domain type*.

9.4.7   Storage Parameter Details

Storage parameters are defined differently for different domain types as explained below.

- In confined domains, the storage parameter is always S (storativity) regardless of head.

- In unconfined domains, the storage parameter Sy (specific yield) applies when h > the head at minimum saturated thickness (see *Almost_dry_fraction* under Solve Settings). When head is lower than the level at minimum saturated thickness, the storage parameter equals S * *Almost_dry_fraction* (usually a very small value). This scheme helps with convergence in cases where heads fall below the bottom of the domain, but makes storage changes small under these conditions.

- In confined/unconfined domains, storage is like a confined domain when h >= top elevation, and like an unconfined domain when h < top elevation.

- In confined interface domains, the storage parameter = *S + (porosity / (DensityRatio - 1))* where an interface is present. Where the domain is confined with no interface present (higher heads), the storage parameter = S. Generally, storage contributed by interface shifts greatly exceed elastic storage; *S << porosity / (DensityRatio - 1)*.

- In unconfined interface domains,the storage parameter = Sy + (*porosity* / (*DensityRatio* - 1)) where an interface is present. Where head is high enough that there is no interface (inland from the toe of the interface), the storage parameter = *Sy.* Generally the storage contributed by interface shifts is larger than water table storage; *Sy < porosity* / (*DensityRatio* - 1).

In all but confined domains, the storage parameter is a function of head. The head at the start of a time step is used to determine the storage parameter that applies for the time step, even though the head at the end of the time step may correspond to a different storage parameter. This approximation helps convergence and is minor if time steps are small enough.

9.5   Pumping Wells

Pumping wells may be either discharge-specified or head-specified. The discharge-specified type may be screened in one domain or across multiple domains if the well screen spans multiple model levels. All well input data tables may be accessed through the main menu or by using a pop-up context menu when the cursor is over the plot.

9.5.1   Input Common to all Pumping Wells

With all types of pumping wells, the following parameters are required.

- *Label* is a text label that helps you keep track of multiple wells.

- *X,Y* defines the horizontal coordinates of the well. To graphically edit a well's coordinates, left click to select it. Once selected, the well will be enclosed in a purple square boxe as shown below
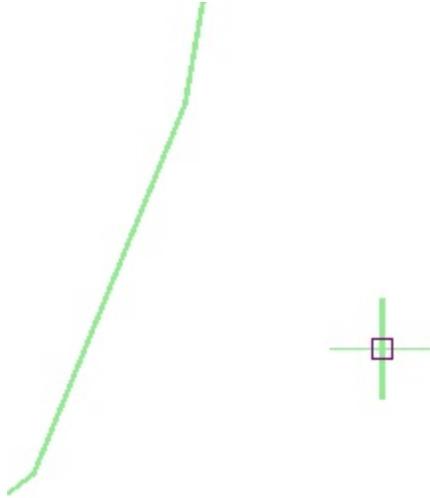
  When highlighted in this way, the well can be moved by clicking on the purple box and dragging it. This will automatically alter the coordinates of the well in the model input table. To stop graphic editing, press Esc to de-select the well.

- *Radius* defines the radius of the well. If there is a high conductivity filter pack around the well screen, the radius should be the radius of the borehole, not the radius of the screen.

9.5.2   Discharge-Specified

With discharge-specified wells, negative rates are used for extraction from the aquifer and positive rates are used for injection into the aquifer. Additional parameters defined here are:

- *Domain* defines the domain that the well screen is in.

- *Discharge* is the discharge of the well in units of $[L^3/T]$. In transient simulations, this parameter may vary from one time period to the next.

9.5.3   Discharge-Specified (Multi-Domain)

Use this type of well to simulate a well with a screen that spans multiple domains and levels in the vertical direction. AnAqSim computes the appropriate discharge from each domain spanned so that the total discharge equals the specified discharge, and the heads at the well radius in each domain match each other. With discharge-specified wells, negative rates are used for extraction from the aquifer and positive rates are used for injection into the aquifer.

- *Domains* defines the domains that the well screen spans.

- *Discharge* is the discharge of the well in units of [L^3/T]. In transient simulations, this parameter may vary from one time period to the next.

9.5.4 Head-Specified

With head-specified wells, you specify a head that applies at the well radius and AnAqSim computes the discharge needed to achieve that head. The discharge of a head-specified well may be checked after solving from the *Analysis* menu.

- *Domain* defines the domain that the well screen is in.

- *Head_at_well* defines the head at the well radius. In transient simulations, this parameter may vary from one time period to the next.

- *Off_Periods* is used in transient simulations if you want the well discharge to be zero during certain periods. The periods you want the well off are delimited with comma(s). For example, the following input is for a transient model with five time periods. The well pumps at rates such that at the end of period 1 the head at the well is 80, the well is off during periods 2 and 3, and the well pumps so that the head at the well is 115 and 118, respectively, at the ends of periods 4 and 5.

Data: head-specified wells

	Label	Domain	X,Y	Radius	Head_at_well	Off_periods
	tw	all	-1362, 2367	0.3	80. 90, 110, 115, 118	2, 3

9.6 Line Boundaries

A variety of line boundary conditions are available in AnAqSim. Each line boundary is a multi-segmented line (polyline) and the user inputs a list of vertexes in sequence from one end of the polyline to the other. The line boundary condition is approximated using linesink elements similar to those described by Jankovic and Barnes (1999).

Most line boundaries have a parameter that varies from one value at the starting vertex to another value at the ending vertex. The interpolation scheme between the end points is described in the next topic.

AnAqSim approximates the specified boundary conditions along line boundaries, as discussed by Fitts (2010). You may check the accuracy of line boundary condition approximations under the *Analysis* menu.

For internal line boundaries, the coordinates of all polyline points must not be outside the subdomain boundary, otherwise numerical havoc will be wreaked! It is possible for

the start or end point of an internal line boundary to coincide exactly with a corner point of an external line boundary.

All line boundary input data tables may be accessed through the main menu or by using a pop-up context menu when the cursor is over the plot.

9.6.1   Input Common to all Line Boundaries

The following input items are common to all of the line boundaries:

- *Label* is a text label that helps you keep track of multiple line boundaries. Please see *River Line Boundary Discharges* for special labeling of River line boundaries so that you may sum the discharges of groups of river boundaries.

- Each line segment has a line element with a certain number of unknown parameters that is defined in the *Parameters_per_line* column. If you choose 1 for this, the strength of the element is constant along the line, if you chose 2 for this, the strength of the element varies linearly along the line, if you choose 3 for this, the strength of the element varies parabolically along the line, etc. Using more parameters per line can increase the accuracy of the boundary condition approximation along the line, but at the cost of additional equations in the system of equations that is solved.   You only need a high number of parameters per line if the heads or discharges along the element are expected to vary complexly. In many cases, 3 or fewer parameters per line is plenty. Experiment with this and reduce the number of parameters to the minimum that gives you reasonable boundary condition accuracy, which you can check using *right-click / Check Line Boundary Conditions*. In the case of a *Discharge-Specified* line boundary, the user does not define *Parameters_per_line* because it is always set to 1; the discharge/length of these line elements is constant and equal to the specified discharge of the line boundary divided by its total length*.

- *Coordinates* defines the x,y coordinates of the polyline vertexes. When you click on a cell in this column, a small text box window appears. In this text box, enter/edit lines of coordinate data, one line per vertex. Each line should have x and y values separated by either a comma or a tab character. You may digitize the coordinates of a polyline in the plot view and then paste the coordinates into this text box. **In the case of normal flux-specified external boundaries, the coordinates must be listed in counter-clockwise order** with the domain to the left of the boundary as you proceed along it. Once input, coordinates can be edited graphically by selecting the line boundary and then moving the vertexes or inserting or deleting vertexes. To graphically edit a line boundary's coordinates, left click to select it. Once selected, the vertexes will be enclosed in purple square boxes as shown below
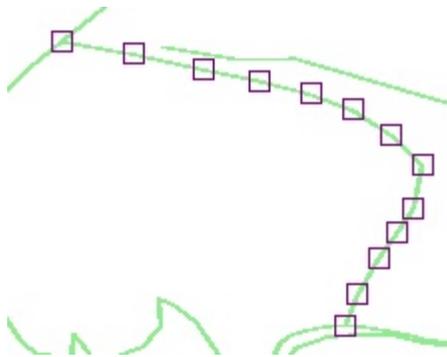
When highlighted in this way, the vertexes can be moved by clicking on the purple box and dragging it. This will automatically alter the coordinates of the line boundary in the model input table. To de-select a line boundary, press Esc. Procedures for graphic editing of line boundaries are covered in the tutorial videos on the web site http://www.fittsgeosolutions.com. To reverse the order of the vertexes (this can be handy if you digitized in the wrong direction), click the **Reverse** button.

Most line boundaries have additional parameters defined at the start and end points, such as heads for head-specified line boundaries. With all such parameters, the same algorithm is employed to interpolate the specified values along the line boundary:

1. apportion the intermediate values to the vertexes based on the number of line segments in the polyline. For example, if there are 4 segments and the end values are 100 and 110, the values at the vertexes would be 100, 102.5, 105, 107.5, and 110. If there are 5 segments, the values at the vertexes would be 100, 102, 104, 106, 108, and 110.

2. Linearly interpolate values to the control points within a line segment, assuming a linear distribution from one end to the other.

The additional parameters specific to each type of line boundary are listed in the following topics.

9.6.2   Head-Specified

With head-specified line boundaries you specify these additional items:

- *Domain* defines the domain that the line boundary is in. Double-click on this cell to open a drop-down list and make a selection.

- Check the *Domain_Boundary* check box if this line boundary is on the external boundary of the domain.

- The *h_start* and *h_end* are the specified head values at the starting and ending vertexes. In transient simulations, these parameters may vary from one time period to the next.

- *Off_Periods* is used in transient simulations if you want the line boundary discharge to be zero during certain periods. This could be handy for a dewatering trench that is turned off and on. The periods you want off are delimited with comma(s). For example, the following input is for a transient model with three time periods. The line boundary discharges at rates such that at the end of period 1 the heads along it are 104, the line boundary is zero during periods 2, and the line boundary discharges at rates such that at the end of period 3 the heads along it are 106.

Data: head-specified lines

Label	Domain	Parameters_per_line	Domain_Boundary	h_start	h_end	Coordinates	Off_periods
lake	all	6	☑	104, 105, 106	104, 105, 106	Edit	2
			☐			Edit	

Boundary condition equations are written at each control point on each line segment. The equation specifies that the modeled head = specified head (interpolated between endpoints of the line boundary). The specified head condition is approximated between control points and may be checked graphically.

9.6.3   Normal Flux-Specified

With normal flux-specified line boundaries you specify these additional items:

- *Domain* defines the domain that the line boundary is in. Double-click on this cell to open a drop-down list and make a selection.

- Check the *Domain_Boundary* check box if this line boundary is on the external boundary of the domain. When the boundary is external, it is **necessary to list the coordinates of the polyline in counter-clockwise order** with the domain to the left of the boundary as you proceed along it. If coordinates are specified in the wrong (clockwise) order, there is a check in the program that will detect and report this error. This error can occur if you mistakenly specified the coordinates in clockwise order around the outside of the domain. This message can also result if you have made errors in specifying additional external line boundaries either at these same coordinates (e.g. have two external line boundaries in the input that share the same vertex coordinates), or have erroneous external line boundaries to the right of this one (see discussion of left/right algorithm).

- The *Normal_flux_start* and *Normal_flux _end* are the specified normal flux values at the starting and ending vertexes. Normal flux is the component of domain discharge normal to the line segment, and has units of discharge per length $[L^3/T/L] = [L^2/T]$. This may also be thought of as the normal component of specific discharge times saturated thickness. The normal flux is positive for flow across the boundary from left to right as you proceed from the start toward the end. Normal flux is negative for flow from right to left as you proceed from the start toward the end.   In transient simulations, the normal flux parameters may vary from one time period to the next.

Boundary condition equations are written for sub-intervals in each line segment (e.g. three subintervals for 3 parameters/line). The equation specifies that the total discharge across the line over the subinterval equals the total to interpolated specified fluxes over the interval. The accuracy of this approximation may be checked graphically.

9.6.4   Head-Dependent Normal Flux (3rd type)

This allows head-dependent normal flux into or out of the model, depending on the modeled head at the boundary. Boundary conditions like this are sometimes called 3rd type, Robin, general head (GHB in MODFLOW), or mixed boundary conditions; they involve both head and flux. The boundary must be an external boundary of a domain. It is **necessary to list the coordinates of the polyline in counter-clockwise order** with the domain to the left of the boundary as you proceed along it. If coordinates are specified in the wrong order, there is a check in the program that will detect and report this error.   This error can occur if you mistakenly specified the coordinates in clockwise order around the outside of the domain. This message can also result if you have made errors in specifying additional external line boundaries either at these same coordinates (e.g. have two external line boundaries in the input that share the same vertex coordinates), or have erroneous external line boundaries to the right of this one (see discussion of left/right algorithm).

This kind of boundary is illustrated conceptually in the vertical profile sketched below. It is as though there is a fictional domain beyond the boundary (orange) outside of the domain (blue) , and at a distance b outside the boundary, there is a fixed head h*. The head difference between h* and the head at the boundary (h) drives a component of discharge normal to the boundary.
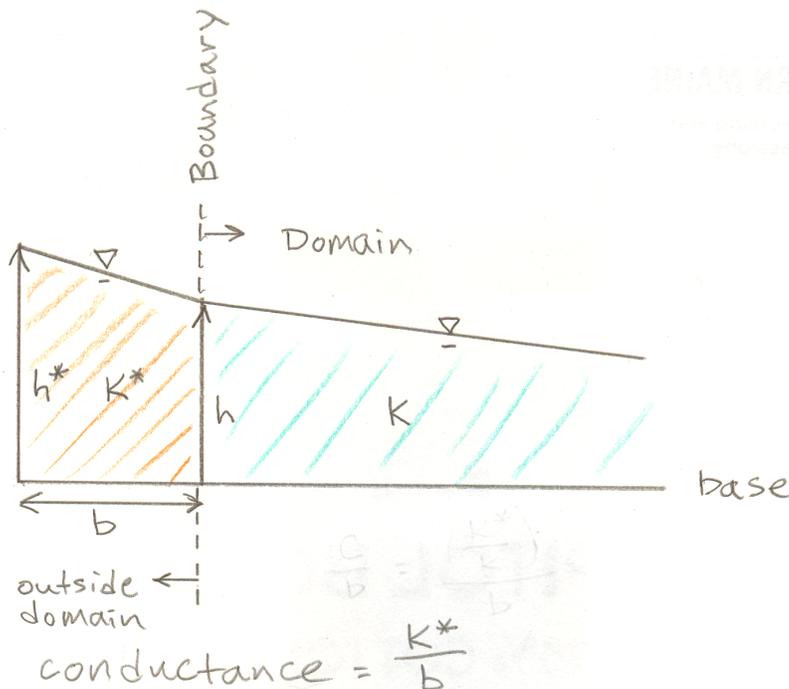
You specify these additional items with this type of boundary:

- *Domain* defines the domain that the line boundary borders. Double-click on this cell to open a drop-down list and make a selection.

- *Conductance_start* and *Conductance_end* = K*/b at the starting and ending vertexes, where b is the horizontal width of the fictional domain normal to the line boundary and K* is the fictional domain horizontal K*. When K*/b is smaller, there is low conductance at this boundary and the normal component of discharge is smaller. The entry for *Conductance_end* can be "=start" if you want the same conductance for the entire line boundary. Alternatively, the entry for *Conductance_end* can be "=start*D" if you want the *Conductance_end* to be equal to *Conductance_start* times the real constant D. Using "=start" or "=start*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated. Prior to release 2016-3, the conductance parameter for this kind of line boundary was defined as C/b = (K*/b/K), which differs from this conductance by the factor K, which is the K of the domain.   When a domain is anisotropic in the plane of the domain, the factor K = $(K1 * K2)^{1/2}$.

- The *h*_start* and *h*_end* are the specified head values at the starting and ending vertexes, representing head a distance b outside the boundary.

The component of aquifer discharge normal to the boundary is Qn [L^2/T], determined by this equation: Qn = (K*/b/K) (Φ-Φ*), where Φ* is the aquifer discharge potential computed for the domain using h*, and Φ is the aquifer discharge potential computed using modeled h at the boundary. Qn is positive for discharge out of the domain. The equation relating h to Φ depends on the domain type - it is a linear relation for confined domains but is nonlinear for unconfined and interface domains. Books describing Φ(h) equations in the analytic element method include [Fitts (2012), Strack (1989), and Haitjema (1995)](#). By writing the boundary condition equation this way, the boundary condition is linear, even when Φ(h) is nonlinear. For a confined isotropic domain, the normal component of aquifer discharge is Qn = (K*/b/K)(T)(h-h*), where T is the transmissivity of the domain and h is the modeled head at the boundary. For an unconfined isotropic domain, the normal component of aquifer discharge is Qn = (K*/b/K)(K/2)(h^2-h*^{2}), where K is the horizontal K of the domain, and h and h* are measured from the base of the domain.

The boundary condition equation Qn = (K*/b/K)(Φ-Φ*) is written at each control point on each line segment. The values of conductance and Φ* are interpolated between the values specified at the start and end of the boundary. The modeled boundary condition is approximated between control points and may be [checked graphically](#).

9.6.5 Discharge-Specified

With discharge-specified line boundaries you specify these additional items:

- *Domain* defines the domain that the line boundary is in. Double-click on this cell to open a drop-down list and make a selection.

- *Discharge* of the entire polyline in volume/time units [L^3/T].

With this type of line boundary, the discharge per length of line is constant along the entire length of the polyline. Negative discharge implies extraction from the aquifer and positive discharge implies injection into the aquifer. In transient simulations, this parameter may vary from one time period to the next. This type of line boundary must always be internal, not on the external boundary of a domain. To specify discharge into or out of the external boundary of a domain, use Normal Flux-Specified boundaries.

9.6.6   River

With river line boundaries you specify these additional items:

- *Label*. Please see *River Line Boundary Discharges* for special label format of River line boundaries so that you may output the total discharges of groups of river boundaries.

- *Domain* defines the domain that the line boundary is in. Double-click on this cell to open a drop-down list and make a selection.

- Check the *Dries_up* check box if this river boundary is allowed to dry up (revert to zero discharge per length) when the head falls below the specified stage. Check *Dries_up* in cases where the river is small and would dry up if aquifer heads drop below stage. If *Dries_up* is not checked, the river's discharge is computed based on a head difference and a river bed conductance factor in accordance with Darcy's Law. There are two possibilities for how the head difference is computed: 1) if the domain head is above the base of the resisting bed, then it is the difference between the stage and the head in the domain, and 2) if the domain head is below the base of the resisting bed, then it is the difference between the stage and the elevation of the base of the resisting bed. The following graph illustrates how the discharge/length to the river line boundary would vary with or without *Dries_up* checked, for the following assumptions: stage = 100, resisting bed base = 98, resisting bed conductance C = K_v w / d = 0.2.
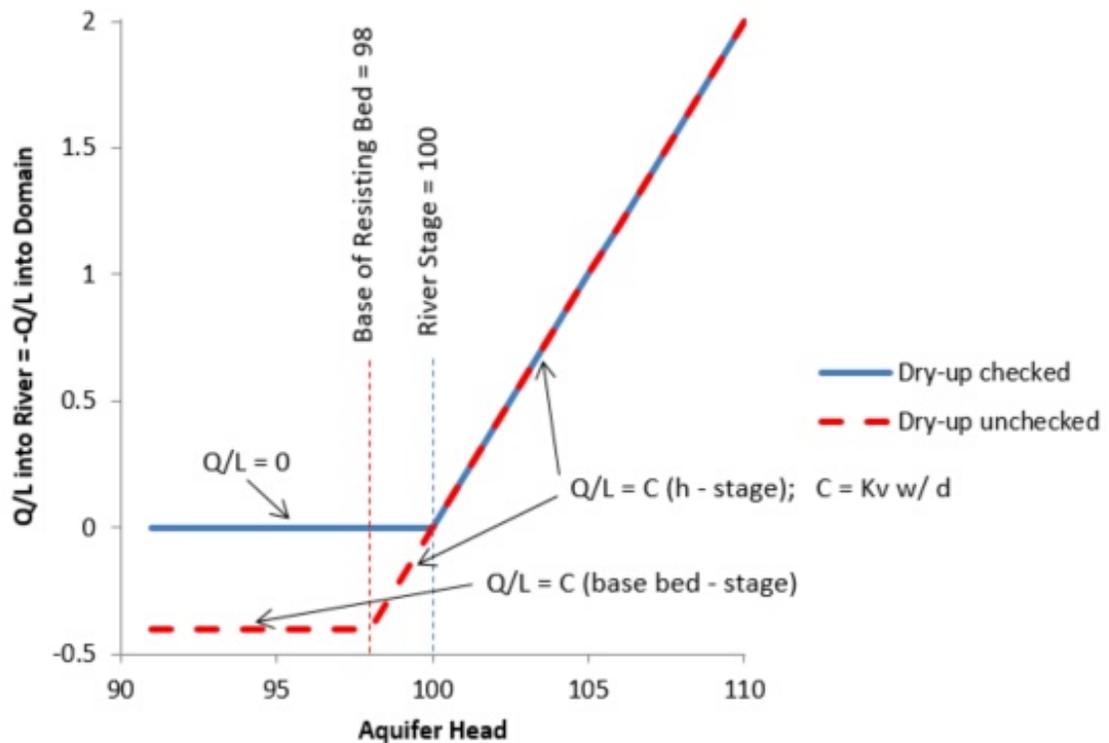
- *Stage_start* and *Stage_end* are the specified river stages (water surface elevations) at the starting and ending vertexes. In transient simulations, these parameters may vary from one time period to the next.

- *Conductance_start* and *Conductance _end* are the specified river bed conductance factors at the starting and ending vertexes. The conductance factor equals K_v w/d where K_v is the vertical hydraulic conductivity of the resistant river bed, *w* is the horizontal width of the river bed, and *d* is the thickness of the resistant river bed. K_v w/d has units of [L/T]. The entry for *Conductance _end* can be "=start" if you want the same conductance for the entire line boundary. Alternatively the entry for *Conductance _end* can be "=start*D" if you want *Conductance _end* to be equal to *Conductance_start* times the real constant D. Using "=start" or "=start*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated.

- *Base_resisting_layer_start* and *Base_resisting_layer_end* are the specified elevations of the bottom of the river bed resisting layer at the starting and ending vertexes.

Boundary condition equations are written at each control point on each line segment. The equation specifies that the discharge/length of the linesink be related to the stage

and conductance(interpolated between endpoints of the line boundary) as described above. The specified head condition is approximated between control points and may be checked graphically. In the graphical check, the sign of the discharge is negative for flow from the domain to the river, consistent with other discharging elements.

Instead of digitizing rivers and adding line boundaries to the data table one row at a time, you may import river line boundary data from tab-delimited text that has been copied into the clipboard from a file or from a spreadsheet (e.g. Excel). This may be a useful way to transfer large amounts of river data from other software into AnAqSim.

9.6.7   Drain/Fracture

Drain/Fracture line boundaries allow modeling of thin features like drains or fractures that conduct water at a high rate compared to the surrounding domain. The drain or fracture is represented in the model as a line with zero actual width using line dipole functions. With these line boundaries, specify these additional items:

- *Domain* defines the domain that the line boundary is in. Double-click on this cell to open a drop-down list and make a selection.

- *Conductivity_start* and *Conductivity_end* = bK* at the starting and ending vertexes, where b is the horizontal width of the drain normal to the direction of the line, and K* is the drain hydraulic conductivity. This formulation assumes the drain fully penetrates the domain. The entry for *Conductivity_end* can be "=start" if you want the same conductivity for the entire line boundary. Alternatively the entry for *Conductivity_end* can be "=start*D" if you want the *Conductivity_end* to be equal to *Conductivity_start* times the real constant D. Using "=start" or "=start*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated. Prior to release 2016-3, the conductance parameter for this kind of line boundary was defined as bR = (bK*/K), which differs from this conductance by the factor K, which is the K of the domain.   When a domain is anisotropic in the plane of the domain, the factor K = $(K1 * K2)^{1/2}$.

Boundary condition equations are written at each control point on each line segment. The equation specifies that the modeled discharge within the drain (jump in stream function across drain) equals *Conductance*/K (interpolated between endpoints of the line boundary) times the gradient of the aquifer discharge potential in the direction of the drain. This condition is approximated between control points and may be checked graphically. In the graphical check, the discharge is positive for flow from the start of the boundary toward the end of the boundary, as defined by the order of vertexes. To achieve better accuracy of this approximation, use shorter line segments and/or more parameters/line. Shorter segments are particularly helpful at the ends of the line boundary, where the boundary condition changes rapidly.

9.6.8   Leaky Barrier

Leaky Barrier line boundaries allow modeling of thin features like sheet pile walls or slurry walls that create a barrier to flow, but allow some leakage. The leaky barrier is represented in the model as a line with zero actual width using line doublet functions. With these line boundaries, specify these additional items:

- *Domain* defines the domain that the line boundary is in. Double-click on this cell to open a drop-down list and make a selection.

- *Conductance_start* and *Conductance_end =* K*/b at the starting and ending vertexes, where b is the width of the barrier normal to the direction of the line, and K* is the horizontal conductivity of the barrier. It is assumed that the barrier fully penetrates the domain. The entry for *Conductance_end* can be "=start" if you want the same conductance for the entire line boundary. Alternatively the entry for *Conductance_end* can be "=start*D" if you want the *Conductance_end* to be equal to *Conductance_start* times the real constant D. Using "=start" or "=start*D" is particularly handy for parameter estimation, to limit the number of parameters being estimated. Prior to release 2016-3, the conductance parameter for this kind of line boundary was defined as R/b = (K*/b/K), which differs from this conductance by the factor K, which is the K of the domain. When a domain is anisotropic in the plane of the domain, the factor K = $(K1 * K2)^{1/2}$.

Boundary condition equations are written at each control point on each line segment. The equation specifies that the modeled aquifer discharge component normal to the barrier equals *Conductance*/K (interpolated between endpoints of the line boundary) times the jump in aquifer discharge potential across the barrier. This condition is approximated between control points and may be [checked graphically](#).   In this check, discharge is positive for flow going from left to right as you proceed along the boundary from the start towards the end. To achieve better accuracy of this approximation, use shorter line segments and/or more [parameters/line](#). Shorter segments are particularly helpful at the ends of the line boundary, where the boundary condition changes rapidly.

9.6.9   Inter-Domain

Inter-domain line boundaries define the boundaries between abutting domains. It is possible to use these boundaries to abruptly shift the layering (levels) in the model. For example, there could be a single level on one side of the boundary and multiple levels on the other side. For inter-domain boundaries, you specify these additional items:

- *Domains_left* defines the domains that are to the left of the boundary as you proceed from the first towards the last vertex listed under *Coordinates*. Imagine you were on the ground walking along the boundary heading towards the last vertex - the left domains would be on your left and the right domains would be on your right. Select these domains by clicking on the *Select* button.

- *Domains_right* defines the domains that lie to the right of the boundary as you proceed from the first towards the last vertex listed under *Coordinates*. Select these domains by clicking on the *Select* button.

It is easy to make a mistake specifying *Domains_left or Domains_right,* so AnAqSim checks this and will alert you with an error message if it detects an error. This error can occur if you mistakenly specified the left domain(s) on the right or vice versa. This message can also result if you have made errors in specifying additional external line boundaries either at these same coordinates (e.g. have two external line boundaries in the input that share the same vertex coordinates), or have erroneous external line boundaries to the right of this one (see discussion of left/right algorithm).

On one side of the inter-domain boundary, there should be a single domain. The reasons for this are explained in the *AnAqSim Modeling Concepts* section. At inter-domain boundaries, the domain discharges normal to the boundary are matched so that over sub-segments of each line segment, the total normal discharge of domains to the left matches the total normal discharge of domains to the right. The number of sub-segments per line segment equals the number of parameters per line. The heads in all domains match at control points along the line segment. The number of such control points equals the number of parameters per line. Fitts (2010) provides more discussion of how boundary conditions are approximated at inter-domain boundaries. The accuracy of the modeled boundary conditions may be checked graphically.

9.6.10 Head-Specified External with Gradient

This is a useful line boundary for simple models where you want to impose a uniform groundwater flow across your area of interest, but do not want to specify a lot of other details. Head-specified line boundaries are placed around a roughly circular outer boundary, and the specified heads create a uniform cross-flow in the absence of other elements. This feature creates a 12-sided approximately circular boundary consisting of specified head line boundaries, which forms the entire external boundary of the domain.

With this type of line boundary you specify these additional items:

- *Domain* defines the domain that the line boundary is external to. Double-click on this cell to open a drop-down list and make a selection.

- *X,Y* defines the coordinates of the center of the circle (put at the center of the area of interest).

- *Radius* is the radius of the circle.

- *Gradient* is the desired horizontal hydraulic gradient at the center of the circle, in the absence of any other elements.

- *Gradient_direction* is the angle that the down-gradient direction of the gradient makes with the positive x axis, in degrees.

Boundary condition equations specifying head are written at each control point on each line segment. The specified head condition is approximated between control points and may be checked graphically.

In all types of domains, this boundary by itself produces a uniform flow across the circle, with a uniform aquifer discharge (specific discharge * saturated thickness), and head at the center of the circle will equal the *Average_head* specified for the domain. With a confined domain, the specified gradient will be the gradient everywhere along the circle boundary and everywhere within the circle in the absence of other elements. With unconfined or interface domains and no other elements, the gradient changes with saturated thickness (higher gradient with less saturated thickness), and the specified head gradient will occur at the center and along the midline of the circle.

The head contours below show a model with only this type of line boundary and an unconfined domain where the saturated thickness ranges from 33.4 to 11.5 across the circle; note the larger hydraulic gradient where the saturated thickness is smaller. The aquifer discharge is constant within the circle, as is the saturated thickness * hydraulic gradient. In this case, the average head of 25 and the specified gradient occur at the center and midline.
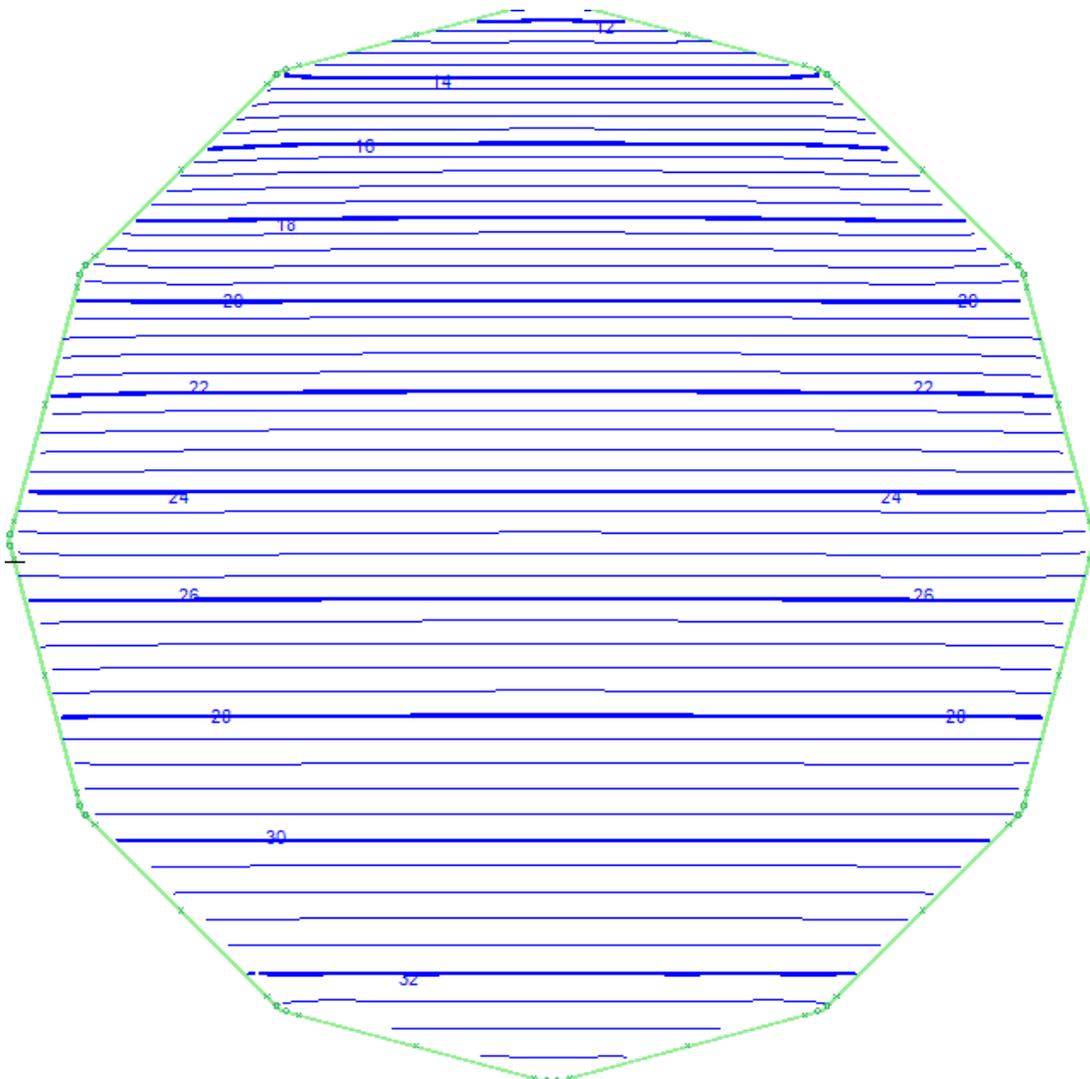
9.6.11 Import River Data from Clipboard

This menu selection is an alternate way to create new river line boundaries, in addition to Model Input Menu / Line Boundaries / River. This alternative, found under Model Input Menu / Line Boundaries / Import River Data from Clipboard, allows you to import text data to build river line boundary input multiple rows at a time, instead of digitizing and creating river boundaries one at a time. This could be useful for importing river data from GIS, for example.

This command reads tab-delimited text from the clipboard and creates new River data rows in the database. If you have the data in an Excel spreadsheet, you can highlight and copy the portion of the sheet into the clipboard, and then execute this command to

import that data. An example of the correct data format is shown below in the selected area of an Excel spreadsheet bounded by the green dotted line:

	A	B	C	D	E
1	x	y	cond	sed base	stage
2	test label	all		3	FALSE	
3	1	2	10		70	75, 76, 77
4	3	4			
5	5	6			
6	7	8	11	61	63
7	9	10			
8	11	12			
9	13	14	12	55	57

First Row

The first selected row (row 2 of the spreadsheet) contains data that applies to all subsequent rows. In this first row of input data are:

• a text label for these river line boundaries (column A)

• the *Domain* the river line boundaries are in (column B)

• the *Parameters_per_line* for the river line boundaries (Column C)

• The *Dries_up* value for the river line boundaries (enter "true" or "false")

Vertex Rows

The subsequent rows (rows 3-9 of the above spreadsheet) contain data for each vertex of the river line boundaries. In these rows, the data are:

• X coordinate of vertex (column A)

• Y coordinate of vertex (column B)

• *Conductance* at the vertex (column C) [optional parameter]

• *Base_resisting_layer* at the vertex (column D) [optional parameter]

• *Stage* at the vertex (column E) [optional parameter, may have multiple comma-separated values for a transient simulation with changing stages]

In the above example, two river line boundaries are input. The first one includes the vertex data from rows 3-6 and the second one includes vertex data from rows 6-9. If a vertex row contains only the X, Y coordinates (like rows 4,5,7,8), those vertexes are within a line boundary, not at the start or end of one. A vertex at the start or end of a river line boundary must have all 5 columns of data to specify beginning and ending values of

Conductance, Base_resisting_layer, and *Stage*. All 5 columns of data are required of the first vertex row (e.g. row 3) and the last vertex row (e.g. row 9). The example data above, when imported, created these two rows of river line boundary input:

Label	Domain	Parameters_per_line	Dries_up	Stage_start	Stage_end	Conductance_start	Conductance_end	Base_resisting_layer_start	Base_resisting_layer_end	Coordinates
test label:#1	all	3	☐	75,76,77	63	10	11	70	61	Edit
test label:#2	all	3	☐	63	57	11	12	61	55	Edit
			☐							Edit

Note that the label for these two rows took the label specified in the first row of the clipboard data and then added ":#1", ":#2"...to the labels of the multiple river line boundaries created. This gives each a unique label, and the common root of the label ("test label:#" in this case) is used by AnAqSim in Analysis / River Line Boundary Discharges to sum up the discharges of all river line boundaries with this common root, in addition to the discharge of each river line boundary. For the example above, the coordinates of "test label:#1" are:

1, 2

3, 4

5, 6

7, 8

and the coordinates of "test label:#2" are:

7, 8

9, 10

11, 12

13, 14

9.7   Area Source/Sink

Area sources or sinks are used to distribute extraction/discharge over an area in a domain as opposed to wells and line boundaries, which have extraction/discharge at a point or line. Area source/sink extraction/discharge rates are in units of volume/time/area [L^3/T/ L^2] = [L/T], like is typical of precipitation or recharge rates. Area source/sinks can represent recharge, vertical leakage, or storage fluxes in transient models. See the *Recharge, Leakage and Transient Storage* section for more discussion.

In steady-state models in domains that are in a single-level part of the model, it can be efficient and appropriate to use a uniform (extraction rate independent of x,y) area source/sink over the entire domain to represent a constant rate of recharge, for example. The Uniform, Domain area source/sink is for such cases. In these cases, the governing equations for aquifer flow are satisfied exactly.

In other cases where the vertical leakage or storage fluxes vary with location (x,y), you'll need to use spatially variable area sinks (SVAS). SVAS are usually needed in multi-level areas of a model, and in transient models.

SVAS and uniform area sinks are mutually exclusive. To avoid this dilemma, no SVAS basis points are placed inside any area that is also covered by a uniform area sink.

All area sink input data tables may be accessed through the main menu or by using a pop-up context menu when the cursor is over the plot.

9.7.1   Uniform, Domain

With this, the model simulates a constant rate of recharge/leakage over the entire domain.

- *Domain* defines the domain that the uniform recharge/leakage applies in.

- *Top_flux* defines the recharge/leakage rate that is applied at the top of the domain, with [L/T] units. This is positive for flux into the aquifer.

- *Bottom_flux* defines the recharge/leakage rate that is applied at the bottom of the domain, with [L/T] units. This is positive for flux into the aquifer.

9.7.2   Spatially Variable Area Sink Overview

With non-uniform vertical leakage or transient flow, the area source/sink extraction/discharge needs to be spatially variable (varies with x,y) because leakage and/or storage fluxes are not constant over the entire domain. AnAqSim uses multi-quadric interpolator functions to simulate spatially variable area source/sinks (SVAS) (see Fitts (2010) or Strack and Jankovic (1999)). With SVAS, the governing equations of aquifer flow are approximated, and this approximation can be quite good if the density of basis points is high enough (Fitts, 2010). You can check the accuracy of the approximation with the *Analysis/Graph Conditions Along a Line* or *Analysis/Graph Transient Line Conditions* which does the same check as shown in Figures 10 and 11 of Fitts (2010).

When you use SVAS, you may assign them over the area of a domain or over the area of a polygon. In either case, SVAS are created in all domains at all levels that are within the area of the specified domain or polygon. If there is a multi-level area of the model, you only input SVAS in one of the levels in the stack. AnAqSim automatically assigns SVAS in the other subdomains that are vertically above or below the ones you specify the SVAS in. When you specify a SVAS by domain or polygon, you specify the condition at the top of

the top layer of the model and at the bottom of the bottom layer of the model in the area of the domain or polygon. This condition may be flux-specified, or head-dependent flux -specified. The top or bottom condition may be a specified constant, or it may be defined by an irregular surface (new in release 2017-1).

You have two ways of assigning SVAS and associated basis points:

1. spreading basis points in all model levels at a specified spacing over the area of a domain (*Spatially Variable, Domain*), and

2. spreading basis points in all model levels at a specified spacing over the area of a polygon (*Spatially Variable, Polygon*). SVAS polygons are used to refine the top condition, bottom condition, and/or basis point spacing. SVAS polygons first erase any basis points that were inside the polygon and then replaces them at the specified top/bottom conditions and spacing.

In addition, you can refine basis point spacing (but not top/bottom conditions) in the near vicinity of wells (Spatially-Variable, Well Basis Points). This places basis points with an efficient radial, logarithmic spacing to achieve accurate solutions near wells.

The spatially-variable area sink functions are smooth within a domain, but can change abruptly across a domain boundary. An example of this is the pond boundary in the example model in the Fitts (2010) paper. This is a nice feature because at some boundaries like the pond shoreline, you want the extraction to change abruptly.

The rules governing SVAS specifications are summarized below:

• No SVAS are allowed in domains that have uniform area sinks.

• No SVAS basis points are placed in areas also covered by uniform area sinks.

• SVAS-polygon specifications overwrite SVAS-domain specifications, where they overlap.

• SVAS-polygon specifications with a higher nesting level overwrite SVAS-polygon specifications with a lower nesting level.

• The basis point spacings defined with the SVAS-well feature overwrite SVAS-domain and SVAS-polygon basis point spacings, but do not alter the top and bottom conditions in the well vicinity.

9.7.3   Spatially Variable, Domain

This sets up a spatially-variable area source/sink over a whole domain with a uniform spacing of basis points. The governing equation is met perfectly at the basis points, which are distributed on hexagonal centers at the specified spacing. The governing equation is approximated in between basis points.

• *Domain* defines the domain that the spatially-variable area source/sink applies in.

- *Condition_Top* defines the type of condition that applies at the top of the domain. This is either flux-specified (e.g. a specific recharge rate) or head-dependent flux-specified (e.g. leakage from a lake into the underlying domain). With the latter condition, the source/sink extraction rate is computed using the head difference between a specified head and the domain head, and the vertical resistance to flow offered by the upper or lower half of the domain, in accordance with Darcy's Law.

- *Top_flux_or_head* defines the flux (flux-specified) or head (head-dependent flux-specified) at the top of the domain. This applies throughout the domain. In transient simulations, this parameter may vary from one time period to the next. If you select "none" under *Top_surface,* the value specified here applied over the entire area of the domain. If you select a surface instead of "none" under *Top_surface,* the value specified here is a multiplier that scales the surface (see description below under *Top_surface*). If you want to use the surface values directly, enter "1.0" here. If, for example, you want to use double the values of the surface as your top flux, enter "2.0" here.

- *Top_surface* allows you to define a surface for a spatially-variable top condition. This could be used to represent an irregular distribution of recharge at the top of the model, or an irregular distribution of head in an aquifer below the model which the lowest model level leaks to/from. To use this feature, you must first define the surface by adding it under *Area Source/Sink/SVAS Top/Bottom Condition Surfaces*. Once you have created the surface, you may select it in the drop-down list that appears in this column. Your choices will be "none" (just use the specified constant value under *Top_flux_or_head*), or one of the previously-defined surfaces. If you choose a surface, the value of the top/bottom condition flux (flux-specified) or head (head-dependent flux-specified) will be equal to the value of the surface at x,y times the number entered under *Top_flux_or_head.* If you select a surface and put "1.0" in the *Top_flux_or_head* field, the value at x,y, will be the value at x,y in the specified surface. If you had entered "1.3" in the *Top_flux_or_head* field, the value at x,y, will be the value at x,y in the specified surface * 1.3. When you have a surface selected, the *Top_flux_or_head* field becomes a multiplier that allows you to scale the magnitude of the surface used in the model. In transient models, you can vary this scaling factor from one time period to the next by entering multiple values separated by commas (see this guidance for transient inputs)

- *Condition_Bottom* defines the type of condition that applies at the bottom of the domain and is analogous to the *Condition_Top* (see above).

- *Bottom _flux_or_head* defines the flux (flux-specified) or head (head-dependent flux-specified) at the bottom of the domain, analogous to the *Top_flux_or_head* (see above). This applies throughout the domain. In transient simulations, this parameter may vary from one time period to the next.

- *Bottom_surface* allows you to define a surface for a spatially-variable bottom condition, and is analogous to the *Top_surface* (see above).

- *Node_spacing* defines the distance between basis points in the spatially-variable area source/sink. Smaller numbers here improve the approximation of the governing flow equations, at the cost of more computation. The appropriate spacing of nodes depends on the transmissivity of the domains and on the vertical conductances involved with the domain, and on how abruptly heads vary with position. A parameter called the leakage factor [L] can provide useful guidance for assigning basis point spacing. For more discussion of appropriate node spacing and the leakage factor, see Strack and Jankovic (1999) or Hansen (2002). The guidance in these papers recommends basis point spacings on the order of the leakage factor to 2x or 3x the leakage factor. The leakage factors are computed and displayed on the left of plots, as discussed in the *Plot View* section of this user guide. Checking extraction along a line in AnAqSim is the best way to know if your spacing is small enough for a satisfactory approximation of the governing equations. The two curves in the extraction plot will not deviate significantly if the basis point spacing is small enough.

9.7.4   Spatially Variable, Polygon

This modifies spatially-variable area source/sinks within a polygonal area. This can be used in addition to the *Spatially Variable, Domain* option to make the spacing of basis points denser in some areas, and to specify different flux or head-dependent flux conditions in some areas.

- *Label* is a text label that helps you keep track of multiple spatially variable, polygon area source/sinks.

- *Nesting_level* is a parameter that allows you to nest several spatially variable, polygon area source/sinks inside each other. If you are not nesting or overlapping these *Spatially Variable, Polygon* features, specify 1 for this. If you want to place a smaller polygon with denser basis point spacing inside another polygon with less dense spacing, use nesting level 1 for the larger polygon and nesting level 2 for the smaller one inside of it. You may nest up to 5 levels to drill down to finer basis point spacing in the key areas (e.g. near a well). The polygons with the higher nesting level overwrite the basis point spacing and flux condition of polygons with lower nesting levels.

- *Condition_Top*, *Top_flux_or_head*, *Top_surface*, *Condition_Bottom*, *Bottom _flux_or_head, Bottom_surface,* and *Node_spacing* are all the same as defined for *Spatially Variable, Domain* area source/sinks.

- *Coordinates* defines the coordinates of the closed polygon, inside of which the specified flux and node spacing apply. The polygon is automatically closed from the last vertex listed to the first one listed here. You may digitize the polygon using Digitize/Polyline in the Plot View. You may also later edit the polygon graphically, moving, inserting, and deleting vertexes.

9.7.5   Spatially Variable, Well Basis Points

Here you can define special, efficient spatially-variable area sink (SVAS) basis point spacings that apply near pumping wells, to improve the accuracy of modeled vertical leakage and storage fluxes. The basis points are drawn with density that is high near the well and decreases with radial distance from the well. The closest basis points to the well are placed at a radius two times the radius of the well. Basis points are placed on larger and larger circles out from the well, where the distance between circles increases logarithmically between adjacent circles (like a steady well flow flow-net).   The outer limit of these basis points is determined by the *Max_node_spacing* parameter - circles of basis points continue to be added in the outward direction, until the basis point spacing would exceed *Max_node_spacing.* Every other circle has all the basis point locations rotated by an angle that is half the angle between basis points on a circle.   For example, if you chose 4 *Points_per_circle,* the angle between basis points on a circle would be 360/4 = 90 degrees, and every other circle would have these positions rotated by 45 degrees compared to adjacent circles of basis points (see figures below). These special well basis points replace basis points in this area that were established with SVAS domain or polygon input. The well basis point input only refines the basis point spacing and does not alter the top and bottom conditions that are determined by the SVAS domain or polygon input.

These data are input:

- *Well_Labels* allows you to select which wells this row of input data applies to. You may select one or multiple wells. Make sure all wells in your model have unique labels, so you can identify them in this list.

- *Max_node_spacing* defines the maximum node spacing at the outer edge of these special basis points. Generally you should have this spacing slightly smaller (about 10-30% less) than the background basis point spacing for the well vicinity.

- *Points_per_circle* defines how many basis points will be placed on each circle around the well. Chose from the drop-down list. A higher number here will make the approximation of storage fluxes and vertical leakage near the well more accurate, but at the expense of more basis points, more equations, and more computation.

The algorithm for placing basis points is illustrated below with a well flow-net plot and basis point placement shown with green triangles for the case of 6 points per circle (basis points shown for 5 outermost circles).
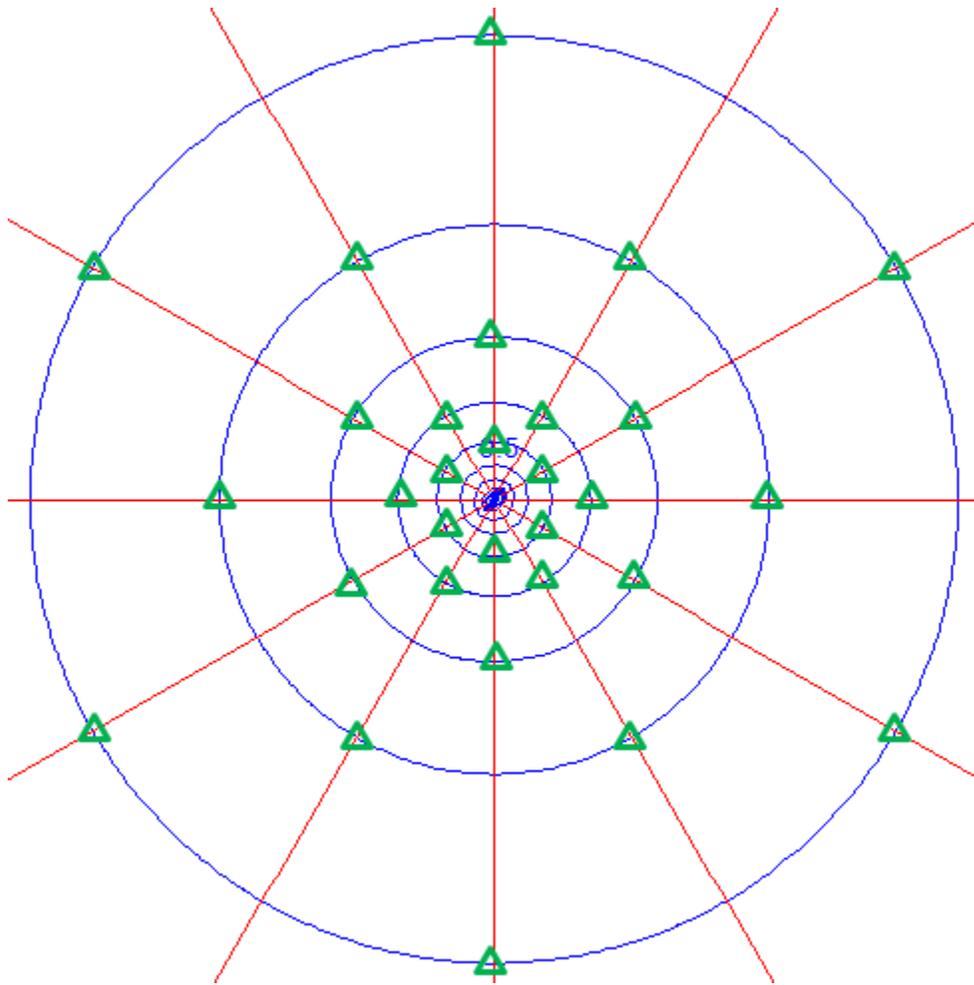
An AnAqSim plot showing well basis points with 8 points per circle is shown below, with a circle drawn in red around the area of special well basis points. Outside this circle is the regular hexagonal basis point spacing defined by SVAS domain or polygon inputs.
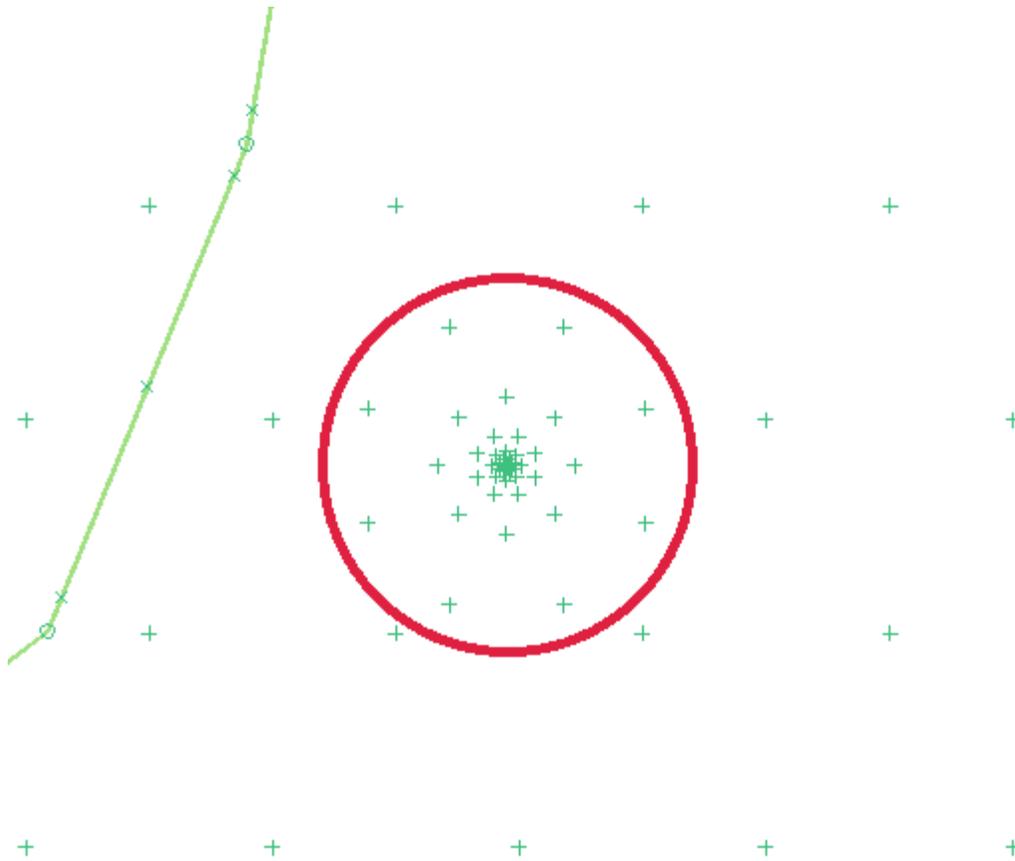
9.7.6   SVAS Top/Bottom Condition Surfaces

This allows you to input a surface, so that spatially-variable area sink (SVAS) top and bottom conditions can be a assigned using an interpolation function of (x,y) position instead of being constant and independent of (x,y).   Some possible applications of this include:

- recharge rate that varies spatially - e.g. more in uplands, less in lowlands.

- irregular distribution of head in an aquifer below the model, which the lowest model level leaks to/from.

- an irregular, sloping freshwater head surface that an interface aquifer leaks toward at the sea shore. The freshwater head increases with depth below sea level, so the specified head condition at the top of an interface domain should be a surface.

The data points that define these surfaces do not correspond to basis points in the SVAS - these data points only define the surface that defines flux or head-dependent flux that is assigned at each basis point. Such surfaces may be referenced by both SVAS, domain

and SVAS, polygon input. The surface is created using interpolation between data points that are specified as *x,y,value* triplets.   You must specify at least 3 data points.

If you specify exactly 3 data points, the surface will be a planar surface that passes through each of the 3 points, using the function

Value(x,y) = Ax + By + C   (A, B, C are scalar constants)

If you specify 4 or more data points, the surface is defined using an interpolation function described by Hardy (1971):

Value(x,y) = $a_1 r_1 + a_2 r_2 + a_3 r_3 + ... + D$ (a_1, a_2 ,a_3... and D are scalar constants and r_1, r_2 ,r_3... are radial distances from each (x,y) data point)

This surface passes through each data point and is a smooth but irregular surface.

- *Label* is a text label that helps you keep track of multiple SVAS top/bottom condition surfaces.

- *Data* defines the list of *x,y,value* triplets that define the surface. When you click on this button, an input window pops up, where you can enter the data in a comma-delimited, tab-delimited, or space-delimited form like the following:

  x1, y1, value1

  x2, y2, value2

  x3, y3, value3

  x4, y4, value4

  ...

You can enter the data manually, you can paste the data in from other software (e.g. a copied block of cells in an Excel spreadsheet), or you can digitize the surface data within AnAqSim.

When you use a surface for an SVAS top or bottom condition, be sure to read about how the surface is scaled by the number(s) you input for the *Top_flux_or_head* in the SVAS domain or SVAS polygon input.

9.8   Input of Transient Model Parameters

Many parameters in AnAqSim models may vary during a transient simulation. In all cases, the parameters must be constant during each time period, but may change from one time period to another. The following *Model Input* parameters may be transient:

- Discharges at discharge-specified wells and discharge-specified line boundaries

- Heads at head-specified wells and head-specified line boundaries

- Normal fluxes at normal flux-specified line boundaries

- Stages at River line boundaries

- The flux rate (top or bottom) at a flux-specified spatially-variable area source/sink

- The head (top or bottom) at a head-dependent flux-specified spatially-variable area source/sink

For each of these parameters, you may enter one value that applies through the entire simulation, or you may enter multiple values separated by commas, one value per time period. If you enter fewer values than the number of time periods, the last value entered applies to all subsequent time periods.

10   Plot Input Menu

Items under this menu control what appears in the plot created when you select *Make Plot/Level...* from the menu.

10.1   What to Plot

This line of input controls what gets plotted when you select *Make Plot/Level 1, Level2, etc*.

- *Window* defines the coordinates of the rectangular window within which the plot is made. The input for this cell should be four numbers delimited by commas: x minimum, y minimum, x maximum, y maximum. Alternately if this cell is left blank, the entire drawing will be shown, which includes model areas and areas covered by the basemap, if displayed. You may automatically set these coordinates to the current view by selecting *Plot Input/ Set What to Plot/Window to Current View* or using the right-click context menu to do the same. Zoom and pan to get the view you want before making this selection.

- *Basemap* defines if a basemap is to be displayed and how it will be displayed. If *none* is selected, no basemap is drawn. If either *gray* or *color* are selected, AnAqSim will draw the basemap listed under *Basemap_file,* displayed in either gray or in the original colors when a plot is made. The DXF (drawing exchange format) file is generally created by a CAD or GIS program.   A basemap is very helpful for digitizing the coordinates of model features.

- *Basemap_file* defines the DXF file to be used as a basemap. Clicking on this cell opens a dialog where you select the DXF file to form the basemap. When making a plot, AnAqSim searches the path specified here for the DXF file. If that fails, it searches the directory where the model input file (*.anaq) is located a DXF file with the same name.

- *Elements* defines if model elements are drawn, and if so, how they are drawn. If *none* is selected, no elements are drawn. If *Elements* is selected, lines will be drawn along line boundaries and symbols will be drawn at wells, for elements in the level of the plot. For wells, the symbol that is drawn is a "+" symbol that is scaled to the plot window, and a circle that is the exact well radius. If the plot scale is large compared to the well radius, you may see the "+" but not the circle. When you pass the cursor over a well or line element, it is shown with a dotted line and a label pops up explaining what that element is. *Elements (all levels)* does the same, but plots elements in all model levels. *Elements w/details* adds circles at the corners of line boundaries and moving the cursor over one of the circles displays text information about the boundary condition at the corner.   At the start and end points of line boundaries, these circles are slightly offset away from the start or end point, which is helpful when multiple line boundaries

start/end at the same point. *Elements w/details, control points* does the same and also plots symbols at the locations of controls points (points where boundary conditions are met). These are collocation points along line boundaries ("x" symbols) and the basis points for spatially-variable area source/sinks ("+" symbols).

- Checking *Contours* causes the plot to display contours in accordance with the settings defined under *Plot Input/Contour Settings*. Every fifth contour is drawn with a heavier line.

- Checking *Pathlines* causes the plot to display pathlines in accordance with the settings defined under *Plot Input/Pathline Settings* at the locations defined under *Plot Input/Pathlines*.

- Checking *Vectors* causes the plot to display vectors in accordance with the settings defined under *Plot Input/Vector Settings*.

- *SVAS_polygons* defines whether or not to draw the polygons that define the area covered by spatially-variable area sinks (SVAS) by polygon. If drawn, they can be selected and edited by moving vertexes, inserting vertexes, or deleting vertexes.

- *AreaPathline_polygons* defines whether or not to draw the polygons that define the areas where you are starting pathlines within areas. If drawn, they can be selected and edited by moving vertexes, inserting vertexes, or deleting vertexes.

- Checking *Calibration* causes the plot to display calibration information. For head calibration targets, residuals (modeled head - observed head) are plotted at the location of the observed head. The residual is plotted just below a small circle at the target location. When the cursor moves over the plotted residual, the calibration target label, modeled head, and observed head are also shown. Head calibration targets are defined under the *Analysis Input/Calibration Targets/Head* menu, where you can choose to display or not display specific head calibration targets. Velocity calibration data are also displayed, if the *Show* field is checked. For velocity targets, scaled horizontal velocity vectors are displayed - the observed and the modeled. When the cursor is over the velocity vectors, details about the horizontal and vertical velocities are displayed.

- Checking *Analysis_Inputs* causes the plot to display hydrograph points listed under *Analysis Input/Hydrograph Points*, lines entered under *Analysis Input/Transient Line Conditions*, polygons entered under *Analysis Input/Vertical Leakage Over Polygon Area*, and the line entered under *Analysis/Graph Conditions Along a Line*, if there is one. When the cursor moves over any of these features, pop-up text explains what they represent. Hydrograph points and transient line condition lines are only drawn if their level matches the level plotted. The line associated with *Graph Conditions Along a Line* plots regardless of the level of the plot.

10.2   Set Plot Window to Current View

Selecting this sets the plot view coordinates in the *What to Plot/Window* cell to the coordinates of the current plot window. Make this selection after using the zoom functions in the *Plot View Menu* to get the window you want for subsequent plots.

10.3   Set Plot Window to Entire Model

Selecting this sets the plot view coordinates in the *What to Plot/Window* cell to the limits of all elements in the current model, making the window for subsequent plots encompass the entire model extent.

10.4   Contour Settings

This selection lets you customize how contours are drawn.

- *Surface* is a drop-down list used to define what is contoured. You can chose to contour heads (h), the difference in head from this level to the level below (dh), the extraction (areally-distributed discharge per area due to leakage and storage fluxes), or the interface elevation in fresh/salt interface models. The (dh) option allows you to contour the head difference between levels in the active model. If you do this for a plot in level 1, it will contour the head difference between level 1 and level 2. This can be useful for showing vertical head gradients, where flow is upward or downward, etc, but is not used for showing drawdown.

- *Points_Evaluate* defines the number of points where the surface function (e.g. head) is evaluated before being fed into the contouring algorithm. Choosing a larger number here will make a smoother, more detailed contour plot, at the cost of more computation and time to plot.

- *Subtract* gives you the option of creating a contour plot of the difference between the active AnAqSim model and the surface contoured previously with a different run. To use this, you must save the grid of data that was used for the contoured surface of an earlier run. See *Plot View Menu/Plot File Menu/Save Contour Grid Data to Binary File* to learn how to save the surface data. If you select "no", the model contours only the specified surface of the present model. If you specify either of the two yes options, the plot will contour the difference between the surface of the active model and the saved surface (AnAqSim will ask for the file name of the saved surface). If you want a plot of drawdown comparing two AnAqSim runs, you would set *Surface* to "h" and set *Subtract* to one of the "yes" options, depending on whether you want drawdown displayed negative or positive.

- *Increment* defines the contour increment - the difference between adjacent contour levels. For example, if the contoured surface data ranged from 12.3 to

14.7, you left the *Minimum* and *Maximum* cells blank, chose an *Increment* of 0.1, the levels contoured would be 12.3, 12.4, 12.5...14.7. AnAqSim automatically uses a heavier line for every 5th contour (e.g. 12.5, 13.0, 13.5...) and labels these. If you set *Increment* to zero or a negative number, the increment will be determined automatically so that there will be about 25 contours total. This can be a good idea when you are uncertain of the range of values to contour.

- *Minimum* sets the minimum contour level. If this is left blank, the minimum contour level will be a multiple of *Increment* that is closest to the minimum value on the contoured surface.

- *Maximum* sets the maximum contour level. If this is left blank, the maximum contour level will be a multiple of *Increment* that is closest to the maximum value on the contoured surface.

10.5   Pathline Settings

These settings control the drawing of all flow pathlines you enter under Plot Input/Pathlines. In steady simulations, the pathlines are traced through a steady flow field. In transient simulations, pathlines are generally traced through the simulated transient flow field, and time along a pathline is synchronized with the simulation time. The exception is if you check *Freeze_transient* pathlines, as described below. The rest of this discussion of transient pathlines assumes *Freeze_transient* is not checked. Each time step in a transient model has a flow solution that, for the purposes of pathline tracing, applies from the start to the end of the time step. Transient pathlines are only traced during the simulated time window. Transient pathlines terminate when one of the following conditions occurs: 1) the simulation time along the pathline is less than 0 or greater than the total simulation time (sum of time period lengths), 2) the total elapsed time exceeds *Total_time* or, 3) the pathline exits the model or enters a well or linesink. Steady pathlines terminate when condition 2 or 3 in the previous sentence is reached.

- *Step_size* defines the size of steps used in the algorithm that traces pathlines. Pathlines are composed of many small straight line segments, and the length of these little segments is defined by *Step_size*. The size of segments equals *Step_size* times the largest dimension (x or y) of the plot window size, which is either a) the distance across the window specified under *Plot Input/What to Plot (Window)*, or if that field is blank, the distance across the entire model (the minimum window that spans all line boundaries). So, the definition of it changes as you change the settings of *Plot Input/What to Plot/Window*. Using a smaller *Step_size* causes smoother, more accurate pathlines at the cost of more computation time.

- If *Time_markers* is checked, arrows are plotted at specific time intervals along the pathlines drawn. This is useful for defining the capture zones of wells within certain time limits, or simulating solute transport times. When you move the

cursor over an arrow, a text box pops up to display information about the elapsed time, domain, and elevation of the pathline at the location of the arrowhead. This allows you to know the 3D configuration of a pathline. The scale of the arrows is set as a fraction of the plot window size (similar to the step size as discussed above.

- *Time_marker_increment* is the amount of elapsed time between arrows that are drawn, if *Time_markers* is checked.

- *Total_time* is the total amount of time allowed on pathlines. This can be set to show, for example, the limits of the 2 year capture zone for a water supply well.

- *Start_time* is the simulation time at the starting points for pathlines in transient simulations.   Simulation time starts at zero at the beginning of a transient simulation. For example, in a 300 day simulation, you could have the starting time for well pathlines be 300 days and trace pathlines upstream and the simulation time would decrease along the pathline. Alternatively, pathlines traced in the downstream direction have increasing simulation time along the pathline.

- If *Freeze_transient* is checked and this is a transient simulation, the pathlines will not be traced through the transient flow field, but instead will be traced through a "frozen" snapshot of flow field at the time *Start_time*, with pathlines traced as though this flow field existed indefinitely like a steady state flow field. This is generally not recommended, since the frozen flow field doesn't actually exist for longer than an instant. This option is offered because capture zone delineation rules in some states (e.g. Massachusetts) require such an analysis.

- *Capture_constrain, if* checked, allows you to plot only the pathlines that are captured by certain wells or internal line boundaries specified in the next two input items. Capture constrain only applies to pathlines traced in the downstream direction. This can be helpful for defining the capture zone of wells, stream reaches, drains, etc. For example, to define the zone of contribution to a well, you could start numerous pathlines at the water table within a polygon area, and constrain the displayed pathlines to those captured by the well. The staring points of those pathlines will define the zone of recharge captured by the well.

- *Capture_wells* allows you to select which well(s) you want the capture constraint to apply to. If you click the *Select* button, you can select one or more wells from among the list of wells - their labels will be displayed in this list. All three well types are included in this list: discharge-specified, discharge-specified (multi-domain), and head-specified.   You should not mix well pathlines emanating from well with plots involving capture constraint at the same well. The well pathlines emanating from the well are not considered "captured" because in the tracing they start at the well, but don't end at the well.

- *Capture_lines* allows you to select which internal line boundaries you want the capture constraint to apply to. If you click the *Select* button, you can select one or more line boundaries from among the list of line boundaries - their labels will be displayed in this list. These types of line boundaries are included in this list: head-specified (internal, not external), discharge-specified, river, drain/fracture.

Pathlines can be displayed with three aspects:

- Starting points are plotted as circles and there is pop-up information about the starting point if you hover the mouse over the circle.

- Pathline traces.

- Arrowheads at each time marker, with pop-up information about the pathline at that point if you hover the mouse over it.

You can control which of these three is displayed in the plot by selecting *View Manager* from the *Plot View Menu*. You can toggle the display of these on/off by clicking on the *Freeze* column in the View Manager.

10.6   Vector Settings

These settings control the drawing of vectors. Vectors are computed using the domain's function for the discharge vector. When vectors are drawn, the center of the vector is at the spot where the vector was computed.

- *Vector_Type* is a drop-down list of possible vector types to draw: average linear velocity (v) [L/T], specific discharge (q) [L/T], or domain discharge (Q, equals q times saturated thickness) [L^2/T].

- *Points_Evaluate* defines the number of points where the vector will be computed and displayed. Select values in the lower end of the range for a plot that is not cluttered with too many vectors.

- *Scale_Factor* is a factor that controls the scale that the vectors are drawn to. Larger numbers here produce larger vectors. A value of 1 makes it so the largest vector drawn has a length equal to the spacing between points where vectors are evaluated.

10.7   Pathlines

This menu is where you specify the starting points and tracing directions of pathlines that are drawn. You can specify single pathlines, pathlines distributed along a line, or pathlines distributed along a circle. Well pathlines are like circle pathlines but linked to a particular well. Pathlines are demonstrated in detail in the tutorial videos at the website.

10.7.1 Single

This table allows input for single pathlines.

- *Label* is a text label that helps you keep track of multiple single pathlines.

- *Direction* is the direction that pathlines are traced from the starting point and it may be either upstream or downstream.

- *X,Y* are the coordinates of the starting point for the pathline. The coordinates are two real numbers delimited by a comma. You may digitize the coordinates with the *Digitize/Point* selection in the plot view menu.

- *Start_level* is the level(s) of the pathline at the start point(s). It is easiest to understand this parameter with some examples. A value of 1.0 starts a pathline at *X,Y* at the top of the domain that is level 1 at this location. A value of 2.25 starts a pathline at the elevation where 25% of the saturated thickness is above and 75% of the saturated thickness below in the domain that is level 2 at *X,Y*. A value of 2.99 starts a pathline with 99% of the saturated thickness above and 1% of the saturated thickness below, very near the base of the level 2 domain at *X,Y*. Pathlines are plotted for all levels encountered along the path, not just the start level or the plotted level which applies to the contours, vectors and information shown to the left of the plot. A single *Start_level* may be entered, or multiple levels may be entered separated by commas which starts an array of pathlines with the same x,y coodinates but at different levels. For example, you could enter "2.1, 2.3, 2.5, 2.7, 2.9" to start pathlines at five different elevations in level 2.

- *Show* is a checkbox that allows you to selectively display or not display this single pathline. In order for this pathline to appear in a plot, this must be checked **and Pathlines must be checked** in *Plot Input Menu / What to Plot*.

10.7.2 Line

This allows starting a number of pathlines equally spaced along segments of a polyline.

- *Label* is a text label that helps you keep track of multiple line pathlines.

- *Direction is* the same as defined for Pathline/Single.

- *Coordinates* are the coordinates of the polyline along which the pathlines start. Edit the coordinates by clicking the *Edit* button in this column. The coordinates are two or more lines, each with two real numbers delimited by a comma (X, Y). You may digitize the coordinates with the *Digitize/Polyline* selection in the plot view menu and then paste them in.

- *Number* is the number of pathlines to start in each segment of the polyline. This number of start points are equally spaced along each segment.

- *Start_level* is the level(s) of the pathlines at their start points(s). It is the same as defined for Pathline/Single. A single *Start_level* may be entered, or multiple levels may be entered separated by commas which starts an array of pathlines with the same x,y coodinates but at different levels. For example, you could enter "2.1, 2.3, 2.5, 2.7, 2.9" to start pathlines at five different elevations in level 2.

- *Show* is a checkbox that allows you to selectively display or not display these pathlines. In order for these pathlines to appear in a plot, this must be checked **and *Pathlines* must be checked** in *Plot Input Menu / What to Plot*.

10.7.3 Area

This allows starting a number of pathlines equally spaced within an area defined by a polygon. This can be useful for tracing pathlines from a source area, or for defining capture zones for wells.

- *Label* is a text label that helps you keep track of multiple area pathlines.

- *Direction is* the same as defined for Pathline/Single.

- *Coordinates* are the coordinates of the polygon that defines the area within which the pathlines start. Edit the coordinates by clicking the *Edit* button in this column. The coordinates are multiple lines, each with two real numbers delimited by a comma (X, Y). You may digitize the polygon coordinates with the *Digitize/Polyline, Digitize/Circle,* or *Digitize/Ellipse* selections in the plot view menu and then paste them in. You may also later edit the polygon graphically, moving, inserting, and deleting vertexes.

- *Nesting_level* is a parameter that allows you to nest several area pathline polygons inside each other, with different start point spacings. If you are not nesting or overlapping these, specify 1 for all. If you want to place a smaller polygon with denser or less dense spacing inside another polygon, use nesting level 1 for the larger polygon and nesting level 2 for the smaller one inside of it. You may nest up to 5 levels to drill down to finer basis point spacing in the key areas (e.g. source area). The polygons with the higher nesting level overwrite the spacing and other inputs of lower nesting levels.

- *Spacing* defines the distance between pathline start points. Start points are spread out on a hexagonal array of points based on this spacing.

- *Start_level* is the level(s) of the pathlines at their start points(s). It is the same as defined for Pathline/Single. A single *Start_level* may be entered, or multiple levels may be entered separated by commas which starts an array of pathlines with the same x,y coodinates but at different levels. For example, if the source area you were trying trace spanned level 2, you could enter "2.1, 2.3, 2.5, 2.7, 2.9" to start pathlines at five different elevations in level 2.

- *Show* is a checkbox that allows you to selectively display or not display these pathlines. In order for these pathlines to appear in a plot, this must be checked **and *Pathlines* must be checked** in *Plot Input Menu / What to Plot*.

10.7.4 Circle

Circle pathlines start equally spaced on a circle that the user specifies. For wells, the well pathline is a better choice, since it is linked to the well and will move with the well if its location is changed.

- *Label* is the text label of the pumping well you want to trace from.

- *Direction is* the same as defined for Pathline/Single.

- *X,Y* are the coordinates of the center of the circle. The coordinates are two real numbers delimited by a comma. You may digitize the coordinates with the *Digitize/Point* selection in the plot view menu.

- *Radius* is the radius of the circle. For a well, make sure this radius is larger than the well's radius, otherwise the pathlines will be terminated before they can start.

- *Start_level* is the level(s) of the pathlines at their start points(s). It is the same as defined for Pathline/Single. A single *Start_level* may be entered, or multiple levels may be entered separated by commas which starts an array of pathlines with the same x,y coodinates but at different levels. For example, you could enter "2.1, 2.3, 2.5, 2.7, 2.9" to start pathlines at five different elevations in level 2.

- *Show* is a checkbox that allows you to selectively display or not display these pathlines. In order for these pathlines to appear in a plot, this must be checked **and *Pathlines* must be checked** in *Plot Input Menu / What to Plot*.

10.7.5 Well

Well pathlines are like circle pathlines, but are linked to a specific well. When you move a well graphically, the reference to the well is maintained and pathlines will be drawn correctly. The circle that the pathlines start on is centered on the well center and has a radius that is 1.2 times the well radius. The direction of tracing is upstream if the well is extracting and downstream if the well is injecting.

- *Well Label* is the text label of the pumping well you want to trace from. Double click this cell to select from a list of all well labels.

- *Number* is the number of pathlines to start. Pathline start points are equally spaced around the circle centered on the well.

- *Start_level* is the level(s) of the pathlines at their start points(s). It is the same as defined for Pathline/Single. A single *Start_level* may be entered, or multiple levels

may be entered separated by commas which starts an array of pathlines with the same x,y coodinates but at different levels. For example, you could enter "2.1, 2.3, 2.5, 2.7, 2.9" to start pathlines at five different elevations in level 2.

- *Show* is a checkbox that allows you to selectively display or not display these pathlines. In order for these pathlines to appear in a plot, this must be checked **and *Pathlines* must be checked** in *Plot Input Menu / What to Plot*.

11   Analysis Input Menu

Items under this menu allow you analyze the results of a simulation after the system of equations is solved (*Solve*).

11.1   Hydrograph Points

These allow you to track head (or drawdown) vs. time at specific points during a transient simulation. They may be used to check the simulated water level at some other point such as an observation well location. After solving, you can make hydrographs with the model-simulated head (or drawdown) and observed values read from a file; this is done with *Analysis / Graph Head Hydrographs* or *Analysis / Graph Drawdown Hydrographs*.

- *Label* is a text label that helps you keep track of multiple hydrograph points.

- *Level* is the model level of the hydrograph point.

- *X,Y* are the coordinates of the hydrograph point. The coordinates are two real numbers delimited by a comma. You may digitize the coordinates with the *Digitize/Point* selection in the plot view menu.

- *Show* is a checkbox that allows you to selectively include or exclude this particular hydrograph point in written or graphed outputs.

A similar functionality is available in the plot context menu (right-click over the plot), which allows you to create hydrographs of all model levels at any location in a transient model. However, hydrographs created with the plot context menu lack initial (t=0) heads, and the location is determined by the cursor position.

The locations of hydrograph points can be included in plots. These locations may also be edited graphically, like can be done with well locations: click on a hydrograph point, then drag the purple square to the new location.

11.2   Transient Line Conditions

This allows you to track head or extraction along a line during a transient simulation. Tracking extraction along a line is useful for checking the accuracy of the governing equation approximation in transient simulations. Tracking head along a line allows you to see how head profiles evolve during a transient simulation. After solving, use these menu selections to make graphs: *Analysis/Graph Transient Line Conditions.*

- *Label* is a text label that helps you keep track of multiple transient line conditions.

- *Level* is the model level of the line.

- *Coordinates* are the coordinates of the end points of the line.   Click the Edit button to edit coordinates. The coordinates are two lines, each with two real numbers delimited by a comma (X, Y). You may digitize the coordinates with the *Digitize/Polyline* selection in the plot view menu and then paste them in.

- *Points_on_line* is the number of points on the line where head and extraction are computed. More points yield smoother, more accurate profiles at the cost of more computation.

These lines can be displayed on plots. These lines may also be edited graphically, like can be done with line boundaries: click on a line, then drag the purple squares at the vertexes to new locations.

11.3   Calibration Targets

Here is where you input information about targets for calibration. Targets may be heads or head differences between two locations.

11.3.1   Head

- *Label* is a text label that helps you keep track of head calibration targets.

- *Level* is the model level of the target.

- *X,Y* are the coordinates of the target. The coordinates are two real numbers delimited by a comma. You may digitize the coordinates with the *Digitize/Point* selection in the plot view menu.

- *Observed_Head* is the target head.

- *Time* is the time of the observation for a transient model. This time is measured from time = 0 at the start of the simulation. In a steady model, this parameter is ignored. In a transient model, you may input head targets at the same location but different times. The Head Difference calibration feature may be used to calibrate to temporal variations in head (e.g. rate of drawdown) during a transient simulation, in addition to spatial head differences.

- *Show* - if this is checked, this particular calibration point data will display when a plot is made with *What to Plot / Head Calibration* is checked. This allows you to make a plot with just certain calibration points posted (e.g. those in a one domain, or one model level, or at a certain time in a transient run).

Head calibration targets may be imported from spreadsheets like Excel. Start by creating a spreadsheet table with the same columns as AnAqSim's head calibration table. Then highlight and copy the block of data you want to paste, like is shown in the first image below for an Excel spreadsheet. Then right-click over the AnAqSim head calibration table

and select *Paste New Rows*. The new rows will appear as shown in the second image below. If your data has X and Y in separate columns, you can use Excel's CONCATENATE function to combine the X and Y columns with a "," between them into one column.

	A	B	C	D	E	F
1	Label	Level	X,Y	Observed_Head	Time	Show
2	MW-1	1	1080, 8743	245.87	0.00	TRUE
3	MW-2	2	876, 7659	243.91	0.00	TRUE
4	MW-3	1	1173, 6236	238.89	0.00	TRUE
5	MW-4	2	914, 7847	231.34	0.00	TRUE
6	MW-5	3	1006, 7572	147.02	0.00	TRUE
7						

Data: calibration heads

Label	Level	X,Y	Observed_Head	Time	Show
MW-1	1	1080, 8743	245.87	0.0	✔
MW-2	2	876, 7659	243.91	0.0	✔
MW-3	1	1173, 6236	238.89	0.0	✔
MW-4	2	914, 7847	231.34	0.0	✔
MW-5	3	1006, 7572	147.02	0.0	✔
					☐

11.3.2 Head Difference

This allows comparison of model-calculated head differences to target head differences. The difference is computed between two head targets listed in the *Analysis Input/Calibration Targets/Head* table.   This may be used to calibrate to temporal variations in head at one location during a transient simulation (e.g. rate of drawdown) , or to spatial head differences (e.g. head gradients). The labels are selected from drop-down lists generated from the head calibration targets data.

- *Label1* is the text label for the first head calibration target.

- *Label2* is the text label for the second head calibration target.

11.3.3 Velocity

- *Label* is a text label that helps you keep track of velocity calibration targets.

- *Level* is the model level of the target.

- *X,Y* are the coordinates of the target. The coordinates are two real numbers delimited by a comma. You may digitize the coordinates with the *Digitize/Point* selection in the plot view menu.

- *Observed_Horizontal_Velocity* is the target magnitude of the horizontal component of the average linear velocity.

- *Observed_Horizontal_Direction* is the target direction of the horizontal component of the average linear velocity. This is measured in degrees from the x axis, positive in the counter-clockwise direction.

- *Observed_Vertical_Velocity* is the target vertical component of the average linear velocity. Positive is upward, negative is downward.

- *Time* is the time of the observation for a transient model. This time is measured from time = 0 at the start of the simulation. In a steady model, this parameter is ignored. In a transient model, you may input head targets at the same location but different times. The Head Difference calibration feature may be used to calibrate to temporal variations in head (e.g. rate of drawdown) during a transient simulation, in addition to spatial head differences.

- *Show* - if this is checked, this particular calibration point data will display when you write calibration data to the run log and will display on the plot when a plot is made with *What to Plot / Head Calibration*. This allows you to make a plot with just certain calibration points posted (e.g. those in a one domain, or one model level, or at a certain time in a transient run).

This option is useful for calibrating a model to a known discharge direction indicated by an observed chemical plume. You can specify the direction of the observed horizontal velocity parallel to the observed plume and compare this to the modeled horizontal velocity direction. If you also know the magnitude of velocity in the plume, you can compare modeled to observed horizontal and vertical velocity components.

11.3.4 Head-specified Well Discharges

Here you select one or more head-specified wells to compare their simulated total discharge to some target or observed value of total discharge. This could be used to calibrate to an observed total discharge from a group of flowing artesian wells, for example.

- *Label* is the label give to this well or group of wells.

- *Wells_Selected* shows you a list of the head-specified wells from which you select the one(s) that you would like to calibrate to.

- *Observed_Discharge* is the observed or target discharge of this well or group of wells.

- *Time* is the time in a transient simulation (not used in a steady simulation).

11.3.5 Internal Line Boundary Discharges

Here you select one or more internal discharging line boundaries (head-specified internal or river) to compare their simulated total discharge to some target or observed value of total discharge. This could be used to calibrate to an observed discharge at a stream gauge or discharge from a pit dewatering system, for example.

- *Label* is the label give to this well or group of wells.

- *LineBoundaries_Selected* shows you a list of the internal head-specified line boundaries and river line boundaries from which you select the one(s) that you would like to calibrate to.

- *Observed_Discharge* is the observed or target discharge of this line boundary or group of line boundaries.

- *Time* is the time in a transient simulation (not used in a steady simulation).

11.4   Vertical Leakage Over Polygon Area

This feature allows numerical computation of vertical leakage discharges integrated over a polygon area. The area may be defined as the area covered by a specified domain or the area covered by a user-defined polygon. The discharges are computed by evaluating the vertical leakage rate [L/T] at a large number of regularly-spaced points, and numerically integrating contributions over the small increments of area each point represents. The total leakage discharge [L^3/T] over the area is reported. This tool will examine vertical leakages for each level that exists within the domain or polygon specified. For example, if levels 1,2,3, and 5 exist in the area of the polygon, the output will include:

- downward leakage into the topmost level of the model, regardless of whether the topmost level is level 1, 2, etc.

- downward leakage out of level 1

- downward leakage out of level 2

- downward leakage out of level 3

- downward leakage out of level 5

After solving, use Analysis / Write Leakages Over Polygon Areas to Run Log to view computed total leakage discharges. This output is also available in automated runs.

Input parameters for this feature are:

- *Label* is a text label that helps you keep track of multiple polygon areas.

- *Polygon_Source* selects whether your polygon will be defined as the extent of a model *Domain* or by the *Cooordinates* of a user-defined polygon, which could cross domain boundaries.

- *Points_Evaluated* is the approximate number of points evaluated (if your polygon is a rectangle parallel to the x-y axes, this number is the actual number of points, but if your polygon is irregularly-shaped, the actual number of points will be less than what is specified here. The algorithm sets up a spacing of points so that the specified number of points fills a rectangle that encompasses the entire polygon.   More points yields more accurate estimates of integrated leakage discharge at the cost of more computation.

- *Domain* sets the domain that defines the polygon area, if you choose Domain as the *Polygon_Source*. If you choose Cooordinates as the polygon source, this input will not be used and may be left blank.

- *Coordinates* are the coordinates of the polygon area, if you choose Cooordinates as the *Polygon_Source*. If you choose Domain as the polygon source, this input will not be used and may be left blank. Click the *Edit* button to edit coordinates. The coordinates are multiple lines, each with two real numbers delimited by a comma (X, Y) to represent the vertex of the polygon. You may digitize the coordinates with the *Digitize/Polyline* selection in the plot view menu and then paste them in here.   The coordinates may define a closed polygon (last point = first point), or it may omit the closing point (AnAqSim will close the polygon for you).

These polygons can be displayed on plots. They may also be edited graphically, like can be done with line boundaries: click on a polygon, then drag the purple squares at the vertexes to new locations. You may also insert or delete vertexes once the polygon is selected.

12   Solve

Pressing *Solve* begins the process of solving the system of equations associated with your model. The program constructs the system of equations and then invokes a direct LU solver to determine the unknown strength parameters associated with wells, line elements, and area sink elements. Boundary conditions like river boundaries in unconfined aquifers generate nonlinear equations. When a model contains some nonlinear equations, the solution is found by repeated iterations, where the non-linear equations are linearized (an approximation) with the linearization updated at each iteration. The program displays iteration information in the run log as iterations proceed and when they terminate due to one of two conditions: 1) the solution converged within the *check settings*, or 2) the solution did not converge within the check settings but stopped iterating after the *maximum number of iterations*.   The solve process is executed on a background thread, so you may continue with some input editing tasks as the solve process runs. You can abort the solve process by selecting *Cancel Solve*.

Solve must be executed after any change to the *Model Input* so that results examined in the *Make Plot* or *Analysis* menus reflect the current input. Changes to the *Plot Input* do not require re-solving. In transient models, changes to *Analysis Input/Hydrograph Points* or *Analysis Input/Transient Line Conditions* requires you to reestablish the initial heads files and to Solve again before running related items under the *Analysis* menu.

Solving takes longer for more complex simulations with more equations. It is good modeling practice to start with a relatively simple model and incrementally add complexity, so you can monitor the level of complexity, number of equations, and solve times as you proceed. The spatially-variable area source/sinks can add large numbers of equations to the system if the spacing of basis points is small. To save computation time, you can increase the basis point spacing of spatially-variable area sinks and reduce the number of parameters per line on line boundaries. When you want to make a final, presentation-quality model, you can alter these for more accuracy and better aesthetics.

Another way to keep computation minimal and start simple is to make as much of the model linear as possible (linear boundary conditions can solve in a single iteration, as opposed to non-linear boundary conditions). You can do this by making domains confined (fixed T and saturated thickness). You can also employ head-specified line boundaries (linear) instead of river line boundaries (often nonlinear). For complex problems, it probably makes sense to do most of the modeling work in with linear features and then change to nonlinear features later in the modeling process.

You may adjust *Solve Settings* and *Check Settings* to control the *Solve* process.

13   Cancel Solve

This allows you to abort the ongoing solve process. Use this if you don't want to wait for solve to finish and you know the input is not what you want. If you select *Make Plot/All Selected Features* after canceling the solve process, AnAqSim will attempt to make a plot with the available solution. For example, a transient solution canceled after 5 time steps were complete will be able to display plots and pathlines up through this point in the simulation. In a steady simulation, the plot will show the last iteration results before canceling.

14   Make Plot Menu

Selections here cause a plot to be generated. The view is automatically shifted to the map view with any of these selections.

14.1   All Selected Features, Same Level/Time

This creates a map-view plot of the model results, using your settings from the *Plot Input* menu. The level and time of the plot are the same as in the prior plot. If this is the first plot made, it defaults to level 1 at the first time step. To use this selection, you must have solved a model.

Everything shown in the plot will be specific to the selected level and time step , except for pathlines, which may traverse multiple levels and progress through time. Regardless of the level and time selected, entire pathlines are plotted.

14.2   All Selected Features, Choose Level/Time

After selecting a model level and time step (for transient models) in the pop-up window, this creates a map-view plot of the model results, using your settings from the *Plot Input* menu. To use this selection, you must have solved a model.

Everything shown in the plot will be specific to the selected level and time step , except for pathlines, which may traverse multiple levels and progress through time. Regardless of the level and time selected, entire pathlines are plotted.

14.3   Model Elements Only

This selection is used if you are part way through constructing a model and want to digitize features or see what model input features you have so far, without solving the partially-constructed system. This allows you to see the layout of model elements and of control points (points where boundary conditions are met). The control points are collocation points along line boundaries ("x" symbols) and the basis points for spatially-variable area source/sinks ("+" symbols). To see control points and basis points, make sure to select "elements w/details, control points" under Plot Input / What to Plot / Elements.

Along line boundaries, there are circles at the line segment end points. If you move the cursor over one of these circles, information about the line boundary condition at that corner is displayed. At the start and end points of line boundaries, these circle are slightly offset away from the start or end point, which is helpful when multiple line boundaries start/end at the same point.

The pop-up window allows you to select the model level of the elements you want to display.

When you make a plot this way, there is no valid solution to examine, since the *Solve* step is bypassed. So, you can't use any of the post-solve analysis features such as *Make Plot /All Selected Features* or any of the options under the *Analysis* menu, and some of the options under the *Plot Context* menu.

14.4   SVAS Top/Bottom Condition Surface

Use this to contour and examine a SVAS top/bottom condition surface that you have defined. When you select this, a dialog will pop up that asks you to select one of the surfaces you have defined. Then a plot will be made that shows contours of the surface using contours settings defined under Plot Input/Contour Settings.   The plot will also show dark blue circles at each surface data point location, and if you move the mouse over the circle, it will display the specified value at that data point.

Depending what you have selected under Plot Input/What to Plot, the plot may or may not also show the basemap, elements, and SVAS polygons.

When you make a plot this way, there is no valid solution to examine, since the *Solve* step is bypassed. So, you can't use any of the post-solve analysis features such as *Make Plot /All Selected Features* or any of the options under the *Analysis* menu, and some of the options under the *Plot Context* menu.

15   Analysis Menu

Under this menu there are many choices for analyzing a model after the system of equations has been solved. Remember to *Solve* the system after making input changes and before using this menu. Some choices generate graphs, others generate text output to the run log. Many of the functions under this menu are demonstrated in detail in the tutorial videos at the website.

15.1   Check Boundary Conditions at Latest Iteration

Selecting this causes the program to run through a check of all boundary conditions at the end of the most recent iteration (solved model of last time step if it converged, or last iteration in the case of a model that didn't converge), to see if they are met within the tolerances specified under *Model Input/Solution/Check Settings*. The results are written to the run log and the view is switched to the view with the run log. This can be useful to see which boundary conditions are causing non-convergence, if your model doesn't converge. It may steer you to relax certain check settings or reconfigure the model.

In the following example, all of the boundary conditions met the check tolerances, so no specific boundary condition is listed:

Run Log:

Begin check of boundary conditions 5/31/2011 4:57:04 PM ...
The following will list just specific boundary conditions that did not meet the following criteria specified in model input / solve:
  1) Head conditions [L]: deviation of modeled values are no more than 1.000E-003
  2) Discharge (Q) conditions [L^3/T]: deviation of modeled values are no more than 1.000E-003
  3) Discharge per length (Q/L or Qn) conditions [L^2/T]: deviation of modeled values are no more than 1.000E-004
  4) Extraction (Q/area) conditions [L/T]: deviation of modeled values are no more than 1.000E-006
Checking 1 head-specified line boundaries...
Checking 3 river line boundaries...
Checking 1 Qn-specified line boundaries...
Checking 1 inter-domain line boundaries...
Checking 231 basis points at spatially variable area sinks...
End of list - all boundary conditions have been checked.

After decreasing the Qn_check_tolerance to 2×10^{-9} and the Extraction_check_tolerance to 4×10^{-7}, the same model had some the boundary conditions exceeding their check tolerances:

Run Log:

Begin check of boundary conditions 5/31/2011 5:01:00 PM ...
The following will list just specific boundary conditions that did not meet the following criteria specified in model input / solve:
 1) Head conditions [L]: deviation of modeled values are no more than 1.000E-003
 2) Discharge (Q) conditions [L^3/T]: deviation of modeled values are no more than 1.000E-003
 3) Discharge per length (Q/L or Qn) conditions [L^2/T]: deviation of modeled values are no more than 2.000E-009
 4) Extraction (Q/area) conditions [L/T]: deviation of modeled values are no more than 4.000E-007
Checking 1 head-specified line boundaries...
Checking 3 river line boundaries...
2.012E-009 = modeled discharge per length deviation at river control point at (133.242, -79.310) in domain well upper
2.110E-009 = modeled discharge per length deviation at river control point at (137.418, -73.785) in domain well upper
Checking 1 Qn-specified line boundaries...
Checking 1 inter-domain line boundaries...
Checking 231 basis points at spatially variable area sinks...
5.571E-007 = modeled deviation in extraction rate at spatially-variable basis point at (225.764, -55.658) in domain well upper
5.571E-007 = modeled deviation in extraction rate at spatially-variable basis point at (225.764, -55.658) in domain well middle
End of list - all boundary conditions have been checked.

15.2   Head-Specified Well Discharges

Under this menu, selecting *Write Discharges to Run Log* writes a listing of head-specified well discharges to the run log. If the model is steady, it will list one discharge per well. If the model is transient, it will list *period, step, time, and discharge* at each time step. The time listed is the time at the end of the step. The discharge listed applies from the start to the end of the step. Discharges of such wells are not known prior to solving the system (*Solve*).

If you select *Graph Transient Discharges*, a dialog pops up where you select one well, and a graph is made of the discharge vs. time at the selected well. Discharges are constant during a time step, so plots show abrupt changes in discharge between steps. Exporting the graph or the underlying data is described here.

15.3   Discharge-Specified Well Heads

Under this menu, selecting *Write Heads to Run Log* writes a listing of heads at all discharge-specified wells to the run log. This is head at the specified well radius. If the model is steady, it will list one head per well. If the model is transient, it will list *period, step, time, and head* at each time step. Heads at such wells are not known prior to solving the system (*Solve*).

If you select *Graph Transient heads*, a dialog pops up where you select one well, and a graph is made of the head vs. time at the selected well.   Exporting the graph or the underlying data is described here.

15.4   Internal Head-Specified Line Boundary Discharges

Under this menu, selecting *Write Discharges to Run Log* writes a listing of internal head-specified line boundary discharges to the run log. If the model is steady, it will list one

total discharge per line boundary. If the model is transient, it will list total discharges vs. time, with *period, step, time, and discharge* at each time step. The time listed is the time at the end of the step. The discharge listed applies from the start to the end of the step. The discharges of such line boundaries are not known prior to solving the system (*Solve*).

If you select *Graph Transient Discharges*, a dialog pops up where you select one internal head-specified line boundary, and a graph is made of the total discharge vs. time at the selected line boundary.   Discharges are constant during a time step, so plots show abrupt changes in discharge between steps. Exporting the graph or the underlying data is described here.

15.5   River Line Boundary Discharges

Under menu, selecting *Write Discharges to Run Log* writes a listing of river line boundary discharges to the run log. If the model is steady, it will list one total discharge per line boundary. If the model is transient, it will list *period, step, time, and total discharge* at each time step. The time listed is the time at the end of the step. The discharge listed applies from the start to the end of the step. Discharges are negative for flow out of the domain into the river and positive for flow from the river into the domain.

AnAqSim will also print the sum of a group of river line boundary discharges, if your river line boundaries have labels with a common root. A common root consists of any label that contains the common character string "_____:#". For example, if you had three river line boundaries and their labels were *dark creek:#1, dark creek:#2,* and *dark creek:#3*, the discharges of each of these 3 would be printed as well as the sum total discharge of these 3. The discharges of river line boundaries are not known prior to solving the system (*Solve*).

If you select *Graph Transient Discharges*, a dialog pops up where you select one river line boundary, and a graph is made of the total discharge vs. time at the selected line boundary.   Discharges are constant during a time step, so plots show abrupt changes in discharge between steps. Exporting the graph or the underlying data is described here.

15.6   Multi-Domain Well Dicharges by Domain

Under menu, selecting *Write Discharges to Run Log* writes a listing of the discharge contributed by each domain in a discharge-specified multi-domain well to the run log. If the model is steady, it will list one discharge per domain. If the model is transient, it will list *period, step, time, and discharge* at each time step in each domain.   The time listed is the time at the end of the step. The discharge listed applies from the start to the end of the step. The discharges in each domain are not known prior to solving the system (*Solve*).

If you select *Graph Transient Discharges*, a dialog pops up where you select one discharge-specified multi-domain well, and a graph is made of the discharge contributed by each domain vs. time at the selected well. Discharges are constant during a time step, so plots show abrupt changes in discharge between steps. Exporting the graph or the underlying data is described here.

15.7 Graph Conditions Along a Line

This selection allows you to create graphs of head, domain elevations, interface elevations, aquifer discharge, or extraction rate along a line. When selected, a dialog box pops up which requires the following input.

- *Model Level(s)* selects the model level(s) of the graphed condition. You may select one or multiple levels (e.g. you may want to plot a profile of heads in all levels along the line).

- *Period, Step, Time* selects the snapshot in time that you graph for a transient run. Starting with release 2016-2, this allows selection of the solution at the end of any time step. Prior to 2016-2, only the final time step solution was available for this function.

- *Line Coordinates* are the coordinates of the end points of the line. The coordinates are two lines or input, each with two real numbers delimited by a comma (X, Y). You may digitize the coordinates with the *Digitize/Polyline* selection in the plot view menu and then paste them in. This line can be displayed on plots. The line may also be edited graphically, like can be done with line boundaries: click on a line, then drag the purple squares at the vertexes to new locations.

- *Number of points along the line* is the number of points where the head, normal discharge, or extraction is evaluated along the line. More data points make for a smoother plot.

- Select whether you want to graph a *Vertical profile, Discharge tangent to line, Discharge normal to line,* or *Extraction* vs. distance along the line.

If you select *Vertical profile*, you have the option of showing *head, domain boundaries, fresh/salt interface elevations*, *pathlines,* and/or *pathline time markers*. Depending on what you select under *Level*, you will make a profile of the selected levels.

There are two options for plotting fresh/salt interface profiles. The *Dupuit* option computes the interface position using the model-simulated head for the domain, regardless of the elevation of the interface within the domain. This assumes that there is no vertical resistance to flow within a domain and no vertical head gradients within a domain (this is commonly called the Dupuit approximation, which is inherent in AnAqSim's domain discharge model). The *Dupuit* interface position is always computed

using the model-simulated head, and assuming pressure equilibrium between this fresh water head and static salt water.

The *Dupuit* plotting option can be used for single- and multi-level domain interface models. In multi-level models where there are significant head differences between levels, the *Dupuit* option will generate an interface that has discontinuities at boundaries between levels. This saw-tooth shaped interface is consistent with the Dupuit and Ghyben-Herzberg approximations used in AnAqSim. With the *Dupuit* approximation, head jumps abruptly at horizontal boundaries between domains at different levels, so the interface elevation computed with the Ghyben-Herzberg equation must also jump.

Such discontinuities in the interface may be realistic for cases where a thin aquitard, included in the AnAqSim model by adjusting the K3 of an aquifer domain above or below, is the main cause of vertical resistance between levels. On the other hand, if the main source of vertical resistance is not an aquitard but the actual K3 of the domains involved, the interface discontinuities are just an artifact of the Dupuit approximation. These discontinuities can be smoothed by selecting the *Non-Dupuit* plotting option.

The *Non-Dupuit* option uses heads that are calculated in a post-processing step, heads that do vary with elevation within a domain. In multi-level models simulating vertical leakage between levels, there is information available to approximate the vertical distribution of head within a domain. Using the simulated heads in the domain and in overlying or underlying domains, and the vertical hydraulic conductivity of the domains (K3), and assuming that the simulated domain head applies at the mid-height of a domain, is is possible to construct an estimated vertical profile of heads within a domain at a point. This profile is non-Dupuit, since head varies with elevation in a domain. For example, if the K3 of the lower half of level 1 domain A equals K3 of the upper half of level 2 domain B, the approximated head would vary linearly from the midpoint of domain A to the midpoint of level B. On the other hand, if the K3 of the lower half of level 1 domain A were twice the K3 of the upper half of level 2 domain B, the approximated head would vary linearly in the lower half of domain A and linearly in the upper half of domain B, but the vertical head gradient in the upper half of B would be twice that in the lower half of A. If you elect the *Non-Dupuit* option for interface position, the interface position is computed so that this approximated head balances pressure with static salt water. The *Non-Dupuit* option will have these characteristics compared to the *Dupuit* option:

- Where no vertical leakage can be computed, no non-Dupuit interface can be computed, so none can be displayed. This is true in single-level areas of models.

- Interfaces will not abruptly jump at the boundary between different levels like with the Dupuit option, but will instead be continuous since the approximated head profile is continuous.

- In areas with an upward component of vertical flow, the non-Dupuit interface will be steeper than the Dupuit interface.

- In areas with a downward component of vertical flow, the non-Dupuit interface will be shallower than the Dupuit interface.

- In areas with little or no component of vertical flow, the non-Dupuit and Dupuit interfaces will be similar.

When you check *pathlines,* the profile will show thin lines along each pathline, which is projected onto the line using a normal (right angle) projection.   If you check *pathline time markers,* a dot is displayed at each time marker (arrow in the plot) location. The pathlines and markers drawn will be those from the most recent plot made with the open model. In a transient model, the pathline is transient, and not based on the flow field just at the selected time step.

Discharge tangent to line is the component of discharge in the domain tangent to the line $[L^3/T/L = L^2/T]$, positive for flow proceeding from the first towards the second point.

Discharge normal to line is the component of discharge in the domain normal to the line $[L^3/T/L = L^2/T]$, positive for flow from left to right as you proceed from the first to the second point.

Extraction is the extraction $[L^3/T/L^2 = L/T]$ due to recharge, leakage, and storage fluxes in transient simulations. Extraction shows both modeled extraction (Equation 13 in Fitts (2010)) and extraction computed from heads (Equation 6 in Fitts (2010)). The closer these two ways of computing extraction are, the better the approximation of the governing flow equations. The extraction shown applies over the time step selected, which ends at the simulation time listed.

Vertical Leakage is the vertical leakage rate $[L^3/T/L^2 = L/T]$ in and out of the selected level(s), based on modeled head differences between levels and the vertical thicknesses and K3 values of the domains involved. The leakage shown applies over the time step selected, which ends at the simulation time listed.

Storage Flux is the flux into storage from the selected levels $[L^3/T/L^2 = L/T]$, which is positive as heads rise, negative as heads fall. The storage flux shown applies over the time step selected, which ends at the simulation time listed.

When you have made a graph, it shows the selected item(s) vs. distance along the selected line. You may export a bitmap graphic of the graph or the underlying data (see the *Exporting X-Y Graphs* topic). For a transient simulation, this shows the conditions at the end of the simulation.

The location of this line can be displayed on plots.

15.8   Graph Transient Line Conditions

This selection allows you to make graphs that show evolving conditions vs. time along a line during a transient simulation, with data plotted at each time step.

The lines are defined in the *Analysis Input/Transient Line Conditions* menu selection. To make these graphs that include initial heads, the following steps must be taken in this sequence:

1. Lines must be defined in *Analysis Input/Transient Line Conditions,*

2. Create initial head files as outlined in the *Save Locations for Initial Transient Heads* topic.

3. Solve.

4. Select items under this menu choice, which create graphs.

After you have a graph, you may export a bitmap graphic or the underlying data (see the *Exporting X-Y Graphs* topic).

This line can be displayed on plots. The line may also be edited graphically, like can be done with line boundaries: click on a line, then drag the purple squares at the vertexes to new locations.

The options under this menu are:

Head. Creates graphs of head vs. time along the line. The profile includes the initial (t=0) heads along the line. Heads shown are at the end of the time step listed.

Extraction. Creates graphs of extraction vs. time along the line.   Extraction is the extraction [$L^3/T/L^2$ = L/T] due to recharge, leakage out the top, leakage out the bottom, and storage fluxes. Both modeled extraction and extraction computed from heads are shown. In a well-defined model these two curves will be close to each other. If they are not, you probably need denser basis point spacing in your spatially-variable area sinks. The extraction shown applies over the time step that ends at the time listed in the plot legend.

Leakage at Top. Creates graphs of vertical leakage out the top of the domain vs. time along the line.   Leakage [$L^3/T/L^2$ = L/T] is computed based on head differences, vertical hydraulic conductivities (K_3) and saturated thicknesses. The leakage shown applies over the time step that ends at the time listed in the plot legend.

Leakage at Bottom. Creates graphs of vertical leakage out the bottom of the domain vs. time along the line.   Leakage [$L^3/T/L^2$ = L/T] is computed based on head differences, vertical hydraulic conductivities (K_3) and saturated thicknesses. The leakage shown applies over the time step that ends at the time listed in the plot legend.

Storage Flux.   Creates graphs of flux into storage in the domain vs. time along the line. Flux [$L^3/T/L^2$ = L/T] is computed based on the change in head during the time step listed, and applies over the time step that ends at the time listed in the plot legend.

15.9   Sum Discharge Across a Polyline

This sums the domain discharge [L^3/T] across a polyline that has been digitized and is currently residing in the clipboard after being digitized. You must first digitize a polyline in the plot view (*Digitize/Polyline*) before making this selection. The discharge is written to the run log. Discharge is positive for flow going left to right across the polyline as you proceed from the first vertex to the last. For a transient simulation, discharges at each time step are written to the run log, so you can see how discharge evolved during the simulation.

The discharge is computed by numerically summing normal components of aquifer discharge at 1000 equally-spaced points on each segment of the digitized polyline.

15.10   Sum Discharge Across External Boundaries

This sums the domain discharge [L^3/T] across all external boundaries (head specified, normal flux-specified, and head-dependent normal flux) and writes these discharges to the run log. Discharge is positive for flow going left to right across the boundary as you proceed from the first vertex towards the last. For a transient simulation, discharges at each time step are written to the run log, so you can see how these discharges evolve during the simulation.

The discharge is computed by numerically summing normal components of aquifer discharge at 1000 equally-spaced points on each segment of the boundary. The polyline vertexes that are used for this is computation are very slightly displaced into the interior of the boundary from the actual vertexes (within 10^{-4} of the segment length from the actual vertexes).

15.11   Write Leakage Over Polygon Areas to Run Log

This causes AnAqSim to estimate and write leakages [L^3/T] over polygon areas to the run log. The leakages will be computed in all model levels over x-y plane areas defined in Analysis Input / Vertical Leakage Over Polygon Area. The output text also includes the area of the polygon. This output may also be written in automated runs. For a transient run, the discharges at each time step are written to the run log.

These polygons can be displayed on plots. They may also be edited graphically, like can be done with line boundaries: click on a polygon, then drag the purple squares at the vertexes to new locations. You may also insert or delete vertexes once the polygon is selected.

15.12 Write Number of Displayed (Captured) Pathlines to Run Log

Selecting this causes the number of displayed pathlines to be written to the run log. This can be handy if you are trying to optimize the capture of pathlines originating from a contaminant source area. Use in combination with the capture constrain option, so only pathlines captured by selected wells are displayed. This information can also be written to the text output file generated by an automated AnAqSim run, which allows you to use PEST or other parameter estimation software to perform the optimization.

15.13 Graph Head Hydrographs

After solving a transient simulation, this makes head vs. time graphs at the hydrograph points input under the *Analysis Input/Hydrograph Points* menu. The points plotted are at the end of each time step in the simulation, as defined in the *Model Input/Time Steps* menu. To make these graphs, which include initial heads, the following steps must be taken in this sequence:

1. Hydrograph points must be defined in *Analysis Input/Hydrograph Points*,

2. Create initial head files as outlined in the *Save Locations for Initial Transient Heads* topic.

3. *Solve*.

4. Select this menu choice, which creates the graph.

When you select this item, a dialog pops up and asks if you want to read in observed hydrograph data from a file and plot those data as well as the modeled hydrograph. If you select Yes, you will then open a text file that has data that is either tab-delimited or comma-delimited. The file can have as many series and data values as you would like. An example of the file format for a tab-delimited file is shown below. Each row has 2*n columns for n series of data. The first row has the series label in columns 1, 3, 5,... The second and
subsequent rows have time, value pairs for each series. In this example, columns 1 and 2 have time, value pairs for well1, columns 3 and 4 have time, value pairs for well2, and columns 5 and 6 have time, value pairs for well3. Note there can be different numbers of records in different series (e.g. these three series have 11, 7, and 10 records). This tab-delimited format can be generated directly from Excel by saving as a tab-delimited text (.txt) file.

well 1		well2		well3	
1	101	1	201	1	301
2	102	2	202	2	302
3	103	3	203	3	303
4	104	4	204	4	304
5	105	5	205	5	305
6	106	6	206	6	306

7	107	7	207	7	307
8	108			8	308
9	109			9	309
10	110			10	310
11	111				

The equivalent comma-delimited (.csv) file, a format that can also be saved from Excel, would look like:

well 1,,well2,,well3,
1,101,1,201,1,301
2,102,2,202,2,302
3,103,3,203,3,303
4,104,4,204,4,304
5,105,5,205,5,305
6,106,6,206,6,306
7,107,7,207,7,307
8,108,,,8,308
9,109,,,9,309
10,110,,,10,310
11,111,,,,

You may export a bitmap graphic of the graph or the underlying data (see the *Exporting X-Y Graphs* topic).

A similar functionality is available in the plot context menu (right-click over the plot), which allows you to create hydrographs of all model levels at any location in a transient model. However, hydrographs created with the plot context menu lack initial (t=0) heads.

The locations of hydrograph points can be included in plots. These locations may also be edited graphically, like can be done with well locations: click on a hydrograph point, then drag the purple square to the new location.

15.14 Graph Drawdown Hydrographs

This item does roughly the same thing as *Analysis/Graph Head Hydrographs* except that it graphs initial head - head to give you drawdown vs. time instead of head vs. time. To allow log-log graphing of drawdown vs. time, the (time=0, drawdown=0) data point is omitted. See *Analysis/Graph Head Hydrographs* for a detailed description of how to set up this sort of analysis and how to display observed drawdown in addition to modeled drawdown.

15.15 Calibration Results

Items under this menu allow you to write or graph head calibration results. You can include all head calibration targets or only the selected ones.

15.15.1 Write All Targets to Run Log

This writes the results of calibration to the run log, including all head, head difference, and velocity calibration targets. All calibration targets are input under the *Analysis Input/Calibration Targets* menu.

15.15.2 Write Selected Targets to Run Log

This writes the results of calibration to the run log, including only selected head and velocity calibration targets. Calibration targets are input under the *Analysis Input/Calibration Targets* menu.

15.15.3 Graph All Head Targets

This creates a and x-y graph of modeled heads vs. observed heads for all head calibration targets. Head and head difference calibration targets are input under the *Analysis Input/Calibration Targets* menu.

15.15.4 Graph Selected Head Targets

This creates a and x-y graph of modeled heads vs. observed heads for the selected head calibration targets. Head and head difference calibration targets are input under the *Analysis Input/Calibration Targets* menu.

15.16 X-Y Graphs: Exporting, Changing Axes

Several of the choices under the *Analysis* menu lead to X-Y graphs. All such graphs are displayed in a separate window like the one shown below, which has two menu items for exporting. Selecting *Save Chart Image* lets you export a bitmap graphic of the graph to a file in png, jpg, or bmp formats. Selecting *Save Chart Data* exports the underlying data of the graph to a text file in tab-delimited form so that it may be later imported into documents or spreadsheets like Excel.

You can customize the axes of X-Y graphs under the *Change Axes* menu item. You can make either the X or Y axis logarithmic, but this only is possible if all values on that axis are positive.   You can also change the upper and lower limits and the label/grid interval of the X and Y axes.

When the cursor is moved over a curve on the plot, the label of that curve is displayed.
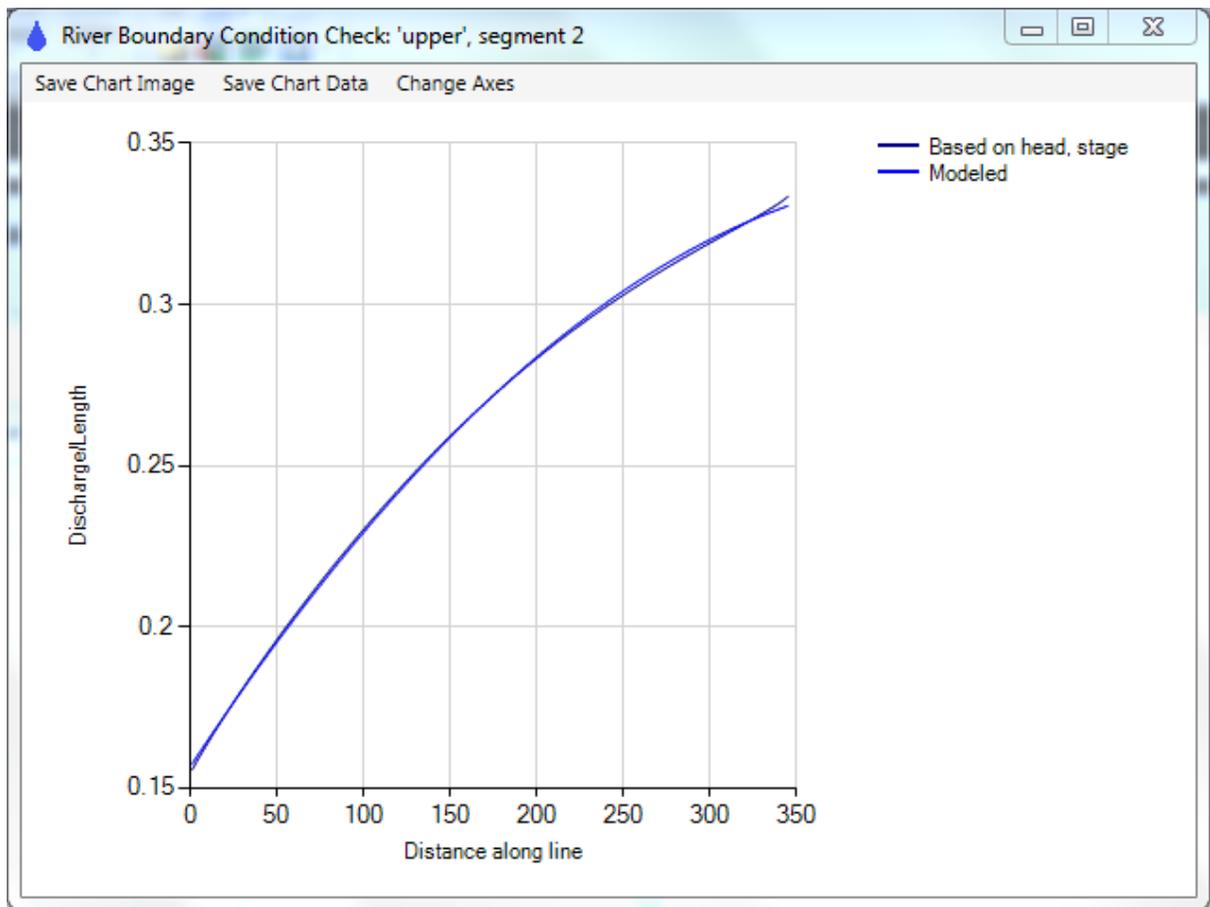
16   License Menu

These are topics about your AnAqSim User Guide license - how to activate it and check its status. This section applies to licensed AnAqSim, not unlicensed AnAqSim, which is free.

16.1   License Status this Computer

Selecting this prints the current license information for this computer to the run log. If a valid license is present, it prints the expiration date.

This pertains to licenses purchased before Feb 2020 as well as those purchased Feb 2020 or later.

16.2   License Key Activation Status

This option pertains only to licenses purchased Feb 2020 or later.

When prompted, enter your activation key code (sent via email to the purchaser at the time of the purchase) and press OK. AnAqSim will print the status of activations associated with the activation key code: the number of activations initially purchased, the number of activations used so far, and the number of activations still available. Each activation is valid on one computer for 1 year from the time of activation. There is no time limit for using activations. For example, if you purchased two activations in March 2020, you could activate on one computer in March 2020 for a valid license Mar 2020 to Mar 2021. You could later use the second activation on the same or a different computer for a valid license from Aug 2021 to Aug 2022.

16.3   Activate License on this Computer

AnAqSim licenses must be activated for key program functions to work. AnAqSim licenses are locked to a specific computer for a specific time period. There are two licensing schemes - one that started with license sales after February 2020 and the other for license sales before February 2020. AnAqSim release 2020-1 and subsequent releases are compatible with either type of license.

Activating a license for AnAqSim purchases made Feb 2020 or later:

1. Make sure your computer is connected to the internet (required).

2. Select *License/Activate License on this Computer*. When prompted, enter your activation key code (sent via email to the purchaser at the time of the purchase) and press OK. AnAqSim will activate a license on your computer, so long as there are still available activations associated with the activation key code. Each activation is valid for 1 year from the time of activation. When an activation expires, use another activation to add another year of AnAqSim use.

3. Check that the license is functioning correctly by selecting *License/License Status this Computer.* The license expiration date should be printed to the run log.

Activating a license for AnAqSim purchases made before Feb 2020 (even if the current date is beyond Feb 2020):

1. Select *Get Site Code* under the *License* menu, which generates a computer-specific code that is used to link the license to the computer. This code is displayed and copied to the system clipboard.

2. Paste the site code from the clipboard into an email to Fitts Geosolutions requesting a license.

3. Fitts Geosolutions will send a reply email with an attached license file named anaqsim.lcn. Save this file to the AnAqSim software directory on your computer. The software directory is typically C:Program Files/Fitts Geosolutions/AnAqSim for a 64-bit installation or C:Program Files (x86)/Fitts Geosolutions/AnAqSim for a 32-bit installation.

4. After copying anaqsim.lcn to the AnAqSim software directory, re-start AnAqSim and check that the license file is functioning correctly by selecting *License/License Status this Computer.* The license expiration date should be printed to the run log.

5. AnAqSim is fully functional on this computer until the license expires. The expire date is either 1, 2, or 3 years after the first license activation for this pre-Feb 2020 kind of license.

17   Help

Pressing this opens this AnAqSim User Guide Windows help file. The User Guide gives and overview of AnAqSim modeling concepts and then details about the program's user interface and all menu items.

Use the contents tab at the left to navigate to and open desired sections or use the search tab to search for topics related to specific word(s). The buttons on the topic bar at the upper right navigate to the next topic, previous topic, and one level up in the topic hierarchy.

18   About AnAqSim

This causes the AnAqSim release number and the Fitts Geosolutions, LLC web site URL to be printed to the run log.

19   Plot View Menu

This menu appears in the upper left below the main menu when you are in the plot view. It does not show in the data view because it only pertains to the plot.

19.1   Plot File Menu

These topics allow you to manipulate plot files; these are not the same as input files with the .anaq extension. These are graphics files of the plots that you create with *Make Plot*. You can save plot files to a variety of vector, bitmap, and CAD file formats. You can also open drawing files in these formats.

This menu allows printing of the plot to any printer installed on your computer.

You may also save the surface data that is the basis of a contour plot, so that you can subtract it from another solution's data to make drawdown plots.

19.1.1   Open

Allows you to open a previously-saved plot. Plots are saved in various VectorDraw (the CAD drawing component in AnAqSim) file formats, including DXF, which is a universal drawing exchange format. See the *Plot File Menu/Save* topic for a list of formats.

19.1.2   Save

Allows you to save the plot to a file in various VectorDraw (the CAD drawing component in AnAqSim) file formats. The formats include DXF, which is a common CAD file format. If you have not already saved the plot to file, a dialog pops up to ask you where to save the file, what type of graphics file it is, and the file name.   You can save plot files to a variety of formats which are listed below:

- VDML, VDCL, VDF, VDI - Vector Draw CAD file formats

- DXF - CAD drawing exchange formats of various vintages (ASCII file).

- WMF, EMF, SVG - Vector graphics file formats

- BMP, GIF, JPG, TIF, ICO, PNG, - Bitmap file formats

- PDF - Adobe portable document format

19.1.3   Save As

This is the same as *Save*, but automatically shows the dialog allowing you to change the filename, format, or destination.

19.1.4 Clear Plot

This clears the plot to a blank view.

19.1.5 Print

This brings up a dialog that allows you to print the plot to any printer available to your computer. The print dialog has many standard options including paper size, paper orientation, and margins. There is a plot preview at the left of the dialog, which shows a detailed representation of the plot if you select *Full Preview* under *Preview Options*. You may scale the plot to fit the paper or you may select a specific scale (e.g. 1 cm on printer = 50 m in model units). The *Print Area* part of the dialog allows you to print the entire extent of the plot, or select a window to print a smaller portion of the plot. To do the latter, click on *Window* and then the *Pick* button. When the plot preview is adjusted to your satisfaction, click the *Print* button to send it to the printer.

19.1.6 Save Contour Grid Data to Binary File

This allows you to save the data that was contoured to a file in binary format. This is useful if you want to contour the difference between two modeled surfaces using the *Subtract* feature under *Plot Input/Contour Settings*. You might use this to compare the drawdown of simulation A compared to simulation B, for example. When you do this, make sure that the *Plot Input/What to Plot/Window* and *Plot Input/Contour Settings/Points_Evaluate* are the same in the saved contour grid data as in the new plot you create.

19.1.7 Save Contour Grid Data to X,Y,Z ASCII file

This allows you to save the data that was contoured to an ASCII file, which then could be imported into other visualization software programs where you could examine the contoured surface. You can export either to .csv (comma-delimited) or .txt (tab-delimited) formats. The data format of that is written is as shown in the following Excel file, which was opened as a .csv file.

	A	B	C
1	x	y	h
2	-1276.29	-577.238	121.4786
3	-1276.29	-450.526	121.9639
4	-1276.29	-323.814	122.4155
5	-1276.29	-197.102	122.8181
6	-1276.29	-70.3896	123.2226
7	-1276.29	56.32252	123.6281
8	-1157.01	-577.238	121.181

The first row is a header row, and the subsequent rows contain

• X (column A)

- Y (column B)

- contoured parameter (column C)

Where heads drop below the base of unconfined domains, the value in the grid is set at the base elevation.

19.1.8 Save Contour Grid Data to ASCII Surfer Grid File

This allows you to save the data that was contoured to an ASCII file in a format that can be directly imported into the Surfer contouring software (Golden Software). You can then import this data into Surfer for analysis and plotting there. The file is saved with the .grd extension recognized by Surfer. Below is an example of such a file. which has blank characters as separators. The first row, "DSAA" identifies the type of file being imported to Surfer. The second line contains the numbers of data values in the x and y directions of the grid. The third line contains the minimum and maximum values of x in the grid and the fourth line contains the minimum and maximum values of y in the grid. The fifth line contains the smallest and largest data values in the grid. The sixth and subsequent lines contain the data values starting from xmin, ymin and working up by x rows to xmax, ymax.

DSAA

72 70

-427.5754841 315.3007776

-326.9260294 395.4048918

16.85028419 22.93994978

22.93994978

22.8778982

22.81579964

22.75365779

...

Where heads drop below the base of unconfined domains, the value in the grid is set at the base elevation.

19.1.9 Save Pathline Start Point Data to ASCII File

This allows you to save the starting point data for the currently-plotted pathlines to an ASCII file, which then could be imported into other spreadsheet or visualization software programs. You can export either to .csv (comma-delimited) or .txt (tab-delimited)

formats. The data format of that is written is as shown in the following Excel file, which was opened as a .csv file.

	A	B	C	D	E	F	G	H	I	J	K	L
1	C:\Users\CFITTS\Documents\AnAqSim\Manual\tutor2_test_capture_pathlines.anaq pathline start points. Elapsed time = 0 and simulation time = 0											
2	x	y	z	pathline label	capturing element label							
3	-448.526	-458.161	117.5562	Line pathline: upstream	test Drain							
4	-447.157	-438.26	117.5932	Line pathline: upstream	test Drain							
5	-445.787	-418.36	117.6322	Line pathline: upstream	test Drain							
6	-444.418	-398.459	117.6741	Line pathline: upstream	test Drain							
7	-443.048	-378.559	117.7195	Line pathline: upstream	test Drain							
8	-441.679	-358.659	117.7684	Line pathline: upstream	test Drain							
9	-440.309	-338.758	117.8207	Line pathline: upstream	test Drain							
10	-436.2	-279.057	117.9928	Line pathline: upstream	upper river							
11	-434.831	-259.157	118.0533	Line pathline: upstream	upper river							
12	-433.461	-239.256	118.1143	Line pathline: upstream	upper river							

The first two rows are header rows, and subsequent rows have these 5 columns of data:

- x (column A)

- y (column B)

- z = elevation (column C)

- *pathline label* from the single, line, area, circle, or well pathline input (column D)

- *capturing element label*. If capture constraint is checked under Pathline Settings, this will display the well or line boundary label that captured the pathline that started at this point. (column E)

19.1.10 Save Entire Pathline Data to ASCII File

This allows you to save the data for currently-plotted pathlines to an ASCII file, which then could be imported into other spreadsheet or visualization software programs. You can export either to .csv (comma-delimited) or .txt (tab-delimited) formats. The data format of that is written is as shown in the following Excel file, which was opened as a .csv file.

	A	B	C	D	E
1	Start pathline 1				
2	pathline label	capturing element label			
3	Line pathline: u| test Drain				
4	x	y	z	elapsed time	simulation time
5	-448.5264819	-458.1607272	117.5562	0	0
6	-442.5237658	-456.2611718	117.2741	44.84959253	0
7	-436.5276393	-454.340917	116.996	89.28742748	0
8	-430.5382489	-452.3997542	116.7218	133.3169272	0
9	-424.5557496	-450.4374562	116.4514	176.9413401	0
10	-418.580306	-448.4537766	116.1849	220.163734	0
11	-412.6120925	-446.448449	115.922	262.9869896	0
12	-406.6512941	-444.4211864	115.6627	305.4137916	0
13	End pathline 1				
14	Start pathline 2				
15	pathline label	capturing element label			
16	Line pathline: u| test Drain				
17	x	y	z	elapsed time	simulation time
18	-447.1569041	-438.260314	117.5932	0	0
19	-441.1519204	-436.3679396	117.3166	43.62403157	0
20	-435.1532227	-434.4557321	117.044	86.81412549	0
21	-429.1609402	-432.5235154	116.7752	129.5725343	0
22	-423.1752092	-430.5710976	116.5104	171.9012989	0
23	End pathline 2				
24	Start pathline 3				
25	pathline label	capturing element label			
26	Line pathline: u| test Drain				
27	x	y	z	elapsed time	simulation time
28	-445.7873264	-418.3599008	117.6322	0	0
29	-439.7751891	-416.4903777	117.3598	42.6480702	0
30	-433.768786	-414.6025133	117.0913	84.848522	0
31	-427 7682152	-412 6961917	116 8267	126 6028462	0

The data for each pathline starts with three header lines. The first header line defines says "Start..." and includes the pathline number, which goes from 1...n with n pathlines in the plot. The second and third lines of the header define the pathline's label under pathline input, and the well or line boundary that captured the pathline, if capture constraint is checked under *Pathline Settings*. Subsequent rows have these 5 columns of data:

- x (column A)

- y (column B)

- z = elevation (column C)

- *elapsed time* at this vertex of the pathline. Elapsed time always starts at zero at the pathline start point, and increases whether the pathline is traced upstream or downstream.

- *simulation time* at this vertex of the pathline. This is zero for a steady simulation, but in a transient simulation this is the simulation time.

The data for each pathline ends with the line *End pathline n*, where n is the pathline number.

19.2   View Manager

This brings up a dialog that allows you to alter the color, line weight, and visibility of various layers of the plot. There are separate plot layers for contours, vectors, etc. To toggle the visibility of a layer of the plot, click the icon in the *On* column. To alter the color or line weight, click on those columns. To omit a layer when printing or saving a plot as a pdf file, toggle the *Plot* column. When done making changes, press the *OK* button. Changes to the On status, line color, line weight persist as long as you keep the AnAqSim application running; these settings return to defaults when you exit AnAqSim.

For pathlines, there are layers for each line of input in the pathline input tables (point, line, area...). These can be turned on/off separately, to show certain groups of pathlines. When a group of pathlines is turned off, the pathline trace and the associated arrow time markers (if checked under *Plot Input/Pathline Settings*) are turned off. Also, in the View Manager you can turn off all time markers but leave the pathline traces. The pathline starting points are all in a separate drawing layer, and these may be turned on/off in the View Manager.

19.3   Digitize

These items let you digitize coordinates from the plot, which are stored in the clipboard and from there may be pasted into input data tables or forms with *Edit/Paste* or *control-V*. It is very helpful to employ a basemap for digitizing. Basemaps are discussed in the *Plot Input/What to Plot* topic. Digitizing operations are demonstrated in detail in the tutorial videos at the website.

19.3.1 Point

This allows you to digitize a single point. The coordinates of the point are stored as text in the form:

X, Y

in the clipboard.

This same functionality, but without the instructional pop-up windows, is available in the plot context menu (right-click over the plot).

19.3.2 Polyline, Measure Distance

This allows you to digitize a series of points, the vertexes of a polyline. After the last point, press Esc or right-click. At this point a message pops up telling you how many vertexes the polyline has and its total length. Also, if the polygon is closed (first and last points the same), the message reports the area of the polygon. This is a handy way to measure distance or area on the plot. The coordinates of the vertexes are stored in the clipboard as a series of text lines of the following form:

X1, Y1,
X2, Y2
X3, Y3
...

These coordinates may be pasted into the Coordinates column for any line boundary.

This same functionality, but without the instructional pop-up windows, is available in the plot context menu (right-click over the plot).

19.3.3 Circle Polygon

This allows you to digitize a series of points that are the vertexes of a polygon which lie equally-spaced on a circle. First you select the number of points on the circle and then you digitize the center point and then digitize a 2nd point on the circle. As you perform the digitizing operation, the circle you are creating is shown in yellow. This is useful if you want to approximate a circular line boundary (e.g. inter-domain boundary).

The coordinates of the vertexes are stored in the clipboard as a series of text lines, each line of the form:

X1, Y1,
X2, Y2
X3, Y3
...

19.3.4 Ellipse Polygon

This allows you to digitize a series of points that are the vertexes of a polygon which lie equally-spaced on an ellipse. First you select the number of points on the ellipse and then you digitize the center of the ellipse, followed by a point that is on the ellipse at one of the two primary axes, and then a third point that defines the final shape of the ellipse. As you perform the digitizing operation, the ellipse you are creating is shown in yellow. This is useful if you want to approximate an ellipse line boundary (e.g. inter-domain boundary).

The coordinates of the vertexes are stored in the clipboard as a series of text lines, each line of the form:

X1, Y1,
X2, Y2
X3, Y3
...

19.3.5 Surface Data Points

This allows you to digitize a list of surface data points in an "x,y,value" form, which can be then input under *SVAS Top/Bottom Condition Surfaces*. After selecting this, a dialog pops up explaining how to digitize multiple points of input. You may continue to input a large number of points in one operation, and the data will be written to the clipboard so it can be pasted into the *Data* column under *SVAS Top/Bottom Condition Surfaces*.

19.3.6 Clear Digitizing Marks

When you digitize, the features that you've digitized show up in the plot. Selecting this item clears those marks from the plot.

This same functionality, but without the instructional pop-up windows, is available in the plot context menu (right-click over the plot).

19.4   Snap Settings

This menu lets you control whether and how the cursor snaps to other objects in the drawing or to a regularly-spaced grid of points. Snapping is generally useful for precision digitizing. The first four items are toggled on or off by selecting them. When a check mark is present next to one of these four items, snapping in that mode is in effect. The four snap modes are listed below:

• **Snap to Elements** causes the cursor to snap to end points and centers of features shown in the "Elements" layer and the "SVAS Polygons" layer of the drawing. This is useful for making sure that line elements and SVAS polygons meet exactly at their end points. There is a dialog for changing what the cursor is snapping to, which can be reached by pressing Ctrl-right mouse button. Generally the default settings in this dialog are best, but in case you want to snap to the midpoint of a line element, you could set it to do that.

• **Snap to Base Map** causes the cursor to snap to drawing entities in the base map. This is mode should generally be off when you are trying to digitize line elements that need to connect to each other.

• **Snap to Digitized Data** causes the cursor to snap to other recently-digitized features that are displayed in yellow. This allows you to digitize several line boundaries and have them connect exactly.

- **Snap to Grid** causes the cursor to snap to points in a regularly-spaced grid with a grid spacing that you set. For example, if you set the snap grid spacing to 10, all digitized points will be multiples of 10. This feature helps you precisely draw a line 100 units long, or a circle with precisely a 100 unit radius, for example. It also can make the coordinates you digitize easy-to-read whole numbers.   Note that if the grid spacing is smaller than the screen pixel size, this mode is ignored and digitizing behaves as though this mode were off. If this is the case, either zoom in or increase the snap grid spacing to make this mode active.

The Snap Grid Spacing sets the spacing between points in the grid that is effecting when Snap to Grid mode is checked.

19.5   Edit Line Boundary

This menu gives you the ability to move, insert, or delete vertexes from a selected line boundary or spatially-variable area sink (SVAS) polygon. To insert a vertex, select the line boundary or polygon (its existing vertexes will display purple squares), then select *Edit Line Boundary/Insert Vertex*, click OK when the dialog pops up, then click where you want the new vertex. When done, press ESC to de-select the line boundary.   To delete a vertex, select the line boundary, then select *Edit Line Boundary/Delete Vertex*, click OK when the dialog pops up, then click the vertex you want to delete. When done, press ESC to de-select the line boundary. When you make these graphic edits, it changes the coordinates listed for that line boundary in the Model Input database.

The same functionality for inserting and deleting vertexes, but without the instructional pop-up windows, is available in the plot context menu (right-click over the plot).

19.6   Annotate

These menu items allow you to add polylines, rectangles, and text to the plot.

19.6.1 Polyline

After selecting this, left-click successively on the desired vertex locations for the polyline. After the last point, press Esc or right-click.

19.6.2 Rectangle

After selecting this, click at one corner location and drag to the opposite corner and click again to draw a rectangle.

19.6.3 Circle

After selecting this, draw a circle with two left-clicks of the mouse. The first click defines the center point and the second click defines a point on the circle and the radius.

19.6.4 Ellipse

After selecting this, draw an ellipse with three left-clicks of the mouse. The first click defines the center point, the second click defines the direction and length of one of the two axes, and the third click defines the length of the other axis normal to the axis defined by the second click.

19.6.5 Text

After selecting this, click where you want the base of the text line to start, then drag in the direction you want the text to run and click a second time. This allows text to be plotted in any direction. Then type the text you want and when you are done typing your text, press *Enter*.

19.6.6 Clear All Annotations

Selecting this will erase all annotations from the plot.

19.7 Crop Plot

This item allows you to crop the plot and just show portions of the plot that are inside of a polygon. You select a polygon with any number of vertexes using the left mouse button, and as you draw the polygon is shown as a shaded area. When finished, click the right mouse button. All drawing entities that are outside or partially outside the polygon will be removed. For example, a line boundary segment that has one end inside and one end outside the polygon will not be drawn.

19.8 Undo

Selecting this undoes the most recent change such as deleting an annotation. This can be repeated to undo several recent changes.

19.9 Redo

This reverses the most recent *Undo* action.   This can be repeated to redo a sequence of recent *Undos*.

19.10 Text Style

This brings up a dialog that allows you to change the characteristics of text displayed on the plot. Changing any of the text characteristics makes this change for all text in the plot including annotations and calibration residuals. The only exception is that the size of the contour label text is fixed and determined automatically as a fraction of the plot window dimensions.

19.11 Plot Info

Selecting this causes information about the plot to be printed in a small window. It shows the coordinates of the plot window. If there are contours, it shows the surface contoured, the minimum and maximum data values, the minimum and maximum contour levels, and the contour increment. If there are vectors, it shows they kind of vector shown and the minimum and maximum vector magnitudes. For transient runs, it shows the time period, time step, and time that applies to plot features like contours and vectors.

19.12 Zoom All

This causes the plot to zoom out so that the view shows all of the model.

19.13 Zoom Window

After selecting *Zoom Window*, select the new display window by clicking at one corner, drag to the opposite corner, and click again. The view will zoom to the newly-selected window.

19.14 Zoom Out

This makes the plot zoom out incrementally showing more of the model. Zooming in is accomplished with *Zoom Window*.

19.15 Zoom Previous

This zooms to the previous view.

20   Checks of AnAqSim

AnAqSim has been checked in many ways against exact analytic solutions and against numerical models like MODFLOW. These checks and the input files associated with them are available at http://www.fittsgeosolutions.com.

21   References

Fitts, C. R. (2002), *Groundwater Science*, Academic Press, San Diego.

Fitts, C. R. (2004), Discrete analytic domains: a new AEM formulation for modeling anisotropy and heterogeneity, Proceedings of EPA/NGWA Fractured Rock Conference, Portland Maine, September 2004.

Fitts, C. R. (2006), Exact solution for two-dimensional flow to a well in an anisotropic domain, *Ground Water, 44(1)*, 99-101.

Fitts, C. R. (2010), Modeling Aquifer Systems with Analytic Elements and Subdomains, *Water Resources Research,* 46, W07521, doi:10.1029/2009WR008331.

Fitts, C. R. (2012), *Groundwater Science 2nd edition*, Academic Press, San Diego.

Fitts, C.R., J. Godwin, K. Feiner, C. McLane, and S. Mullendore (2015), Analytic Element Modeling of Steady Interface Flow in Multilayer Aquifers Using AnAqSim, *Groundwater*, 53 (3), 432–439, doi: 10.1111/gwat.12225

Fitts, C. R. (2018) Modeling Dewatered Domains in Multilayer Analytic Element Models, Groundwater, 56 (4), Early view article DOI: 10.1111/gwat.12645.

Hardy, R.L. (1971.)Multiquadric equations of topography and other irregular surfaces. Journal of Geophysical Research 76, 1905–1915.

Haitjema, H. (1995), *Analytic Element Modeling of Groundwater Flow*, Academic Press, San Diego.

Haitjema, H. M., and O. D. L. Strack (1985), An initial study of thermal energy storage in unconfined aquifers, Tech. Rep. PNL-5818 UC-94e, Pac. Northwest Lab., Battelle Mem. Inst., Richland, Wash.

Hansen, D. (2002), Analytic modeling of leakage in confined aquifer systems, MS thesis, Univ. of Minnesota, Minneapolis.

Jankovic, I. and Barnes, R. (1999), High-order line elements in modeling two-dimensional groundwater flow, *Journal of Hydrology, 226*, 211-223.

Strack, O. (1984), Three-Dimensional Streamlines in Dupuit-Forcheimer Models, *Water Resources Research*, 20 (7), 812-822.

Strack, O. (1989), *Groundwater Mechanics*, Prentice Hall, Engle- wood Cliffs, NJ.

Strack, O. (2003), Theory and applications of the analytic element method. *Reviews of Geophysics, 41(2)*, 1005.

Strack, O. and Jankovic, I. (1999), A multi-quadric area-sink for analytic element modeling of groundwater flow, *Journal of Hydrology, 226*, 188-196.