Modelling density-dependent flow with the use of MODFLOW 6

The effect of spatial and temporal discretization

Author:

Eva Schoonderwoerd

Student number: 4080858

Master: Earth Surface and Water

Credits: 22.5 EC

Under supervision of:

Gijs Janssen

Gualbert Oude Essink

Utrecht University

Deltares

Enabling Delta Life

Abstract

In this study, a variable density groundwater flow and solute transport model that was originally created in SEAWAT is translated to a MODFLOW 6. The spatial and temporal discretization of this MODFLOW 6 model is adjusted in 60 different cases. Increasing the timestep length of a MODFLOW 6 case will reduce the model runtime but may lead to slightly less accurate results. A potential benefit of MODFLOW 6 is the possibility of implementing unstructured grids. Therefore, the cases in this study use regular unstructured grids, i.e. unstructured grids that still consist of regular hexahedra.

MODFLOW 6 is generally faster than SEAWAT, even if the same grid and model settings are used. It is possible to implement coarser unstructured grids to further improve the runtime of a MODFLOW 6 model, without influencing the model results. This can be achieved by implementing coarser grids at lower model layers or by reducing the number of unnecessary refined cells that used to be present in telescopically refined grids of earlier versions of MODFLOW. Refining the grid close to an extraction well leads to a sharp increase in computational time. To prevent this increase, the grid can be coarsened at other locations. However, there is a limit on the number of cells that can be coarsened to keep the runtime low without reducing the accuracy of the results.

Table of Contents

List of figures

List of tables

1 Introduction

At the start of the Holocene, the sea was located further inland in the Netherlands compared to the situation today. In this period, salt water intruded the subsurface, and is still present in the deep subsurface. Due to dewatering (*inpolderen* in Dutch) of the land, this salt water is flowing towards the surface. This results in complications for agriculture, nature and the extraction of drinking water. Moreover, this process is accelerated due to land subsidence, sea level rise and climate change (which results in more arid summers) (Van Baaren et al., 2016).

Especially in Zeeland, a large part of the groundwater is brackish or salt, which is partly due to the fact that a large part of Zeeland is dewatered since the Middle Ages. In most areas in Zeeland, fresh groundwater is only available very close up to the surface, for example in so-called shallow fresh water 'lenses' in sandy creek ridges. Moreover, clay layers often occur in the top part of the subsurface, that limits fresh groundwater to flow deeper into the subsurface (Van Baaren et al., 2016). In dry summers, like in 2018 and 2019, fresh (ground)water is needed for agriculture while the resources are limited. Under future climate change and scenarios with sea-level rise, the situation gets worse.

Due to this decrease in available fresh (ground)water in Zeeland, it is becoming very important to understand how the fresh and saline water distribution may be in the future. Since fresh and salt water have different densities, the groundwater models that are used for this problem implement density dependent flow. To be able to predict how the fresh water availability will change in the future, regional and local three-dimensional numerical models were developed for the province of Zeeland to describe the variable-density groundwater flow under the influence of several climate scenarios. Right now, the existing regional 3D model (made with the code MOCDENS3D (Van Baaren et al., 2016)) and a conceptual local 3D model around parcels of a farmer which extracts fresh groundwater (made with the code SEAWAT (Oude Essink and Pauw, 2018)) are models which require large expensive simulation times due to the nature of variable-density groundwater flow and coupled salt transport.

1.1 Density dependent groundwater flow

The presence of salt water in the groundwater system can have a big impact on the groundwater flow. The density of saline water (1022 kg/m^3) is significantly higher compared to fresh water (1000 kg/m^3). As a result, the salt water in a groundwater system will flow down, whereas the fresh water can float on top of the salt water. This means that the density difference will have an impact on the velocity field in the system. Moreover, the transport (by advection and/or dispersion) of salt should be simulated. The mixing of fresh, brackish or salt water should also be incorporated. This transport of salt alters the concentration of the water, which influences the fluid density and subsequently influences the velocity field.

Since the solute concentration influences the water density, the solute transport equation (advection-dispersion equation) is coupled to the groundwater flow equation. This is different from 'normal' transport modelling, where the flow and transport part of the simulation can be solved independently.

The relation between solute concentration and density is not straightforward and many different equations of state are reported in literature to describe density as a function of solute concentration, pressure and temperature (see e.g. Voss, 1984). Baxter and Wallace (1916) developed an empirical formula that shows a linear relation between water density and solute concentration. This relationship is also implemented in SEAWAT (Guo and Langevin, 2002):

$$\rho = \rho_f + EC \qquad 1\text{-}1$$

Where E = slope of the linear relationship (-). In this study: 1.3419 (-). C = solute concentration (M L^{-3}).

Usually, the hydraulic head is used in groundwater flow calculations. This head is the sum of pressure head and the elevation head:

$$h = \frac{P}{\rho g} + z \qquad 1\text{-}2$$

Where h = hydraulic head (L), P = pressure (M L^{-1} T^{-2}), ρ = density (M L^{-3}), g = gravity (L T^{-2}), z = elevation compared to a reference level (L).

In unconfined aquifers with fresh water, the hydraulic head is used to describe the elevation of the water table. Equation 1-2 shows that the hydraulic head is dependent on the water density, which means that a measured hydraulic head in salt water will be lower compared to a measurement of hydraulic head in fresh water. In that case, the hydraulic head is not a direct measurement of the water table elevation. To overcome this apparent discrepancy, the hydraulic head in salt water can be translated to an equivalent fresh water head:

$$h_f = \frac{P}{\rho_f g} + z = \frac{\rho}{\rho_f} h + z \qquad 1\text{-}3$$

Where h_f = fresh water head (L), ρ_f = fresh water density (M L^{-3}), h = measured pressure head (L). It should be noted that the use of fresh water heads may result in some unexpected patterns, for example a difference in fresh water head does not explicitly mean that the groundwater will flow towards the lower head.

1.2 Fresh, brackish or salt groundwater

Since this study uses a variable density groundwater model, the definition of fresh, brackish and saline water should be determined. Saline (sea) water contains several dissolved solids, but the main component is chloride (Cl^-). Therefore, chloride concentration is usually used as measure of groundwater salt content. A well-known classification can be found in Stuyfzand (1993), which uses the following specifications: 30 - 150 mg Cl^-/l for fresh water, 150 - 300 mg Cl^-/l is brackish and 300 – 1000 mg Cl^-/l is salt water. This classification is used to define the limits for drinking water and is implemented for groundwater that is generally fresh. However, the groundwater in the province of Zeeland has a much higher chlorine concentration and water with a concentration below 150 mg Cl^-/l only exists in deep fresh water lenses (Van Baaren et al., 2016). Therefore, the classification for Zeeland is different and the term agricultural fresh water (*landbouwkundig zoet* in Dutch) is used. In this classification, fresh water has a concentration below 1000 mg Cl^-/l and salt water above 3000 mg Cl^-/l (Van Baaren et al., 2016). The policy of the province and water authority (*waterschap* in Dutch) only uses fresh and brackish water, where fresh water is at maximum 1500 mg Cl^-/l (Provincie Zeeland, 2002 and Scheldestromen, 2013). This limit is also applied in this study, in accordance with the study of Oude Essink and Pauw (2018).

1.3 Groundwater flow and transport modelling

MODFLOW is a widely used computational model that is used to simulate groundwater flow. It is developed by the U.S. Geological Survey (USGS). It can simulate groundwater flow under the influence of several different hydrological stresses, for example extraction or infiltration wells, evapotranspiration and rivers. The original focus of MODFLOW was to only simulate groundwater flow (Langevin et al., 2017).

MT3DMS (Zheng and Wang, 1999) is used for the simulation of solute transport by advection, dispersion, diffusion and chemical reactions. It is developed to be used together with MODFLOW, but it can also be linked to other flow models, provided that they are also a block-centered finite-difference model. One of the options to solve the advection part of the transport equation in MT3DMS is the third-order TVD scheme, which is based on the ULTIMATE algorithm. This algorithm interpolates the concentration of a cell based on the concentration at the nodes of two upstream cells, and one node directly downstream, which

makes it a third-order scheme. The TVD scheme is mass conservative and implemented as explicit (Zheng and Wang, 1999).

SEAWAT (Guo and Langevin, 2002) is another groundwater model that is developed specifically to simulate three-dimensional variable density groundwater flow. The code comprises a combination of MODFLOW and MT3DMS. An important aspect is that fluid mass (adjusted for fluid density), instead of fluid volume, is used in the governing equations. The flow equation is used to define the velocity field due to advection, which is subsequently used to solve the solute transport equation. Note that this is different from the method used in MODFLOW/MT3DMS; since the fluid density influences the velocity field, the flow and transport model are solved simultaneously, instead of in separate model runs.

When using numerical models to solve the solute transport equation, two issues may occur: numerical dispersion or oscillations. Numerical dispersion looks like an extra cause of dispersion, i.e. the concentrations are more smeared. Oscillations occur due to over- and undershooting of the concentration. These two issues are linked: measures that can be taken to limit the numerical dispersion can cause over- and undershooting of the numerical dispersion (Oude Essink, 2001). There are three criteria that should be satisfied to ensure stability of the solute transport equation. Firstly, the Neumann criterion (for one dimension) (Zheng and Wang, 1999; Oude Essink, 2000):

$$\Delta t \leq \frac{0.5}{D/\Delta x^2} \qquad 1\text{-}4$$

Where Δt = timestep length (T), D = dispersion coefficient ($L^2\ T^{-1}$), Δx = cell dimension (L). Dispersion coefficient is calculated as $D = \alpha V$, where α = dispersivity (L) and V = velocity ($L\ T^{-1}$). The Neumann criterion means that the minimum timestep of the simulation is limited by the cell where $D/\Delta x^2$ is the largest. This is caused when D is large or Δx^2 is small.

Secondly, the mixing criterion is relevant (Zheng and Wang, 1999; Oude Essink, 2000):

$$\Delta t \leq \frac{n_e b}{q} \qquad 1\text{-}5$$

Where n_e = effective porosity (-), b = cell thickness (L), q = volume flux per unit area of the cell ($L\ T^{-1}$). This criterion is based on the assumption that the change in concentration in a cell should not exceed the difference between the current concentration in the cell and concentration in the source (Oude Essink, 2000).

Lastly, the Courant criterion should be satisfied (Zheng and Wang, 1999; Oude Essink, 2000):

$$\Delta t \leq \frac{\Delta x}{V} * Co \qquad 1\text{-}6$$

Where Co = Courant number (-), which should be equal or below 1. Generally, Co = 1. This criterion specifies that distance that a particle moves by advective transport should not exceed the cell dimensions.

These criteria are all implemented in the explicit solution of the solute transport equation in MT3DMS and SEAWAT, which is also the case when the TVD (ULTIMATE) scheme is implemented (Zheng and Wang, 1999; Guo and Langevin, 2002). If the user-specified timestep (that is used by MODFLOW) exceeds these criteria, the timestep is further subdivided into so-called transport steps. However, MT3DMS and SEAWAT also contain an implicit solver (Generalized Conjugate Gradient, GCG) that uses iterations to solve (parts of) the transport equation. In that case, the aforementioned stability criteria are not used.

1.4 MODFLOW 6

Recently, the latest version of MODFLOW, MODFLOW 6, was released. The aim of this new version is to be able to use all the capabilities that were developed in all the separate MODFLOW variants in a clear, easily understandable way (Langevin et al., 2017). This means that some rigorous changes were made compared to previous versions, some of these changes will be discussed in this section.

A big difference is that MODFLOW 6 allows the use of unstructured grids. This means that a grid cell can be connected to more than one cell at every side and that cells do not have to have a rectangular (or hexahedral) shape (Langevin et al., 2017). This way, a fine grid can be applied at points of interest, without decreasing the cell size at other, unimportant locations. This concept is based on an earlier MODFLOW version: MODFLOW-USG (Panday et al., 2013). Although the possibility exists to implement irregular shaped cells, this could influence the accuracy of the model results. The best option to decrease the possibility of accuracy errors, is to implement quadtree or octree refinement. With this way of refining, a two-dimensional grid cell is divided into four equally sized smaller cells (quadtree) or a three-dimensional cell is divided into eight cells (octree). These grids can also be smoothed, to make sure that a cell is connected to a maximum of two cells in every direction (Panday et al., 2013).

There are two requirements of a grid to make sure that the model results are accurate. First, the line that connects two cell centers should make a right angle with the interface between the two cells. Second, this intersection should be in the middle of the interface between the cells (Narasimhan and Witherspoon, 1976; Panday et al., 2013). Since these rules are not met for an unstructured grid, the Ghost Node Correction (GNC) Package is developed for MODFLOW-USG and used in MODFLOW 6. A ghost node can be implemented to reduce the error that may arise when the grid requirements are not met. It is a fictitious node of a cell that indicates the location where the head value should be calculated (by interpolation of head values at all real nodes of the cells) (Panday et al., 2013; Langevin et al., 2017). However, it is noted that model results could still be accurate even though the grid requirements are not satisfied, so the GNC package is not always necessary (Panday et al., 2013).

MODFLOW 6 does not use the Basic Package anymore. This package contained the IBOUND variable and the initial heads. The IBOUND variable was used to specify whether a cell has a variable head, constant head or was an inactive (no-flow) cell (Harbaugh, 2005). MODFLOW 6 contains the Specified Head (CHD) package in which the heads for the constant head cells can be specified. The main advantage of the package is that the input is read every stress period, meaning that a constant-head cell could change to an active cell (or vice versa) during the simulation (Langevin et al., 2017).

The internal flow is simulated with the Node Property Flow Package, which replaces the Block-Centered Flow (BCF) and Layer Property Flow (LPF) Packages from older MODFLOW versions. This new flow package uses the horizontal and vertical hydraulic conductivity and the ICELLTYPE variable as input. This last variable is used to define whether a cell is confined or convertible. A confined cell has a constant transmissivity, whereas a convertible cell has a varying transmissivity based on the saturated cell thickness (which could change due to drying or wetting). The storage calculations, that were originally done by the LPF and BCF packages, are now done by the separate Storage (STO) Package (Langevin et al., 2017).

The last important aspect of MODFLOW 6 is that it can have multiple 'models' within one simulation. Every model solves a separate (geo)hydrological process. The official MODFLOW 6 release only contains the groundwater flow model ('GWF'), but in this study a solute transport model is also used ('GWT'). Together, these models have the option to simulate variable density groundwater flow. The transport model is still under development by the USGS, so there may be still some inaccuracies or errors in the model results. Moreover, the documentation on the transport model is scarce, so there is not yet a clear explanation of every process of this model. For example, it is unclear whether timesteps in a simulation are subdivided into transport steps to solve the transport equation, as is the case for MT3DMS and SEAWAT.

1.5 Research questions

The main goal of this study is to determine the suitability of unstructured grids for the modelling of variable density groundwater flow and coupled solute transport with the use of MODFLOW 6. Focus will be on the effect of both spatial and temporal discretization on the model results and computational runtime. For this study, a conceptual model that was developed by Oude Essink and Pauw (2018) with SEAWAT will be converted to MODFLOW 6. The model simulates a situation in the province of Zeeland in the Netherlands, where the groundwater is generally saline, but where fresh groundwater lenses form beneath creek ridges. The model mimics a creek ridge where several ditches and an extraction well (for agricultural purposes) are present and a fresh groundwater lens forms below the creek ridge.

The following research questions will be answered:

1. Can a fully regular grid without local refinement produce accurate results while maintaining a reasonable runtime?
2. What is the effect of implementing a regular unstructured grid, i.e. an unstructured grid with regular hexahedra, on the results and runtime of a variable density groundwater model compared to the same model that has a structured, but refined grid?
3. Can unstructured grids be used to implement fine grids at points of interest, without drastically increasing the runtime of a model?
4. How does the temporal discretization of a variable density groundwater flow and transport model in MODFLOW 6 relates to model runtime and results?

2 Methods

2.1 Original model

A SEAWAT model, created by Oude Essink and Pauw (2018), is implemented in MODFLOW 6. This model is a conceptual model, based on the situation in the province of Zeeland in the Netherlands, where wide creek ridges are present with drains on top (Oude Essink & Pauw, 2018). A schematic representation of the model is shown in Figure 2-1. The ground surface is at 2 m above NAP. The constant head boundaries are saline, i.e. they have a constant concentration of 16.394 kg Cl^-/m^3. Both constant head boundaries are used to simulate a ditch: the ditch at y = 650 m is a ditch that is located on top of the creek ridge, the other is a ditch in the lower lying area (*poelgrond* in Dutch) next to the creek ridge. The hydraulic conductivity up to 5 meters below NAP below the second ditch is low to represent the geology of the low-lying areas (*poelgronden*). The horizontal well is present at 3 m below NAP, where water is extracted from the domain for 13 weeks every year. Because the well is present over a length of 40 m, the extraction is simulated by implementing multiple model wells in all cells in the range 0-40 m. All model wells extract the same volume, which results in a total extraction of 30 m^3/day. The drain (yellow line in Figure 2-1) extracts water from the domain as long as the water table is above the drain elevation (0.3 m above NAP). The extraction rate of the drain is defined as follows (Langevin et al., 2017):

$$Q = C\,(h - H) \qquad 2\text{-}1$$

Where Q = flow from the domain into the drain ($L^3\,T^{-1}$), C = drain conductance ($L^2\,T^{-1}$), h = hydraulic head in the cell containing the drain (L), H = drain elevation (L). This equation shows that the drain extracts water with a rate that is proportional to difference in head in the domain and the elevation of the drain. The constant of proportionality is the drain conductance, which in the original SEAWAT model is constant 100 m^2/day in all cells.

Fresh water enters the domain as recharge over the whole area. The recharge data is based on a timeseries of meteorological measurements that are carried out in the province of Zeeland. Evapotranspiration occurs when there is no recharge. An extended explanation on the derivation of the recharge and evapotranspiration data can be found in the report of Oude Essink and Pauw (2018). The model contains 84 layers, with thicknesses that vary between 0.2 m (near the extraction well) to 2 m in the lowest layers. The aforementioned and other model settings can be found in Table 2-1.

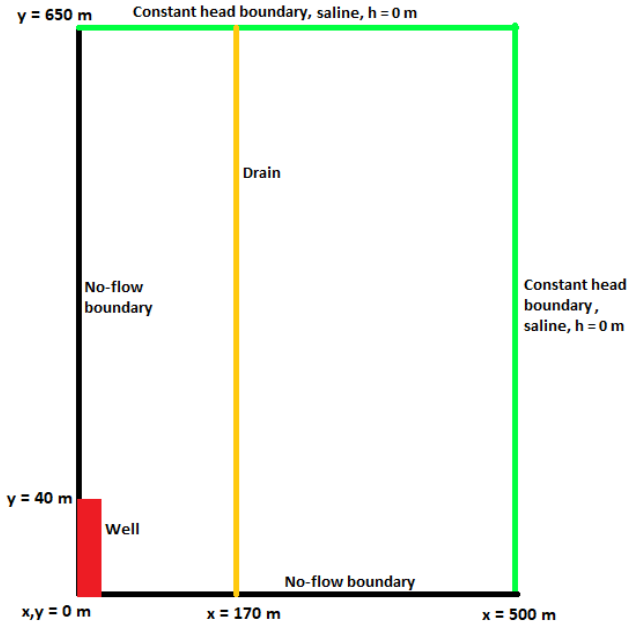

Figure 2-1. Plan view of model. Image is based on similar figure in Oude Essink and Pauw (2018)

Table 2-1. Model parameters and settings

X – dimension	500 m
Y – dimension	650 m
Thickness	41 m
Top	2 m +NAP
Number of layers	84
Well elevation	3 m -NAP
Total well extraction (if well is pumping)	30 m^3/day
Drain elevation	0.3 m +NAP
Initial hydraulic head	0 m
Initial concentration	16.394 kg Cl^-/m^3
DRHODC (slope of linear relation between density and concentration)	1.3419
Horizontal hydraulic conductivity	6.2 m/d
Horizontal hydraulic conductivity at x = 500 m in the upper layers	0.005 m/d
Vertical hydraulic conductivity	4.3 m/d
Vertical hydraulic conductivity at x = 500 m in the upper layers	0.005 m/d
Storage coefficient & specific yield above -1 m	0.15
Storage coefficient & specific yield below -1 m	10^{-5}
Longitudinal dispersivity	0.1
Transversal dispersivity	0.01
Diffusion coefficient	2.16 x 10^{-5} m^2/d
Porosity	0.38

2.2 Spatial discretization packages

MODFLOW 6 contains three different spatial discretization packages: normal discretization (DIS), discretization by vertices (DISV) and unstructured discretization (DISU) (Langevin et al., 2017). The DIS package is comparable to the discretization approach in MODFLOW-2005, as it only contains values of DELR (cell widths), DELC (cell lengths) and layer elevation. This discretization package can only be used for structured grids. The DISV package can be used for grids that are unstructured in the two-dimensional plane, but the same grid is applied for every layer in the domain. In this package the coordinates of all vertices are specified, which are the coordinates of the corners of every cell. This is illustrated in Figure 2-2: the blue numbers are the cells and the corners (the vertices) are denoted with the black numbers. The cells itself are also specified in the DISV package by listing the cell centers and denoting all vertices that define a cell in clockwise direction. These comprise the corner vertices of the cell itself, but also other vertices that are located on the edges of a cell. For example, cell 1 in Figure 2-2 consists of vertex 1, 2, 6 and 5, while cell 3 contains vertex 3, 4, 11, 10 and 7. The specification of vertex 7 is necessary to define the cell connections between cell 3 and cells 2 and 5 (Langevin et al., 2017). Since the DISV package still uses layers, the elevation of all layers should also be specified. MODFLOW 6 uses the data in the DIS and DISV package to derive other grid information like cell connectivity and cell dimensions.

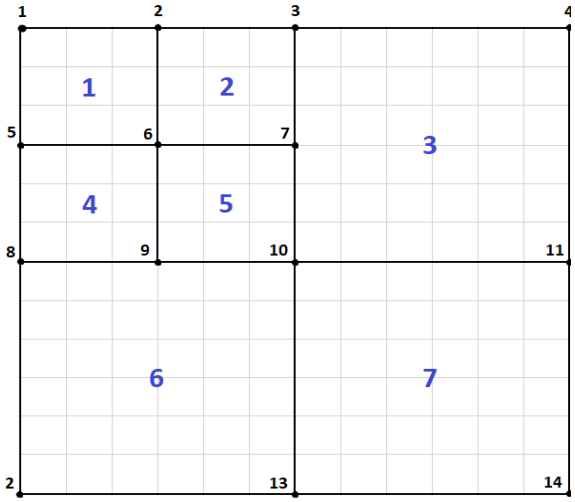

Figure 2-2. Grid cells and vertices

The DISU package is the most general spatial discretization package and can be used to define an unstructured grid in three directions. All information regarding cell connectivity and cell dimensions should be specified. For this study, a Python script (called '*Create_DISU*') is developed to create the input for the DISU package. The script assumes that the cell structure still contains layers such that all cells within a layer

have the same top and bottom elevation. Moreover, the script only works for cells that still have a rectangular (hexahedral) shape, which means that the script cannot be used to describe grids with irregular shapes like Voronoi, triangles, tetrahedra etc. The x- and y-coordinates of the vertices of the two-dimensional grid of one layer are the input for *Create_DISU*. To build a model that is vertically unstructured, i.e. that has a grid that is not the same for every layer, extra files containing the vertex x and y data for every new two-dimensional discretization in a layer should be added. *Create_DISU* creates a list of cell dimensions: area, height, center and all vertices that define a cell. Besides, the connection data is calculated for every cell: all connected cells to a certain cell (both in horizontal and vertical direction), the width of the horizontal connections, the area of the vertical connections and the direction of the connection. Moreover, it calculates the total number of cells, cell connections and vertices. The full script *Create_DISU* contains 481 lines, but some parts are shown in Appendix A .

2.3 SEAWAT to MODFLOW 6

The model input for the MODFLOW 6 model is based on the input for the original SEAWAT model as described in Oude Essink and Pauw (2018). With the use of another Python script (called '*mf2005_to_MF6_DISU*'), the content of the SEAWAT model files is translated to the correct format that can be used by MODFLOW 6. Parts of this script can be found in Appendix B .

The choice of discretization package defines in how a grid cell is identified, which is called the 'cell-id' (Langevin et al., 2017). This cell-id is used to assign the model input and output to the right cell. The different cell-ids are shown in Figure 2-3. This figure contains one stress period for the WEL package, created with the three different spatial discretization packages. The cell-id is shown first, followed by the extraction rate of that cell (-7.5 m^3/day). With the DIS package, the cell identification as in MODFLOW-2005 is used: the cell-id is the layer, row and column of the cell. In the DISV package the cells in a layer are numbered, which is shown in Figure 2-2. Because the same grid is applied for all layers, the cell-id for a DISV case contains the layer and the cell number. The DISU package specifies all cells individually, the cell-id when using the DISU package is therefore equal to the cell number.

```
BEGIN PERIOD 13
   10 1 1   -7.5
   10 2 1   -7.5
   10 3 1   -7.5
   10 4 1   -7.5
END PERIOD
```

```
BEGIN PERIOD 13
   10 1     -7.5
   10 52    -7.5
   10 103   -7.5
   10 154   -7.5
END PERIOD
```

```
BEGIN PERIOD 13
   8263   -7.5
   8314   -7.5
   8365   -7.5
   8416   -7.5
END PERIOD
```

Figure 2-3. DIS (top), DISV (middle) and DISU (bottom) input for the WEL package

Since for this study the model grid is modified, the input should also be adjusted to the correct grid. This is done in *mf2005_to_MF6_DISU* by reading the location of the cell centers of the reference model and the MODFLOW 6 model. Subsequently, the closest cell in the unstructured grid to every cell in the original grid is listed and vice versa. This way, all data of a certain cell in the reference model can be reassigned to a cell at the same location in the new model. It should be noted that this translation of data from one grid to another will have an impact on the results. For example, the constant head boundary of the used model is assigned to the cells at maximum x and y distance. When the cells in the new mesh are smaller compared to the reference mesh, the constant head boundary will also be smaller. As a result, the area containing active cells (variable head cells) will increase in size.

Besides the constant head package (CHD), there are other stress packages that will change when the spatial discretization is altered. First of all, the regions with a constant concentration will remain at the location of the constant heads, so this region will also be smaller when the model resolution increases. The same applies to the area of low hydraulic conductivity that is found below the constant head boundary at x = 500 m, which represents a ditch in the *poelgronden* (see section 2.1). Moreover, the drain at x = 170 m is kept at the width of one cell, but the drain conductances are scaled such that the total conductance of the whole drain is the same in every model.

The dependency of the area containing active (variable head) cells on cell sizes also has an impact on the recharge and evaporation volumes. The input for these stresses has the unit volume per unit area per unit time (LT^{-1}) (Langevin et al., 2017). This volumetric flux is multiplied with the cell area to yield the volume of water that is entering or leaving the system through recharge or evaporation respectively. However, the

evaporation and recharge flux is not applied to cells containing a constant head. If the constant head boundary is smaller due to smaller grid cells, the total volume of recharge and evaporation reaching the system is therefore increased. Lastly, the input for the well package is changed when the grid is refined or coarsened. The total well length and discharge are kept the same, meaning that the extraction rate for every cell has to be scaled accordingly.

The SEAWAT model will be run and will be compared to a MODFLOW 6 run with the same settings. The grid of both models is shown in Figure 2-4. To show that *Create_DISU* works correctly and to test whether the choice of spatial discretization package has any effect on the model output, the same MODFLOW 6 case will be run with the DIS, DISV and DISU package. Note that for these cases, the exact same grid (Figure 2-4) will be discretized with the DIS, DISV and DISU package. An overview of all these cases is shown in Table 2-2; all these cases have a simulation time of 40 years.

Table 2-2. Overview of reference cases in SEAWAT and MODFLOW 6

Case nr	Description	Number of cells	Cell size close to well (m)	Cell size far from well (m)
R-1	SEAWAT	77112	1 x 10	20 x 50
R-2	MODFLOW 6 – DIS	77112	1 x 10	20 x 50
R-3	MODFLOW 6 – DISV	77112	1 x 10	20 x 50
R-4	MODFLOW 6 – DISU	77112	1 x 10	20 x 50

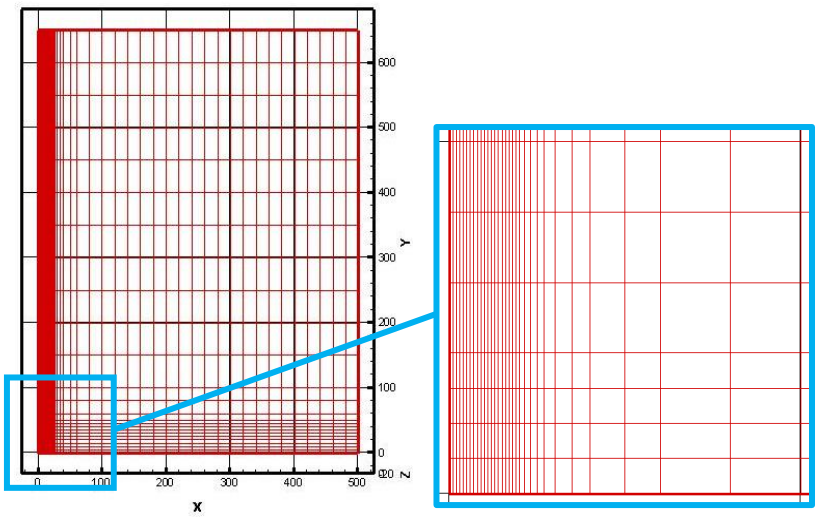

Figure 2-4. Plan view of the spatial discretization of cases R-1, R-2, R-3 and

2.4 Solver settings

Both SEAWAT and MODFLOW 6 use the total-variation diminishing (TVD) method to solve the advection part of the solute transport equation. This technique is mass-conservative. It aims to decrease the sum of concentration differences between neighboring cells for every new transport step. Usually, the numerical dispersion when using TVD is limited, but this increases the possibility of unwanted oscillations in the solution (Zheng and Wang, 1999). The solver settings are kept the same for all cases, since looking at the effect on solver settings is beyond the scope of this research.

The other parts of the solute transport equation is solved in SEAWAT with the use of the Generalized Conjugate Gradient (GCG) solver and the groundwater flow equation is solved with the Pre-Conditioned Conjugate Gradient (PCG) package. The Pre-Conditioned Conjugate Gradient method is also applied as linear solver for the flow part in MODFLOW 6. This is specified in the Iterative Model Solution (IMS) file. The exact settings that are specified in this file for the GWF and GWT model are shown in Appendix C .

2.5 Testing spatial discretization

To test the possible advantage of using unstructured grids for density-dependent flow and transport modelling with MODFLOW 6, the model grids will be systematically altered. The following terms will be used to describe different type of grids: fully regular, telescopic, horizontally unstructured and vertically unstructured. As explained earlier, all grids contain rectangular shaped cells.

Fully regular grids have the same cell size in the whole domain. *Telescopic grids* are also called non-equidistant grids, this type is shown in Figure 2-5. These grids are still regular in the sense that at a cell has only one neighboring cell at every side. However, the width and length of the cells may vary. These types of grids were originally implemented in previous versions of MODFLOW to create a refined grid at a point of interest in a model domain. Obviously, using this type of grid creates many unnecessary refined cells at a location far away from the point of interest. This shows the potential benefit of using MODFLOW 6, with which there is no more need to refine telescopically.

Horizontally unstructured grids are grids that are unstructured when looking at the two-dimensional plan view (see Figure 2-6). However, the same grid is applied for every layer in the domain. The term *'vertically unstructured grids'* is used to describe grids that are still considered a regular grid when looking at the two-dimensional plan view, but this regular grid is not constant for all layers. A schematic representation of the side view of such grid is shown in Figure 2-7. The top layers contain the original telescopic grid as shown in Figure 2-5, but in the lower layers (a part of) the cells are combined in pairs of two which results in a coarser grid. In this study, all grid types that are mentioned here are used in separate model runs, but the types are also combined. For example, some grids will be both horizontally and vertically unstructured.

When a horizontally or vertically unstructured mesh is implemented, care is taken that the cells have no more than two neighbors at every side. This measure is imposed to prevent numerical instabilities that could arise due to too many cell connections (Panday et al., 2013) (also see section 1.4). It will be tested whether a transition from four cells in the finer top layers to one cell in coarser lower layers may also be possible.

Several tests will be carried out to see the effect of spatial discretization. First of all, several fully regular grids will be used to determine if local refinement is necessary to produce accurate and fast results. This is important to know for cases where the model dimensions are much larger compared to the model dimensions in this study and refining the grid may not be possible. Secondly, vertical unstructured grids will be implemented, which will be used to determine the extent of implementing coarser grids without altering the results. Moreover, the effect on the model runtime due to a decrease in number of cells will be determined. Thirdly, horizontal unstructured grids are examined. To determine whether grids can also be coarsened in horizontal direction, several horizontally unstructured cases will be compared to telescopically refined cases that have the same cell size near the well. This comparison will show whether reducing the amount of unnecessary refined cells will positively influence the runtime without altering the model results. Lastly, some cases will be used to determine whether the model can be refined close to the well, without significantly increasing the runtime and/or changing the results. For this, a telescopically refined grid will be coarsened in both horizontal and vertical direction.

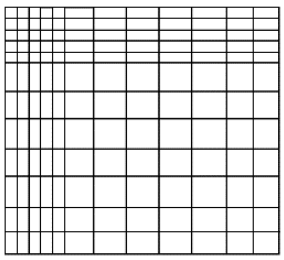

Figure 2-5. Plan view of a telescopic grid

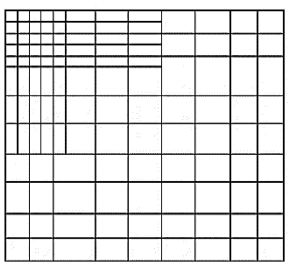

Figure 2-6. Plan view of a horizontal unstructured grid

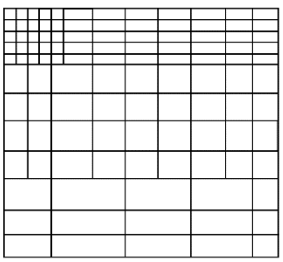

Figure 2-7. *Side* view of a vertical unstructured grid

The following subsections show all the cases that are used to test the effect of spatial discretization. An overview of the number of cells and cell sizes can be found in Table 2-3 and Table 2-4. All structured grids are shown in Appendix C and Appendix E . The grids are visualized with the use of the post-processing software Tecplot version 10 (Amtec Engineering, 2003). The unstructured grids are not visualized in Tecplot, because the correct input files for unstructured grids are unavailable (these input files are more extensive than the input for structured grids). However, three horizontal unstructured cases that have a similar structured counterpart are visualized with the use of Microsoft Excel and are shown in Appendix F

2.5.1 Vertically unstructured grids

Testing the effect of vertical coarsening uses the original discretization of the SEAWAT model at the top as the most refined layer. The coarsening is done by either combining cells in the direction of the x-axis (combining the cell widths in row direction), direction of y-axis (combining the cell widths in column direction) or in both directions at the same time. The last case means that at the transition to a coarser layer, one cell in the coarse layer is connected to four fine cells at its top side. Since the fine layer at the top is a telescopically refined grid, the vertical coarsening is done by either coarsening the cells in the telescopic area or by coarsening all cells in column and/or row direction. The telescopic area is shown in the left image in Figure 2-8. Coarsening the telescopic area is illustrated in the middle pictures in Figure 2-8. The middle top picture shows coarsening in the telescopic part in row direction: only the first six rows are combined and the rest of the grid is kept the same. The middle bottom picture shows coarsening over the full width in row direction, meaning that all rows are combined. For some cases an extra 'coarsening step' is added, meaning that an even coarser discretization is applied for the lowest layers. This is shown in the top right picture of Figure 2-8: the cells of the telescopic part of the previous grid (top middle) are now combined in column direction. The grids are shown in Appendix C , note that all layers still consist of a telescopic grid.

All vertically coarsened cases are listed in Table 2-3. The two-dimensional plan views of the grids are shown in Appendix C . This table shows the cell size near the well and in the base grid. The base grid is the grid far away from the telescopically refined area around the well. It is also shown in the left image in Figure 2-8: the yellow square indicates the base grid. The cell sizes indicate how the vertical coarsening is carried out. For example, the reference discretization at the top has a cell size of 1x10 m near the well and 20x50 m far from the well, case 1 (cell sizes 2x10 and 20x50 m) is therefore coarsened in the telescopic part in row direction and case 23 (cell sizes 1x20 and 20x100 m) is coarsened over the full width in column direction. All vertically coarsened cases have a simulation time of 40 years.

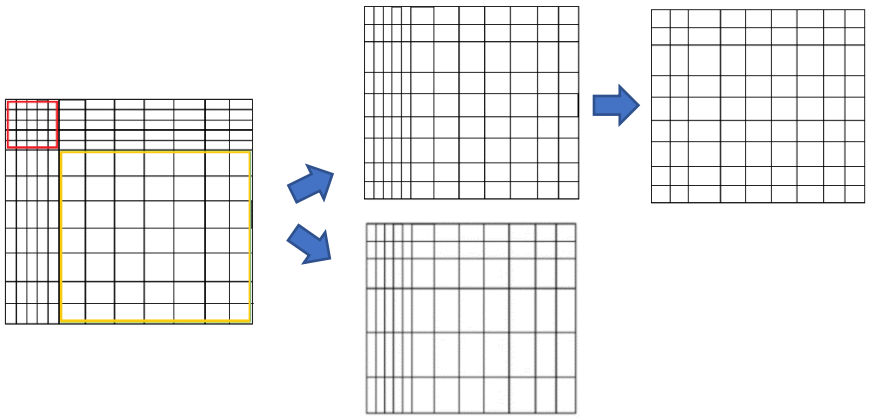

Figure 2-8. Schematic representation of vertically coarsening. Left: original grid. Red square indicates the telescopic part, yellow square is the base grid. Top middle: coarsening in column direction in only the telescopic part. Top right: further coarsening the telescopic part, in row direction. Bottom middle: coarsening the full grid in column direction.

The cases can be organized into five different groups (which are labelled A – E):

- Group A are the cases that are only refined in row direction in the telescopic part of the grid, the cases have different elevations of the transition to the coarser layer.
- Group B are all cases that contain a second coarsening step. The first coarse discretization is similar to one of three cases (2, 6 or 10) in group A, the elevation of the second transition is also varied.
- Group C contains four cases that are refined in column direction in the telescopic part of the grid.
- Group D contains all cases that are coarsened over the full width of the model, in column or row direction.
- Group E consists of cases where the cells in the telescopic part are coarsened in two directions together, resulting in a layer of coarse cells that are vertically connected to four fine cells.

Table 2-3. Overview of the vertically unstructured cases. All cases use the spatial discretization of the SEAWAT model in the fine layers at the top of the model. Coarse layer 1 is the first coarse discretization. Coarse layer 2 is a second coarse discretization in the same model.

Case nr	Group	Nr of cells	Cell size coarse layer 1 (m)		Elevation transition fine – coarse (m -NAP)	Cell size coarse layer 2 (m)		Elevation transition coarse – coarser (m -NAP)
			Close to well	Base grid		Close to well	Base grid	
1	A	73332	2 x 10	20 x 50	- 15			
2	A	68292	2 x 10	20 x 50	- 11			
3	B	66812	2 x 10	20 x 50	- 11	2 x 20	20 x 50	- 19
4	B	66072	2 x 10	20 x 50	- 11	2 x 20	20 x 50	- 15
5	B	64592	2 x 10	20 x 50	- 11	2 x 20	20 x 50	- 13
6	A	65772	2 x 10	20 x 50	- 9			
7	B	63552	2 x 10	20 x 50	- 9	2 x 20	20 x 50	- 19
8	B	62072	2 x 10	20 x 50	- 9	2 x 20	20 x 50	- 13
9	B	60592	2 x 10	20 x 50	- 9	2 x 20	20 x 50	- 11
10	A	63252	2 x 10	20 x 50	- 7			
11	B	61032	2 x 10	20 x 50	- 7	2 x 20	20 x 50	- 15
12	B	59552	2 x 10	20 x 50	- 7	2 x 20	20 x 50	- 13
13	B	58072	2 x 10	20 x 50	- 7	2 x 20	20 x 50	- 11
14	B	56592	2 x 10	20 x 50	- 7	2 x 20	20 x 50	- 9
15	A	60732	2 x 10	20 x 50	- 5			
16	D	70362	2 x 10	40 x 50	- 15			
17	D	61362	2 x 10	40 x 50	- 11			
18	D	52362	2 x 10	40 x 50	- 7			
19	C	74052	1 x 20	20 x 50	- 15			
20	C	69972	1 x 20	20 x 50	- 11			
21	C	67932	1 x 20	20 x 50	- 9			
22	C	65892	1 x 20	20 x 50	- 7			
23	D	70227	1 x 20	20 x 100	- 15			
24	D	61047	1 x 20	20 x 100	- 11			
25	D	51867	1 x 20	20 x 100	- 7			
26	E	71112	2 x 20	20 x 50	- 15			
27	E	63112	2 x 20	20 x 50	- 11			
28	E	59112	2 x 20	20 x 50	- 9			
29	E	55112	2 x 20	20 x 50	- 7			

2.5.2 Structured and horizontally unstructured grids

In this section, all structured and horizontally unstructured grids will be discussed. An overview can be found in Table 2-4. These cases all have a simulation time of 15 years, which is shorter than the vertically unstructured cases of the previous section. The simulation time is decreased because some cases that are discussed in this section will otherwise have a long runtime.

Cases 30 – 34 are fully regular grids to determine the minimum cell size. The cell size is varied between 20x50 m and 1x5 m. Cases 35 – 38 contain structured grids that are telescopically refined with different cell sizes near the well. Note that case 35 is exactly the same as reference case R-4, only the simulation time is shorter. Cases 39 – 41 are also telescopically refined grids, but these cases all have a cell size of 1x5 m near the well. For these cases, the 'base grid' cell size is varied. Cases 35 – 41 are shown in Appendix E .

Cases 42 - 44 are horizontally unstructured grids that have a 'telescopic counterpart' in cases 35 – 41. The objective of these cases is to determine whether a decrease in unnecessary refined cells will positively influence the runtime. Cases 42, 43 and 44 have a grid as in Figure 2-6: the discretization is the same as the telescopic counterpart, but there are no unnecessary refined cases. These cases are also visualized in Appendix F . Note that case 42 and 43 both have a cell size of 1x5 m near the well, but a different base grid cell size. Case 45 is also a horizontally unstructured grid, but this grid is created differently from cases 42 – 44. This case uses the grid of the reference SEAWAT model (shown in Figure 2-4) but is further refined close to the well. This is shown schematically in Figure 2-9. A cutout of the exact grid is shown in Appendix F .

Lastly, cases 46 – 50 contain unstructured grids that are vertically and/or horizontally coarsened. These cases will be used to determine whether implementing an unstructured grid can be used to reduce the runtime of a refined model. These cases are based on case 37; they all have a cell size of 1x1 m near the well, but they contain coarser cells at locations further away (both horizontally and vertically) from the well compared to case 37. Since case 37 is expected to have a higher runtime compared to less coarse cases, like 36 and 35, the cases 46 – 50 will indicate whether this increase in runtime can be undone by implementing an unstructured grid.

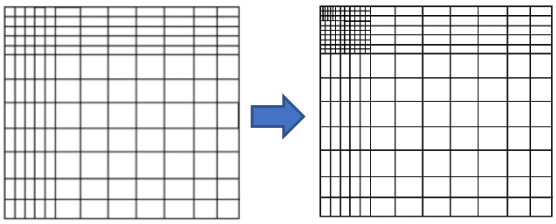

Figure 2-9. Schematic representation of the grid of case 45 (right) that is created by refining the reference grid (left)

2.6 Temporal discretization

The temporal discretization of MODFLOW comprises stress periods and timesteps. A stress period is a certain period of time over which all input variables or hydro(geo)logic stresses are constant. A stress period can be divided into timesteps, which are the periods that are used to solve the equations. The user specifies the length of the stress periods and the number of timesteps within that stress period (Hughes et al, 2017). Stress periods are developed for the convenience of the user to decrease the number of data entries and eventually only the timesteps are used in the model calculations.

In this study, the model input changes weekly, so the stress period length is one week. The time step length will be varied to determine the largest timestep that could still be used without significantly changing the results. This will be done by changing the NSTP variable, which is the number of timesteps within one stress period. The NSTP values that are tested are: 2, 3, 4, 7, 10 and 14, which equals a timestep length between 0.5 and 3.5 days. The original timestep length in the SEAWAT model is one week, which is also the timestep that is used for the spatial discretization tests. The cases are run for 40 years.

Table 2-4. Overview of the fully regular, telescopic, horizontally unstructured and combined cases.

Case nr	Type	Nr of cells	Cell size fine layer(m)		Cell size coarse layer(m)		Elevation transition fine – coarse (m -NAP)
			At the well	Far from well	Close to well	Far from well	
30	Fully regular	27300	20 x 50	20 x 50	-	-	
31	Fully regular	43680	25 x 25	25 x 25	-	-	
32	Fully regular	273000	10 x 10	10 x 10	-	-	
33	Fully regular	1092000	5 x 5	5 x 5	-	-	
34	Fully regular	5460000	1 x 5	1 x 5	-	-	
35	Horizontal telescopic refined	77112	1 x 10	20 x 50	-	-	
36	Horizontal telescopic refined	102816	1 x 5	20 x 50	-	-	
37	Horizontal telescopic refined	239904	1 x 1	20 x 50	-	-	
38	Horizontal telescopic refined	1985088	0.1 x 0.1	20 x 50	-	-	
39	Horizontal telescopic refined	217560	1 x 5	10 x 25	-	-	
40	Horizontal telescopic refined	413280	1 x 5	5 x 20	-	-	
41	Horizontal telescopic refined	705600	1 x 5	5 x 10	-	-	
42	Horizontal unstructured	68544	1 x 5	20 x 50	-	-	
43	Horizontal unstructured	153216	1 x 5	10 x 25	-	-	
44	Horizontal unstructured	144648	1 x 1	20 x 50	-	-	
45	Horizontal unstructured	251328	0.125 x 0.15625	20 x 50	-	-	
46	Horizontal telescopic and vertical unstructured	204624	1 x 1	20 x 50	2 x 1	20 x 50	-9
47	Horizontal telescopic and vertical unstructured	185199	1 x 1	20 x 50	2 x 1 2 x 2	20 x 50 20 x 50	-9 -13
48	Horizontal telescopic and vertical unstructured	172179	1 x 1	20 x 50	2 x 1 2 x 2 4 x 2	20 x 50 20 x 50 20 x 50	-9 -11 -13
49	Horizontal and vertical unstructured	117783	1 x 1	20 x 50	2 x 1	20 x 50	-9
50	Horizontal and vertical unstructured	108058	1 x 1	20 x 50	2 x 1 2 x 2	20 x 50 20 x 50	-9 -13

2.7 Postprocessing

The results are quantified by using the fresh water volume in the domain over time, the location of the interface between fresh and saline groundwater in the domain, chloride concentration of the water in the well and by checking the water volume and solute mass balances. Analogous to the study of Oude Essink and Pauw (2018), fresh water is defined as water containing 1500 mg or less chloride per liter water, which is also explained in section 1.2. A Python script (called '*produceoutput_mf6_DISU*'), that is based on a script that was developed by Oude Essink and Pauw (2018), is used to post-process the results.

The fresh water volume is calculated by taking the sum of the volumes of the cells that have a concentration below this threshold for fresh water. The concentration in the well is approximated by the concentration of the cell at x,y = 0,0. Note that this is an approximation of the real concentration of the well discharge, since the well is located in multiple cells that can contain slightly different concentrations.

The water volume balance contains the total volume (m^3) of the fluxes that are flowing into and out of the system. The mass balance contains the mass of solute (kg CL^-) in the system. These balances are respectively written in the .lst file of the flow part (gwf.lst) and the transport part of the simulation (gwt.lst). The program calculates the difference between the in- and outflowing volume or mass and reports this 'mass balance error' as a percentage. Generally, it can be assumed that the model results are good if the error is below 1 %, in this study however, the solute mass balance error is often slightly higher.

3 Results

This chapter presents the results of this study. As examples of output produced by the model, the interface between the fresh and salt water in the whole domain is shown in Figure 3-1, the concentration distribution after 40 years of case R-4 is shown in Figure 3-2 and. These figures show that after 40 years, the deepest point of the fresh and salt water interface (defined at 1.5 kg Cl^-/m^3) is at about -11 m NAP.

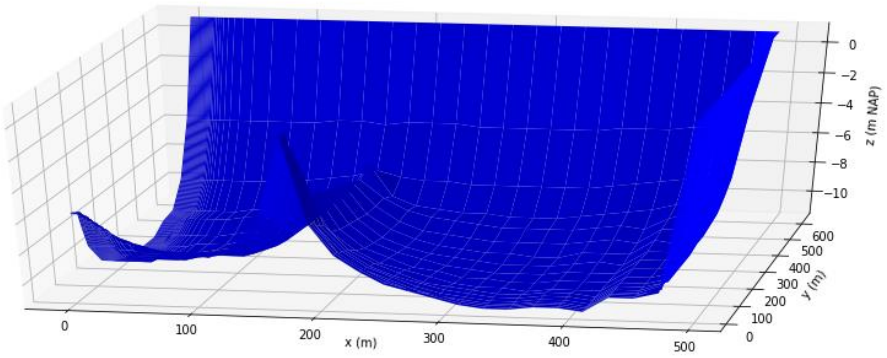

Figure 3-1. 3D plot of the interface between fresh and salt water after 40 years of case R-4 (MODFLOW 6).

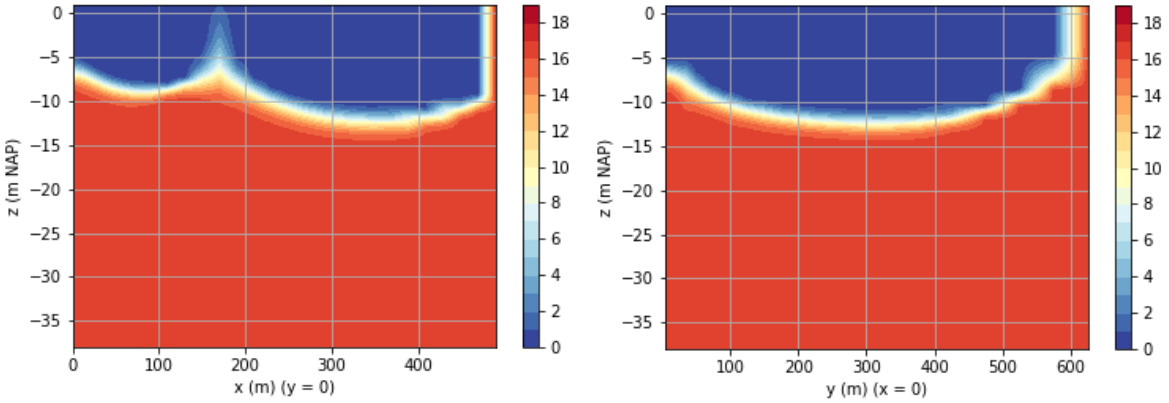

Figure 3-2. 2D plot of concentration distribution along the y-axis (left) and x-axis (right) of case R-4 (MODFLOW 6)

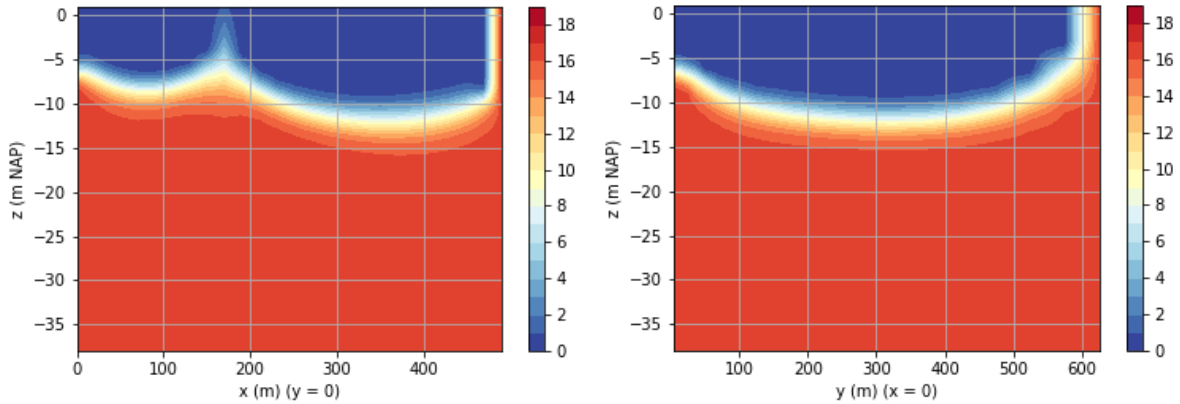

Figure 3-3. 2D plot of concentration distribution along the y-axis (left) and x-axis (right) of case R-1 (SEAWAT)

3.1 SEAWAT and MODFLOW 6

The reference SEAWAT model with a telescopic grid (case R-1) is translated to MODFLOW 6; the results of both models will be shown here. The model of MODFLOW 6 is run with all three discretization packages (DIS, DISV and DISU) separately (cases R-2, R-3 and R-4) to determine whether the choice of discretization package has any effect on the model results. Moreover, cases R-2 – R-4 are run with and without the SAVE-SPECIFIC-DISCHARGE (SSD) option in the Node Property Flow (NPF) package. This will be further explained below. All cases in SEAWAT and MODFLOW 6 use the TVD solver. The results are listed in Table 3-1. Figure 3-4 shows the fresh water volume in the whole domain for the SEAWAT and MODFLOW 6 cases and Figure 3-5 shows the concentration Cl^- in one cell that contains the well. These plots only show case R-2 of the MODFLOW 6 cases, since the other two discretization packages produce plots that are indistinguishable from case R-2.

The results of the cases with three spatial discretization packages in Table 3-1 show negligible differences and also the runtimes of the three runs are the same. This indicates that the type of discretization (DIS/DISV/DISU) does not influence the results, as long as the same grid is applied. The differences between SEAWAT and MODFLOW 6 are more pronounced. This also becomes clear when comparing the figure of the final concentration distribution along the x- and y-axis of the MODFLOW 6 case R-4 (Figure 3-2) and of the SEAWAT case R-1 (Figure 3-3). Note that the water balance in SEAWAT is reported in mass instead of volume, but because of the variable density of the water this mass cannot be easily converted into a volume. Figure 3-4 shows that the fresh water volume of MODFLOW 6 is constantly higher compared to SEAWAT. Moreover, the SEAWAT model seriously over- and undershoots the concentration for the first 15 years. The graph is cut off, the maximum reported concentrations are 41.8 kg Cl^-/m^3. When the SEAWAT solution stabilizes, the concentration in the MODFLOW 6 model are slightly lower compared to SEAWAT. The runtime of SEAWAT is almost twice as high as MODFLOW 6. This difference between the results of MODFLOW 6 and SEAWAT is remarkable since both models use the TVD method to solve the advection term of the solute transport equation. Also, the high solute mass balance error of MODFLOW 6 is strange, since the TVD solver is mass conservative.

In the Node Property Flow (NPF) package, one of the options is to save the specific discharge ('SAVE_SPECIFIC_DISCHARGE'). Although this should be an output control option, it has an impact on the results of the model, which is shown in Table 3-1 and Figure 3-4. The impact of this option on the final concentration distribution along the x- and y-axis is shown in in Appendix G . The results with this option specified show a more gradual interface between fresh and salt water and are closer to the results produced by SEAWAT. When this option is switched off, this interface becomes much sharper. Moreover, the plot of the head distribution changes, but these changes are much less pronounced compared to the concentration plots. Since this option should have no effect on the model output, it is switched off for all other cases, to prevent influencing the results. The problem has been reported to the USGS.

Table 3-1. Results cases R-1 – R-4. All cases contain the same grid. The last row shows the results of case R-1 with the SSD turned on in the NPF package

Case nr	Description	Runtime (min)	Final fresh water volume (m^3)	Incoming solute mass (kg)	Outgoing solute mass (kg)	Solute mass balance error (%)	Incoming water volume (m^3)	Outgoing water volume (m^3)
R-1	SEAWAT	53	1088927	78733530	78733540	0.0	10588807525 kg	10588804725 kg
R-2	MF6 – DIS	27	1233493	71774759	70004985	2.5	10433975	10433193
R-3	MF6 – DISV	27	1233417	71771650	70001907	2.5	10434005	10433222
R-4	MF6 – DISU	27	1233402	71774912	70003846	2.5	10433983	10433217
-	MF6 – DIS SSD on	26	1173565	72072174	69956531	2.98	10439016	10438089

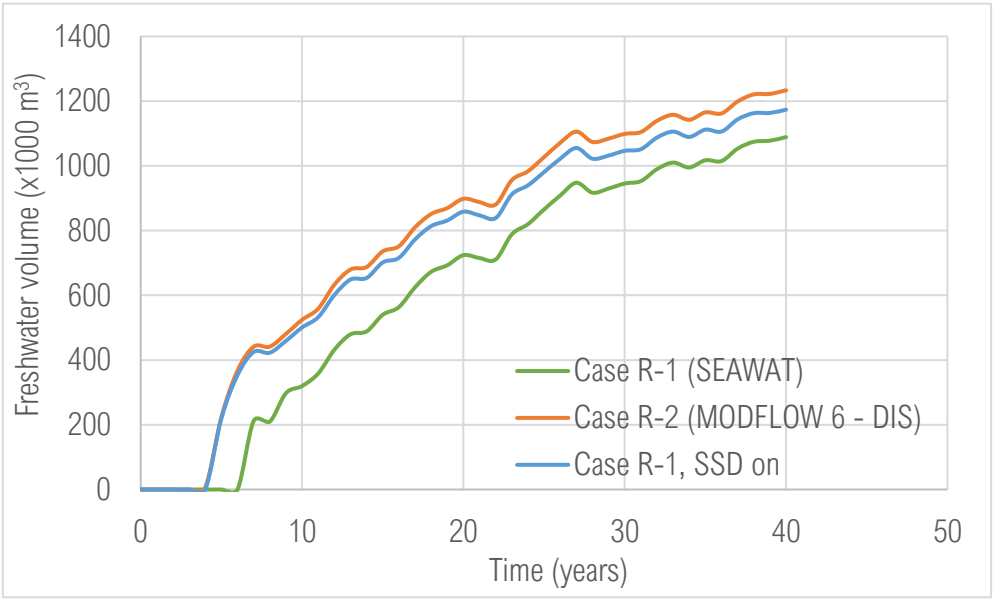

Figure 3-4. Fresh water volume (x1000 m³) over time (years) in case R-1 and R-2. The blue line shows case R-1, where also the SAVE_SPECIFIC_DISCHARGE setting is

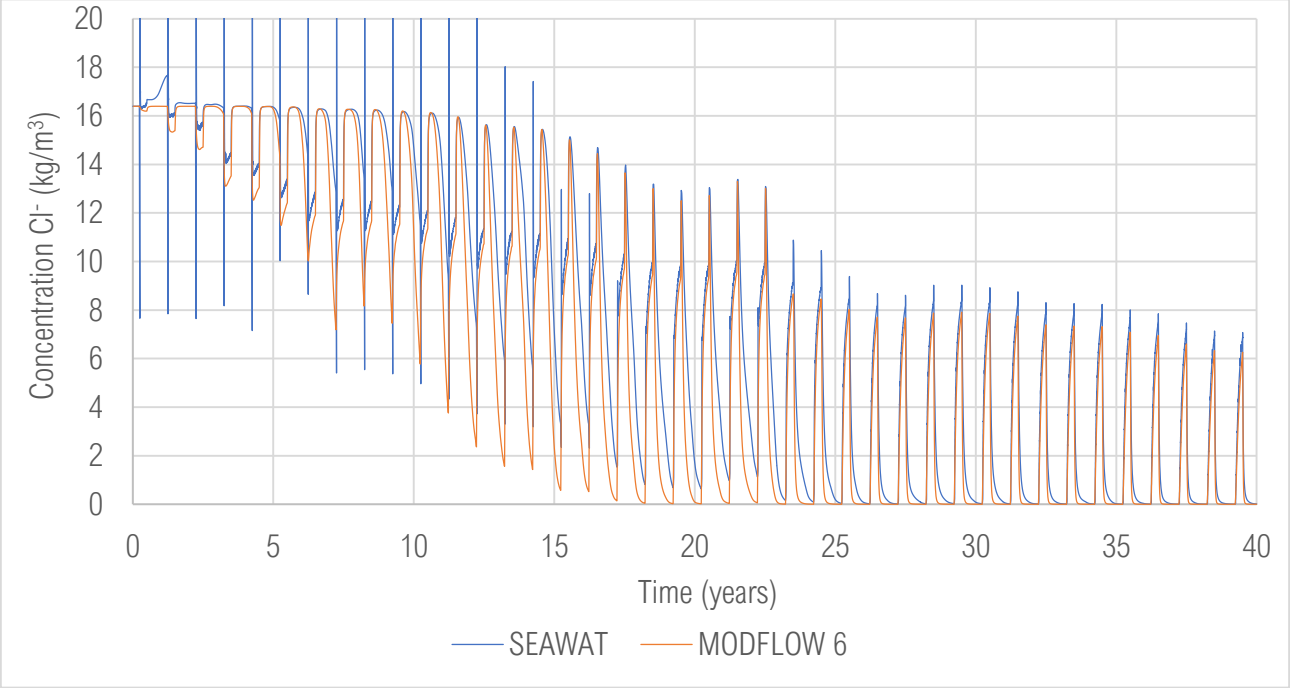

Figure 3-5. Concentration (kg Cl^-/m^3) in one cell containing the well for SEAWAT and MODFLOW6 model

3.2 Case 1 – 29: Vertically unstructured grids

The first 29 cases are used to test the effect of vertical coarsening and are compared to the results of MODFLOW reference case R-4. The final solute mass balance is shown in Figure 3-7. The concentration Cl^- in the well in of all cases is shown in Figure 3-6, the cases that deviate generally more than 0.1 kg/m^3 from the reference case are labelled. The volume of fresh water in the domain over time is not shown, since most cases have essentially equal plots; the difference between the volume of all cases remains below 3000 m^3, which is 0,24% of the total averaged volumes, except for six cases (16, 17, 18, 23, 24 and 25). The same cases stand out with regard to the solute mass balance (Figure 3-7). The water volume balance (not shown here) shows the same pattern: almost all cases show no significant difference in result, except the previously mentioned cases. All vertically unstructured grids are subdivided into five groups (see subsection 2.5.1); every group will be discussed separately here.

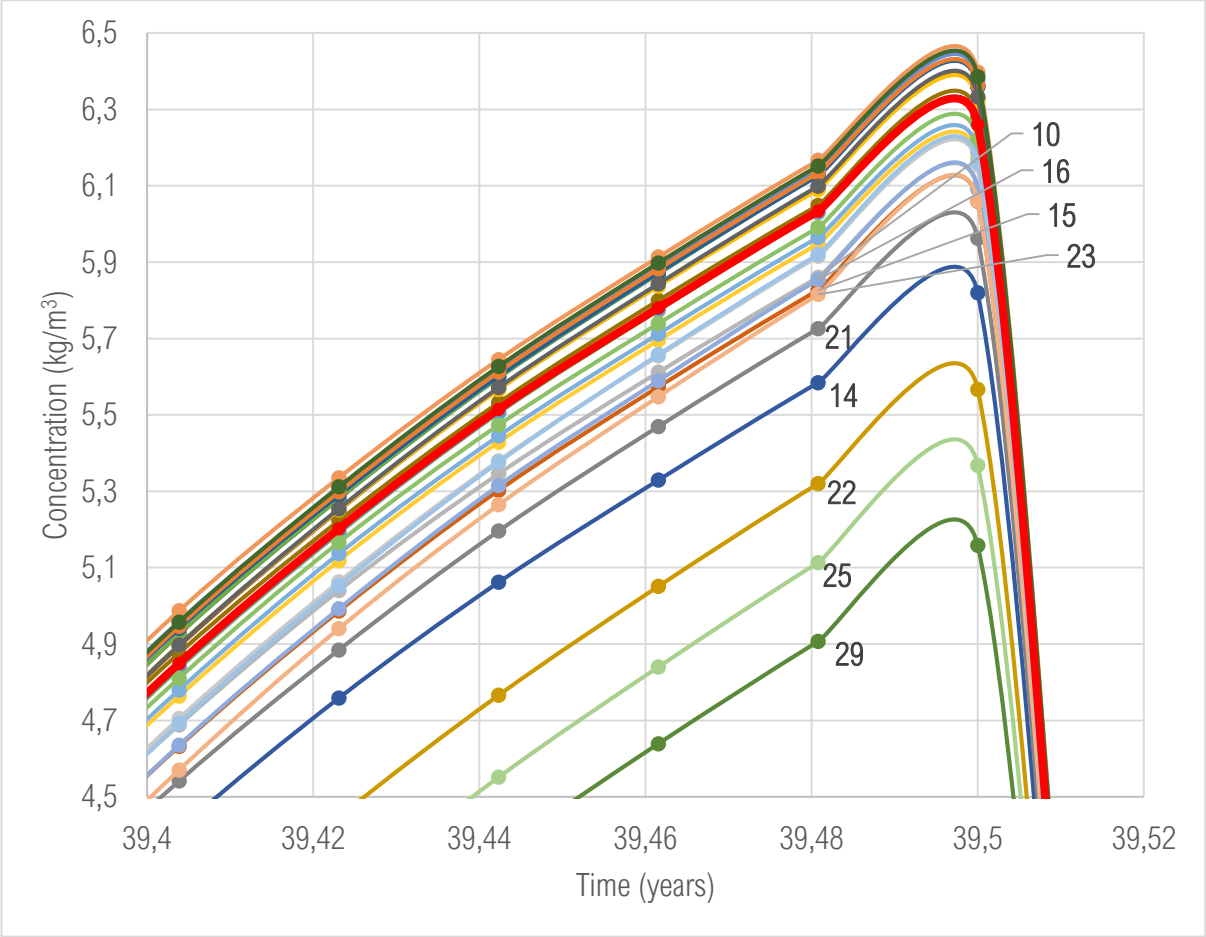

Figure 3-6. Last chloride concentration (kg/m^3) peak at the well of case 1 - 29. The bold red line is the reference case (R-4). The cases that in the full simulation generally differ more than 0.2 kg/m^3 from case R-4 are labelled.

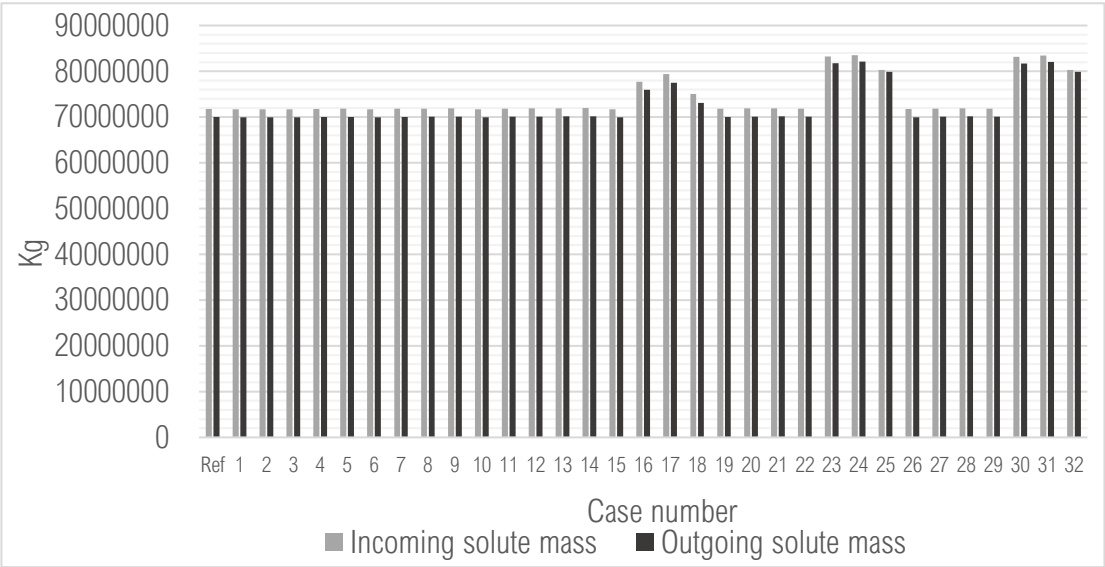

Figure 3-7. Total incoming and outgoing solute mass (kg Cl^-) after 40 years for all vertical coarsened cases. Ref = case R- 4

Group A (cases 1, 2, 6 10 and 15) are the cases that are coarsened in the telescopic area in row direction. The elevation of the transition from the fine to the coarse layer increases (becomes less deep) from case 1 to case 15. The first three cases (1, 2 and 6) do not influence the results at all, case 10 and 15 are slightly different. Both cases have a chlorine concentration at the well that is slightly lower at the final peak compared to the reference case: around 0.2 kg/m^3. This indicates that for this model, these types of coarser layers can be implemented up to -9 m; between -9 and -5 m the results will be slightly influenced but not significantly. This is a remarkable result, since the interface between fresh and saline groundwater is at -11 m, so it is possible to coarsen the telescopic area in the layers that contain the interface at one point.

Group B (cases 3-5, 7-9 and 11-14) are the cases that contain an extra coarse layer at the lowest elevations. This extra coarse discretization consists of cells that are only coarsened in the telescopic area, but in both the row and column direction. This extra coarsening layer has generally no impact on the model results, except for case 9, 13 and 14. These cases show some deviations in the interface of fresh and saline groundwater (not shown here). Case 14 also show a significantly lower chloride concentration in Figure 3-6.

Case 13 and 14 contain case 10 that is further coarsened in the lower layers: respectively at -11 and -9 m. This means that there are respectively 20 and 10 layers between the finest and the coarsest discretization Case 9 is based on case 6 and is further coarsened below -11 m (with 10 layers in between). These cases indicate that a rapid increase from fine to coarse cells in vertical direction can have a negative result on the model output. This negative effect becomes more severe when the transitions to coarse layers are at less deep elevations. This is further proven by case 5, which also contains 10 intermediate coarse layers, but the elevation of the first transition is at -11 m, which is low enough for the results to be accurate.

Group C (cases 19 – 22) are coarsened in one step, but now in column direction. The elevation of the transition from fine to coarse layers increases (becomes less deep) from case 19 to 22. The two coarsest cases (21 and 22) produce less accurate results compared to the reference case: the chloride concentration in the well is significantly lower (see Figure 3-6). Case 20 is still correct with regard to this concentration, but there are some irregularities in the fresh and saline water interface. Altogether it can be concluded that coarsening in column direction results in larger differences in results compared to coarsening in row direction. This is a sensible result, because coarsening in column direction results in cells of 1x20 m. This means that the cell center shifts 5 m, whereas coarsening in row direction only causes the cell center to

shift 0.5 m. As a result, the connection between two vertically connected cells at the transition will be worse with regard to the requirements for a 'good' grid (also see section 1.4) (Narasimhan and Witherspoon, 1976; Panday et al., 2013). However, if this coarse discretization is located at a low enough elevation (case 19), the results will not be influenced.

Group D consists of all cases where the cells are coarsened in the whole area, instead of only in the telescopic part. This group contains all cases that stand out with regard to the solute mass balance and fresh water volume. The coarse discretization contains base grid cell sizes of 40 x 50 m (cases 16, 17, 18) or 20 x 100 m (cases 23, 24, 25). It should be noted that, since the base grid cell size of these cases is increased in the lower layers, these cases have a larger area that is a constant head boundary in the lower layers. However, the coarser discretization is mostly present below the final fresh-saline groundwater interface, so the possible influence of this wider boundary is limited. Still, the increase in area of constant concentration at these boundaries could explain why the total in- and outflowing solute mass is higher for these cases. The images of the final fresh and saline groundwater distribution, of which two are shown in Figure 3-8, indicate that these large base grid cell sizes have influenced the results. For example, for case 18, the upconing of saline groundwater towards the drain at x = 170 m is not clearly visible anymore. These images illustrate that the base grid cell size in the coarse layers of cases 16 – 18 and 23 – 25 were too large and has had a significant impact on the results. The fact that case 16 and 23 are also deviating from the reference case is interesting, since the coarse discretization is 4 m (20 model layers) below the final fresh – saline groundwater interface. Implementing this coarser grid at even lower layers is not done in this study, because only 15 layers are present in the bottom 24 m of the model, so the possible decrease in runtime would be negligible.

Group E contains four cases (26-29) that have been run where the coarse discretization contains cells that are coarsened in both directions (in the telescopic part), meaning that one coarse cell is vertically connected to four fine cells at the transition. Case 29 strongly deviates: the chloride concentration during the last peak is more than 1 kg/m^3 lower compared to the reference case. Except for case 26, all cases show significant distortions in the fresh and saline groundwater interface. This deviation is to be expected, since the model results will improve for grids with cells that have at maximum two neighbors at every side (Panday et al., 2013), which is also related to the aforementioned requirements for a good grid. Case 26 shows that it is still possible to implement a coarse layer that results in cells that are vertically connected to four finer cells, as long as this elevation is low enough (below -15 m). However, the cases that contain a less coarse layer in between (3-5, 7-9, 11-14) show that the coarsest discretization can be at -13 m, but there should be a 'buffer' layer in between.

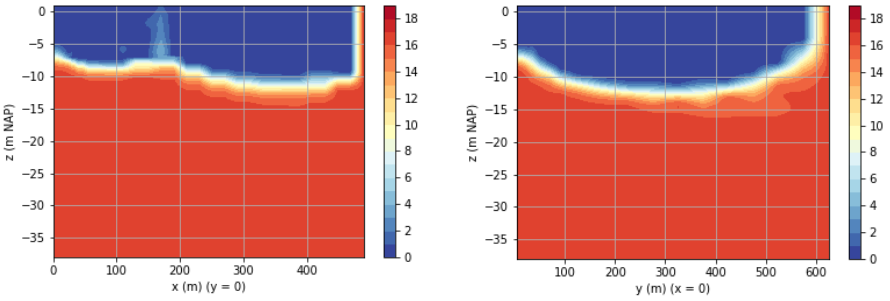

Figure 3-8. Two examples of chloride concentration (kg/m^3) from different cases with inaccurate interfaces between fresh and saline groundwater after 39 years. Left: case 18, right: case 23.

The runtimes of all vertically unstructured cases are plotted in Figure 3-9. The increase in runtime seems to be linearly related to number of cells. The deviating cases with respect to the solute mass balance and/or the chloride concentration in the well are also shown in this figure (blue dots). The runtimes of these cases are all, except one, slightly above the linear trendline, meaning that their runtime is relatively longer. This shows that grids that produce less accurate results introduced a higher degree of error into the results which results in longer runtimes. The cases with a slightly deviating fresh – saline water interface, but with a correct chloride concentration in the well are shown with the green dots. These cases are still incorporated in the calculation of the linear trendline. Although these cases produce slightly incorrect results, the most important parameters (concentration at the well and total volume of fresh water) are still predicted correctly.

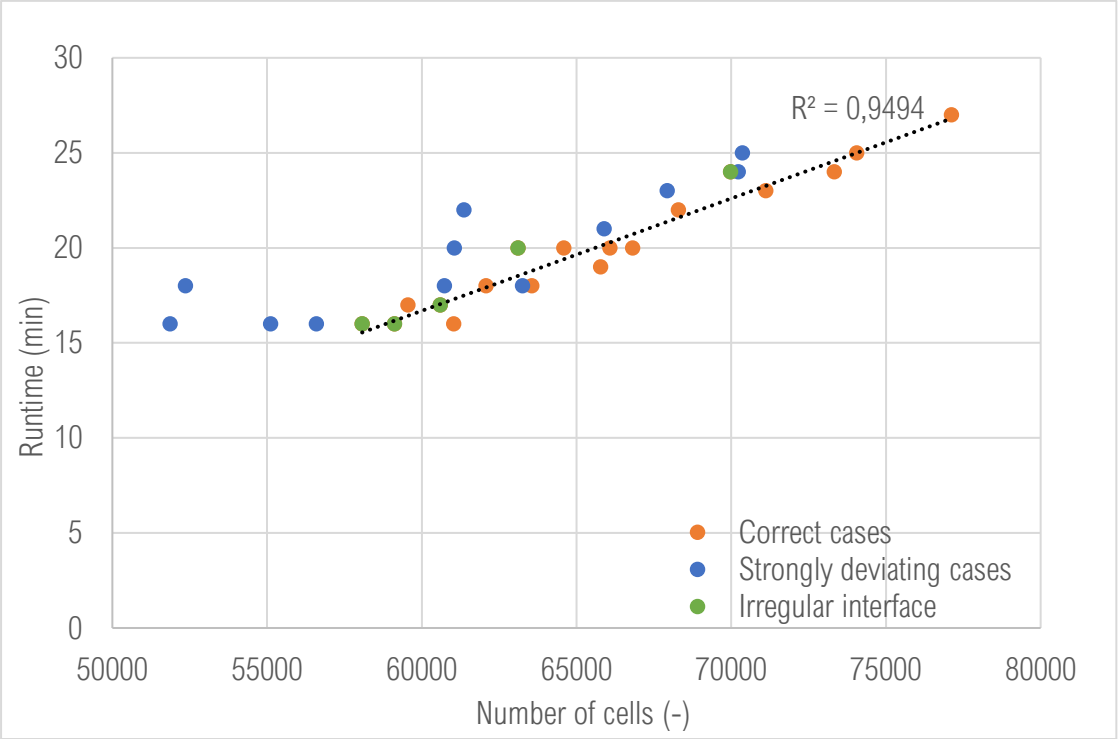

Figure 3-9. Runtime (minutes) versus number of cells (-) for all vertically unstructured cases. The blue dots are the cases with deviating results with respect to the solute mass balance or the chloride concentration in the well. The green dots are the cases that produce correct data but show an irregularity in the fresh-saline groundwater interface. The orange dots are all cases that are correct in every aspect. The trendline includes the orange and green dots

3.3 Case 30 – 50

This section shows the results of case 30 to 50. First, the cases with a structured grid will be shown (case 30 – 41). These structured cases can be subdivided into three groups: fully regular grids (shown in subsection 3.3.1), telescopic grids with different cell sizes near the well and telescopic grids with different cell size far away from the well (both shown in subsection 3.3.2). Subsection 3.3.3 reviews the runtimes of all structured cases. Subsection 3.3.4 shows the results of the horizontally unstructured cases (42 – 45) and the last subsection shows the cases that are both horizontally and vertically unstructured (46 – 50). The solute mass and water volume balance for all these cases are shown in Figure 3-10 and Figure 3-11, respectively.

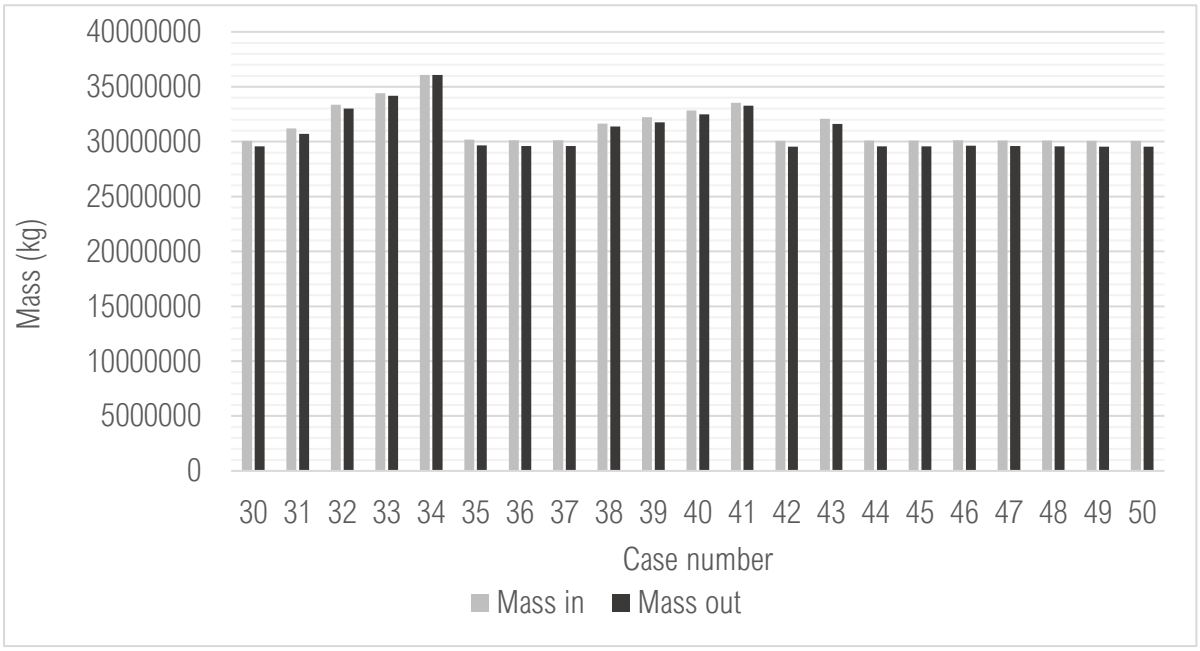

Figure 3-10. Total solute mass balance (kg) after 15 years

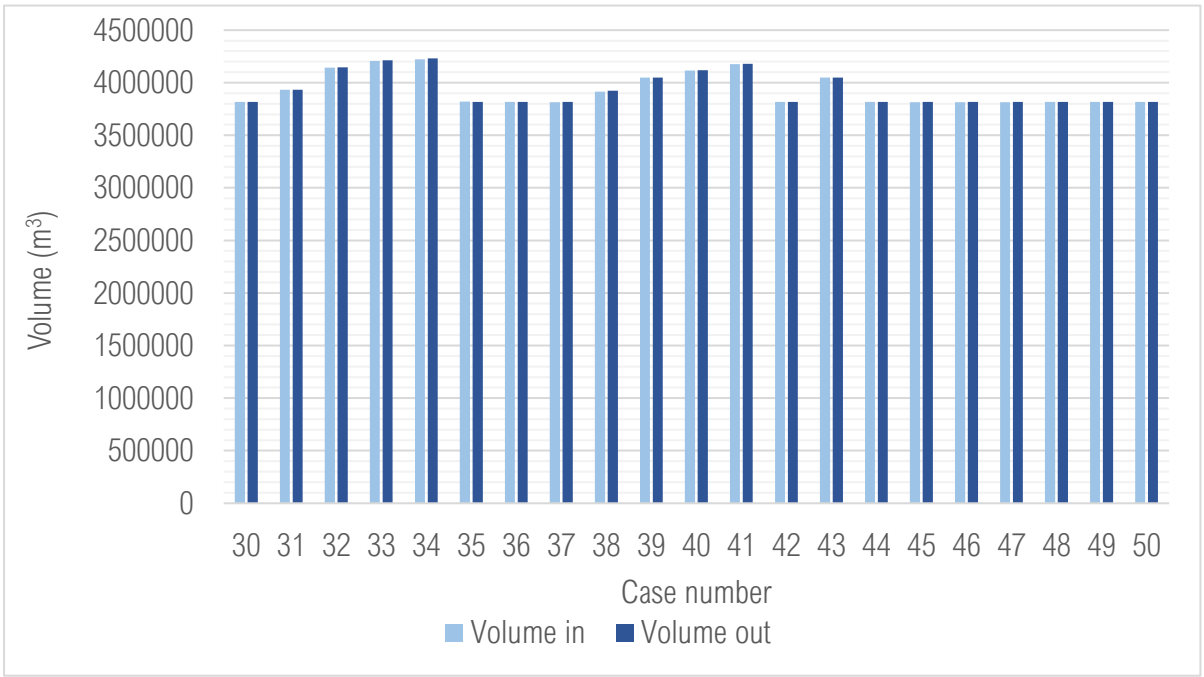

Figure 3-11. Total water volume balance (m^3) after 15 years

3.3.1 Case 30 – 34: Fully regular grids

The results of the runs with a fully regular grid (case 30 – 34) are shown in Table 3-2. Contrary to cases 1 – 29 that are previously discussed, the volume of fresh water in the domain is different when comparing these cases, which is shown in Figure 3-13. The largest final fresh water volume (case 33) is 5% higher than the smallest volume (case 30). The chloride concentration at the well during the full simulation is shown in Figure 3-14.

Table 3-2. Results of the cases with fully regular grids

Case nr	Cell size (m)	Number of cells	Runtime (min)	Final fresh water volume (m^3)	Solute mass balance error (%)	Water volume balance error (%)
30	20x50	27300	2	733438	1.73	0
31	25x25	43680	4	744159	1.6	0.02
32	10x10	273000	42	769405	1.01	-0.03
33	5x5	1092000	347	772604	0.74	-0.09
34	1x5	5460000	2612	753754	-0.03	-0.2

The total in- and outgoing mass and volume (see Figure 3-10 and Figure 3-11) increases with decreasing cell size. For example, the total outgoing solute mass is 18% higher for case 34 (cell size 1x5) compared to case 30 (cell size 20x50). The solute mass balance error also decreases from case 30 tot case 34, which suggests that the model solution becomes more accurate when the cell size decreases. The water volume balance remains low. However, the increase in incoming and outgoing fluxes can also be due to a change in boundary stresses. The constant head boundary is located at the edges of the model domain, the row or column of cells at y = 650 and x = 500. This means that the volume of active cells (i.e. cells with a variable head) increases for refined cases, since the cells that contain a constant head are smaller. As a result, the volume of recharge that reaches the system also increases, because MODFLOW 6 applies the recharge flux only to the active cells. Figure 3-13 shows that the total volume of fresh water in the domain increases for a decreasing cell size. The cases with cell sizes of 10x10 m and 5x5 m (case 32 and 33) show a very similar plot: the final volume of fresh water only differs with 0.4%. Remarkably, the volume of fresh water for case 34 (cell size 1x5 m) is lower. This is unexpected, since the total volume of active cells is larger compared to the other cases.The fresh-saline groundwater interface is more diffuse compared to the other cases (see Figure 3-12), which suggest that there is more mixing of fresh and saline water, which explains the lower volume of fresh water that is in the system. A possible explanation for this can be seen in Figure 3-12: the constant head and constant concentration at x = 500 m of case 34 is very narrow (1 m), so the concentration gradient is large over a very short distance. This results in some unphysical behavior that may cause the saline water to reach higher elevations. More cases should be run to see if a slightly wider boundary could inhibit this effect.

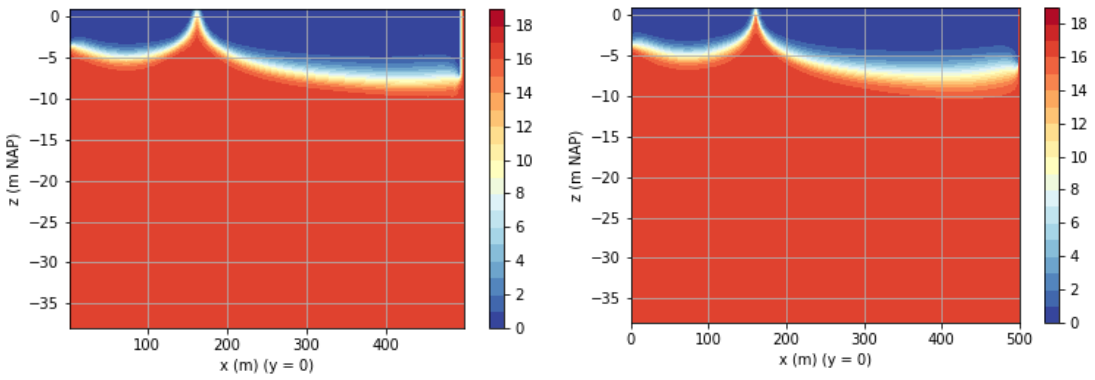

Figure 3-12. Chloride concentration (kg/m^3) of case 33 (left) and 34 (right) at t = 15 years along the y-axis

The chloride concentration of the water at the well varies significantly when comparing these cases, which is plotted in Figure 3-14. For example, the difference in peak concentration between case 30 and 34 during the last peak (at 14.5 years) is almost 5 kg/m^3. The concentration at the well is constantly higher for cases with smaller cell sizes. Especially the two coarsest grids (case 30 and 31) have a concentration profile that deviates. They show almost no decrease in concentration during the first three years and after that, the plots have a more irregular pattern compared to the finer grids. The differences between the other three cases are also large. During the first six years, the concentration peaks at the well are highest for case 34 (1x5 m), but also the lowest during the low concentration periods. After six years, the concentration is generally higher during the whole year for case 34, while case 32 and 33 have lower concentrations. These differences can be as large as 3 kg Cl^-/m^3, which is a significant discrepancy (also see section 1.2 for the classification of fresh and salt water). Figure 3-14 also shows that the peak concentration, which will be reached when the well is pumping, has a larger overall decline for cases with a larger cell size, whereas for the case with cell size 1 x 5 m (case 30), this decline is minimal.

In conclusion, the fully regular cases show that a cell size of 20x50 and 25x25 m is unsuitable to use for this model. The mass balance error is low for the other three cases, but the chloride concentration at the well and volume of fresh water show that the cell sizes still influence the results. However, the runtimes also increase sharply for these cases, so to determine the best option it should be considered whether the solution should be more accurate or computationally less demanding.

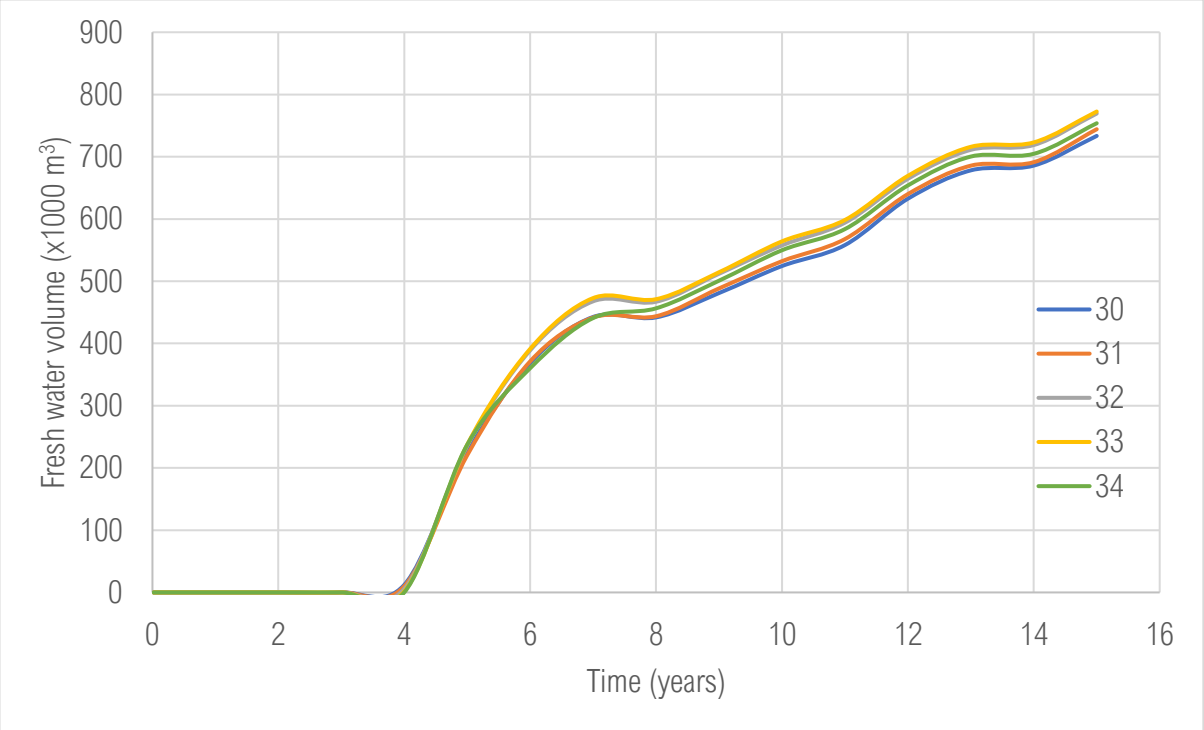

Figure 3-13. Volume of fresh water (x1000 m^3) over time (years) for all cases with a fully regular grid. The cell size decreases from case 30 to 34.

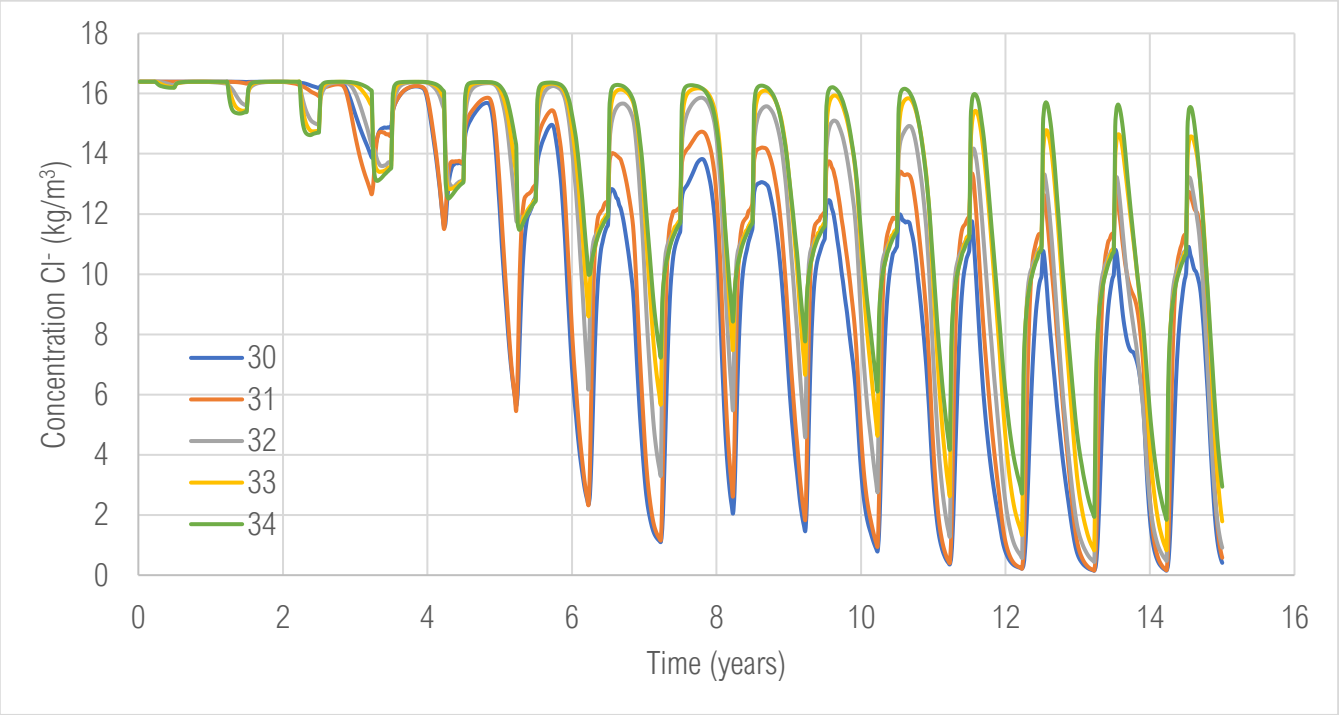

Figure 3-14. Concentration Cl^- at the well of all cases with a fully regular grid. The cell size increases from case 30 to 34.

3.3.2 Case 35 – 41: Telescopic refinement

The telescopically refined cases can be divided into two parts. The first cases (35-38) all have a cell size of 20 x 50 m far away from the well ('base grid'), based on the original grid of the SEAWAT model, and the cell size near the well is changed. Case 39 – 41 all have a cell size of 1 x 5 m near the well and the cell size far away from the well ('base grid') is changed. The results are listed in Table 3-3, the volume of fresh water for the full simulation is shown in Figure 3-15, and close ups of the chloride concentration during the last year are shown in Figure 3-16 and Figure 3-17.

The fresh water volume plots of case 35, 36 and 37 are almost the same, but the fresh water volume in case 38 is significantly higher than the other cases (about 3%). Similarly, the chloride concentration at the well of case 38 differs slightly. Case 38 is the only case where the cell size near the well is decreased in row (x) direction. The similarity between case 35, 36 and 37 indicates that a cell width of 10 m in column (y) direction is fine enough, but case 38 suggests decreasing the cell width in the row direction will influence the results. The fact that case 38 has a solute mass balance error below 1% further supports the assumption that a cell width of 0.1 m in row direction produces more accurate results.

Table 3-3. Results of the telescopically refined cases

Case nr	Cell size (m)		Number of cells	Runtime (min)	Final fresh water volume (m^3)	Solute mass balance error (%)	Volume balance error (%)
	Near well	Base grid					
35	1 x 10	20 x 50	77112	11	735741	1.79	0.01
36	1 x 5	20 x 50	102816	24	735734	1.76	-0.01
37	1 x 1	20 x 50	239904	164	735682	1.74	-0.02
38	0.1 x 0.1	20 x 50	1985088	103765	760176	0.85	-0.25
39	1 x 5	10 x 25	217560	41	760195	1.42	-0.02
40	1 x 5	5 x 20	413280	187	765712	1.09	-0.05
41	1 x 5	5 x 10	705600	379	769261	0.81	-0.06

The four cases that are refined near the well show a strong increase in runtimes when the cell size decreases. This relation is not linear, as was the case for the vertically unstructured cases (see section 3.2). This is also shown in Figure 3-19 in section 3.3.3: the cases follow a straight line when plotted on a log-log scale. This can be explained by the fact that a decrease in cell size close to the well will strongly decrease the timestep length due to the stability criteria (see also section 1.3). This effect is strong near the well, since the flow velocities will be high at this location: both the Neumann and the Courant criterion limit the timestep length at locations with high velocities and small cell dimensions.

The effect of changing the base mesh has a similar impact on the results as was found for changing the cell size of the fully regular cases. A decrease in base grid cell size results in an increase in fresh water volume. Similar to the fully regular cases, this increase can be due to the increase in variable head cells. The concentrations of case 39-41 are similar, but slightly higher compared to case 36. This is remarkable since the cell size at the well is the same. A possible explanation may be that the refining the cell size at the drain means that the upconing of saline water is simulated more accurate, which can be seen in the chloride concentration during the last timestep (see Figure 3-18). The total opconing at the drain is slightly higher for a more refined grid, which could result in higher upconing close to the extraction well. However, testing the effect of cell sizes at the drain is beyond the scope of this research, so more cases should be run to further test this hypothesis.

The solute mass balance error decreases for a decrease in base grid cell size, which suggests that the finer base grid produces more accurate results. The difference in results of cases 36 and 39 – 41, together with the fully regular cases of the previous section, indicate that the original base mesh of 20x50 m that is implemented in the reference cases in SEAWAT and MODFLOW 6 (R-1 – R-4) is too coarse.

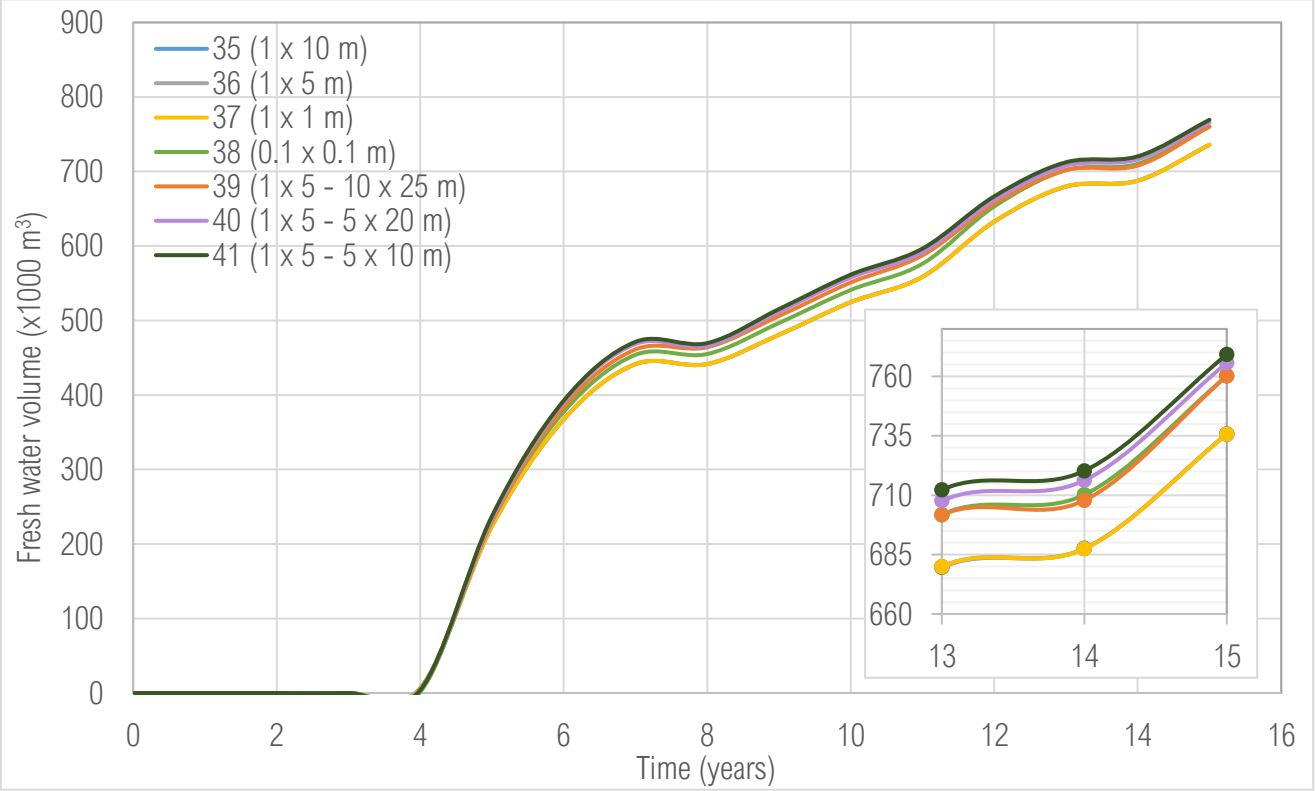

Figure 3-15. Fresh water volume (x1000 m³). The numbers in the brackets are the cell sizes at the well. The cell size far away from the well is 20 x 50 m, except if specified otherwise (with the second cell size). The graphs of 35 and 36 are the same as 37. The close-up shows the last two years.

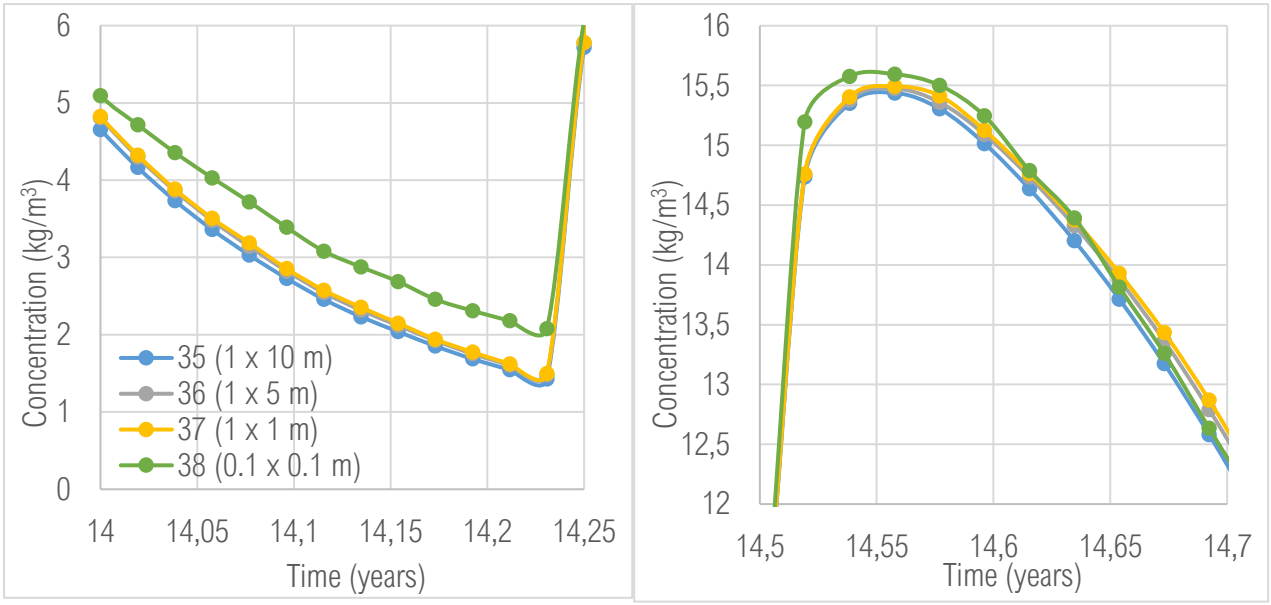

Figure 3-16. Close up of the chloride concentration (kg/m³) at the well at the lowest and highest points during the last simulation year. The graph of 36 in the left plot overlaps with 37.

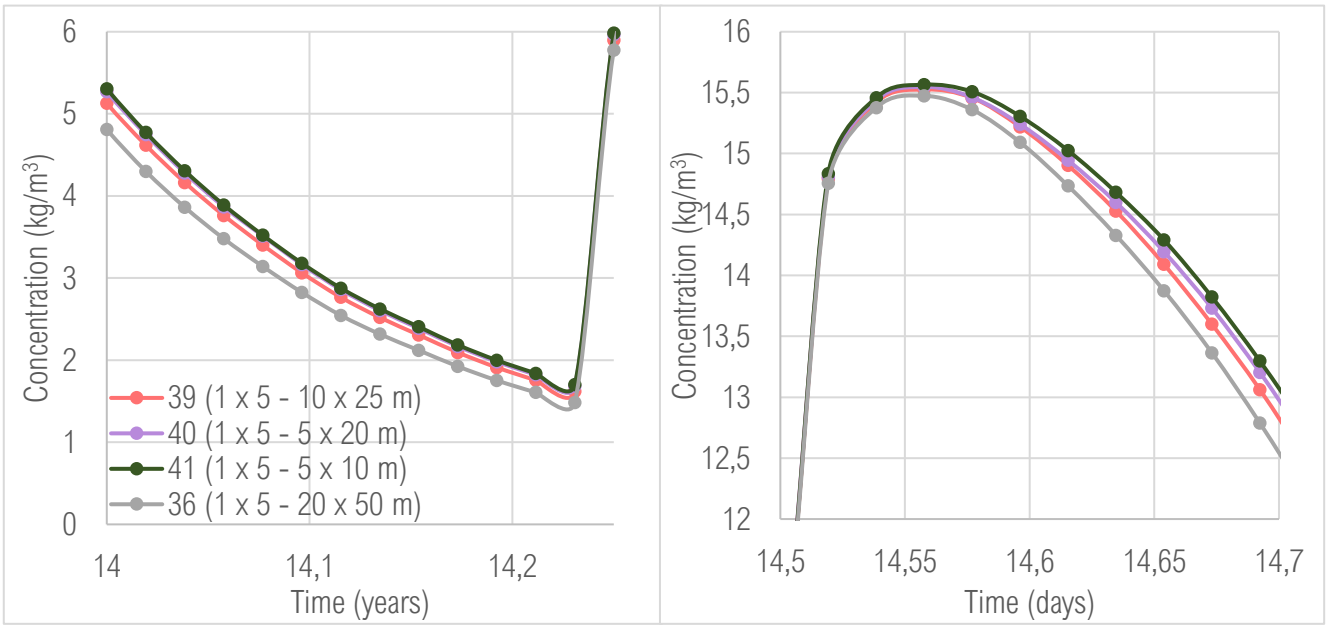

Figure 3-17. Close up of the chloride concentration (kg/m³) at the well at the lowest and highest points during the last simulation year. The graph of 40 in the left plot is the same as 41.

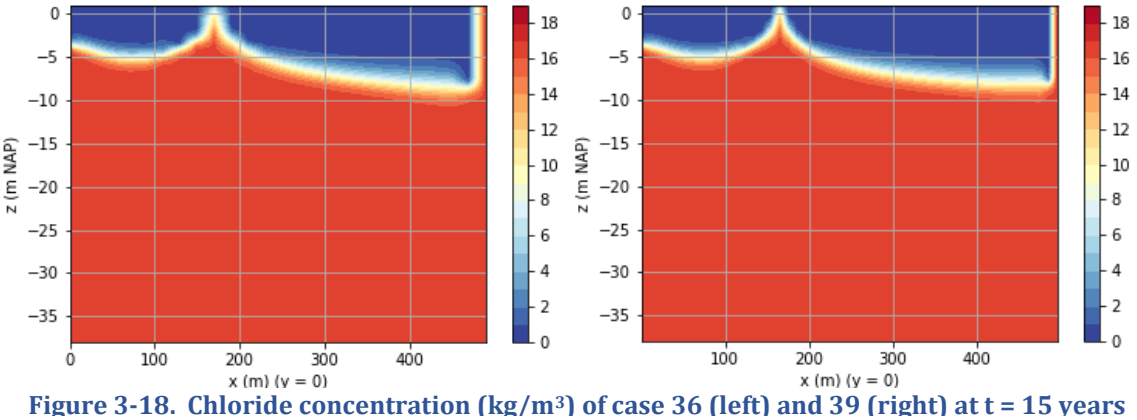

Figure 3-18. Chloride concentration (kg/m³) of case 36 (left) and 39 (right) at t = 15 years

3.3.3 Case 30 – 41: Runtimes

Figure 3-19 shows the runtimes of all structured cases. The cases have been divided in three separate groups (fully regular, refined near the well, refined base mesh). For all groups, an increase in number of cells will result in an increase in runtime. However, the figure suggests that this increase in runtime is lower for the fully regular cases and higher when the number of cells increases due to refinement near the well. This difference becomes very clear when comparing fully regular case 32 (10x10 m) with telescopically refined case 37 (1x1 m near the well). Case 37 has slightly less cells compared to case 32, but the runtime is almost four times higher. The data points of the cases with a refined base mesh show an increase in runtime that is between the other two groups.

This difference can be explained by the stability criteria that are generally applied in solute transport models (also see section 1.3). The timestep of the simulation is determined by the cell where these criteria result in the highest value. Both the Neumann and the Courant criterion are dependent on flow velocity and cell dimension. If the cell size decreases at a location with high velocities (like close to the extraction well), the model timestep should be decreased to meet these criteria. The smaller timesteps will increase the total simulation time.

Cases 39 – 41, with a refined base mesh, indicate that the cell size close to the well is not the only factor that influences the runtime. An increase in number of cells of these cases also increases the runtime, while the cell size at the well is the same. This can have two causes. First, at moments where the well is not extracting water, there may be a different location with relatively high flow velocities (for example at the drain), so a refinement at those locations could also cause the timestep size to decrease due to the stability constraints. Second, an increase in number of cells means that there are more nodes (cell centers) where the groundwater flow and transport has to be solved, which will increase the computational demand and hence increases the runtime.

Unfortunately, there is no documentation (yet) on the exact implementation of stability criteria in solving the transport equation of MODFLOW 6. Therefore, it cannot (yet) be determined for sure that the stability criteria in MODFLOW 6 are implemented similarly as MT3DMS and SEAWAT. Moreover, it is not clear if MODFLOW 6 uses transport steps (as in SEAWAT and MT3DMS); these steps are not printed to the list file. Therefore it cannot be said for certain that there are certain cells close to the well, drain or other locations that influence the timestep size.

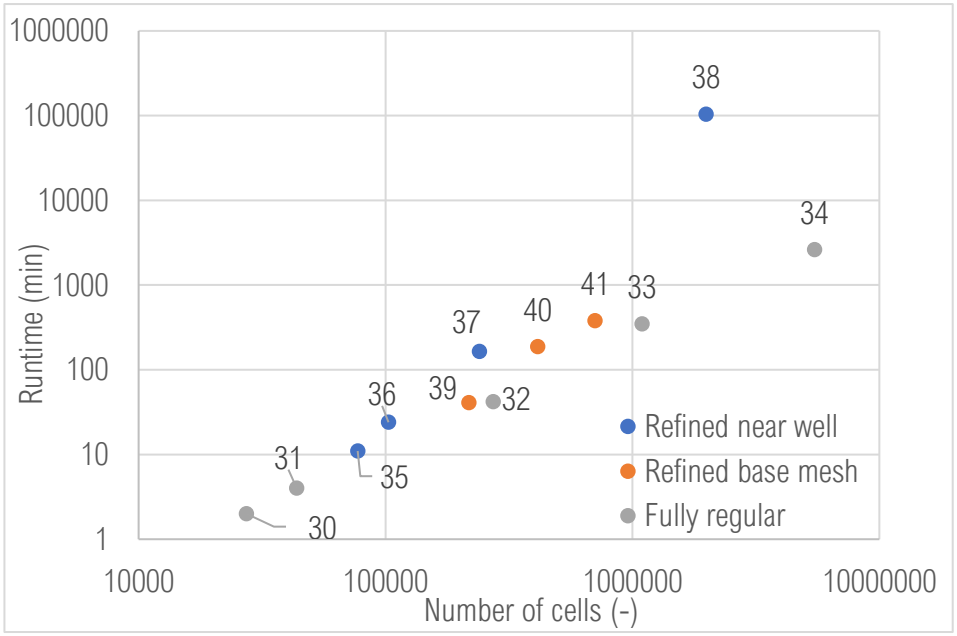

Figure 3-19. Runtimes (minutes) of all cases with structured grids. The data labels show the case number. Note the log-log scale.

3.3.4 Case 42 – 45: Horizontal unstructured

In this subsection, the results of four cases that are horizontally unstructured area shown. Cases 42, 43 and 44 have a 'telescopic counterpart' (cases 36, 39 and 37 respectively), meaning that they have very similar grids with regard to cell size at the well and in the base grid, but they have no unnecessary fine cells at locations far from the extraction well. This significantly reduces the total number of cells. Case 45 is different: in this case the reference grid (of SEAWAT, case R-4 or case 35) is further refined close to the extraction well, instead of removing the unnecessary fine cells of case 38. This is also explained in section 2.5.2. All grids that are discussed here are shown in Appendix F (including case 45).

The results are listed in Table 3-4. The chloride concentration at the well is shown for case 35, 38 and 45 in Figure 3-20. The other non-telescopic cases follow a similar plot as their telescopic counterpart, which is almost equal to the plot of case 35 in Figure 3-20. Together with the fact that the final fresh water volume of these cases is equal, it shows that a non-telescopic grid (cases 42, 43 and 44) will produce equal results as an equivalent telescopic grid.

The only exception is case 45 (cell size 0.125 x 0.156 m at the well). This case has a lower final fresh water volume compared to case 38, but remarkably this volume is equal to the other non-telescopic cases with the same base grid (36, 42, 37 and 44). Moreover, the chloride concentration in the well of case 45 is slightly different compared to the other cases, which is shown in Figure 3-20. The difference in concentration between case 38 and 45 can be up to 1 kg/m^3. The differences between the two cases can be explained by the fact that case 45 is much less refined compared to case 38. Both cases have approximately the same cell size at extraction well, but the cell sizes of case 45 increase fast when moving away from the well. For example, at x = 1.875 and y = 0 m the cell size of case 45 is 0.5x2.64 m, whereas the cell size of case 38 is 2x0.1 m. Furthermore, the other non-telescopic grids have the finest discretization over an area of 23 x 40 m, in case 38 the cell size is back at 1x10 m outside of an area of 10x60 m. Case 45 is more similar to case 35 than to case 38. That means that the region of refinement is too small in case 45 to improve the accuracy of the results like observed in case 38.

The runtimes are plotted in Figure 3-21. For clarity case 35 is also included, even though this case does not have a non-telescopic equivalent. The horizontally unstructured cases are generally faster, which is of course due to the decrease in number of cells. Case 43 has a slightly higher runtime compared to case 39, which is strange since the number of cells is considerably lower. It is unclear what caused these cases to be equally fast, but it may be possible that other processes were running on the computer during the simulation of case 43, which caused an increase in runtime. Both cases should be run again to know for sure if this was the case.

Table 3-4. Comparison of results of telescopic and non-telescopic cases with the same discretization near the well and in the base grid. Case 35 has no non-telescopic counterpart, but is shown as comparison. * = average runtime of two runs

Case nr	Cell size (m)		Number of cells	Runtime (min)	Final fresh water volume (m^3)	Solute mass balance error (%)	Volume balance error (%)
	Near well	Base grid					
35	1 x 10	20 x 50	77112	11	735741	1.79	0.01
36	1 x 5	20 x 50	102816	24	735734	1.76	-0.01
42			68544	11*	735699	1.75	-0.01
37	1 x 1	20 x 50	239904	164	735682	1.74	-0.02
44			144648	95*	735693	1.75	-0.01
38	0.1 x 0.1	20 x 50	1985088	103765	760176	0.85	-0.25
45	0.125 x 0.156		251328	214	735574	1.74	-0.01
39	1 x 5	10 x 25	217560	41	760195	1.42	-0.02
43			153216	43	760263	1.47	-0.02

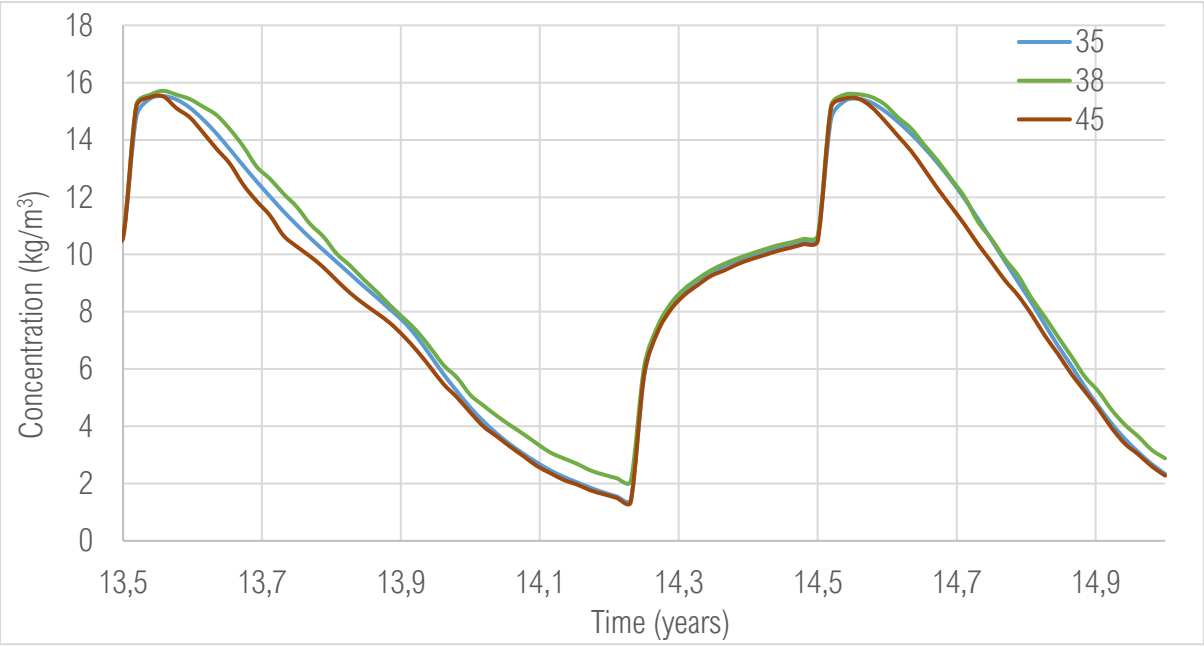

Figure 3-20. Chloride concentration (kg/m³) at the well during the last two years for two structured cases (35 & 38) and one horizontally unstructured case (45).

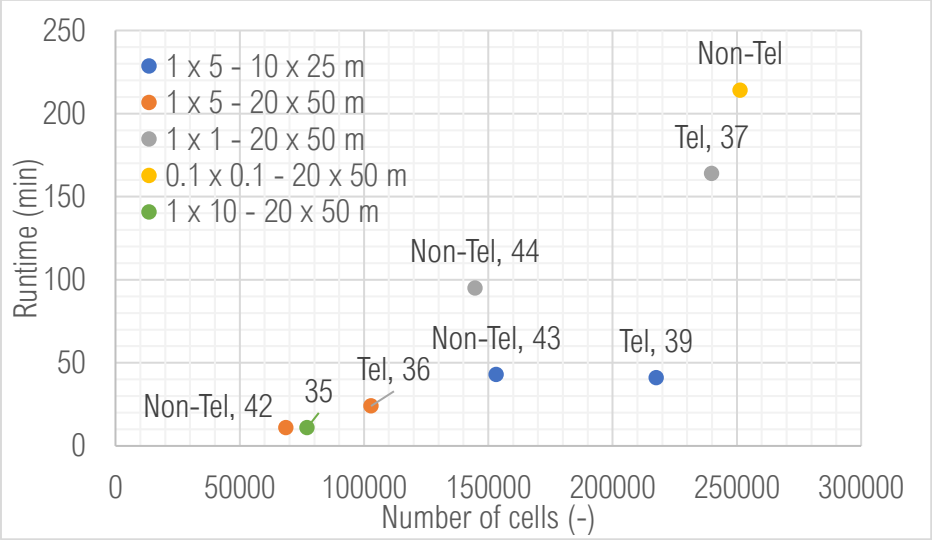

Figure 3-21. Runtimes of cases 35 – 39 and 42 – 45. The legend shows the cell size near the well and the cell size of the base grid. The labels indicate the case number and whether the case is telescopic (Tel) or non-telescopic (Non-Tel). Case 35 is shown but has no non-telescopic counterpart. Case 38 is not shown.

3.3.5 Case 46 – 50: Horizontally and vertically unstructured grids

In this subsection, the cases 46 to 50 will be discussed, which use case 37 (telescopic, 1x1 m near the well) as reference. Together with case 44 (non-telescopic, 1x1 m near the well), these cases show if all the different ways to coarsen a grid, that were discussed in the previous sections, can be combined to reduce the runtime. Eventually, this runtime will be compared to cases that are less fine near the extraction well, to see if unstructured grids can be used to keep the runtime low if the grid has a region with very fine cells. Cases 46, 47 and 48 are vertically unstructured cases that use the telescopic grid of case 37 as fine grid in the top layers of the model; cases 49 and 50 use the non-telescopic grid of case 44 as fine grid. The results are shown in Table 3-5. The evolution of the fresh water volume and the concentration at the well is approximately the same for all cases, so these plots are not shown.

The runtimes are plotted in Figure 3-22, together with the telescopic cases and non-telescopic case 44. Similar to the earlier discussed cases, the runtime decreases significantly for a decrease in number of cells. When comparing the data points of case 44, 47 and 48, it becomes clear that implementing a vertically unstructured grid (orange data points) show a faster decrease in runtime than implementing only a vertically unstructured grid (blue data point). This can be explained by the fact that a horizontal unstructured (non-telescopic grid) still contains very fine cells (of 1x1 m) over the full depth of the model domain, whereas in a vertical unstructured grid the cell size below the extraction well increases for every new coarse discretization. Obviously, implementing a non-telescopic grid that is also vertically coarsened will decrease the runtime even more, which ultimately will be necessary to keep the runtimes of a refined grid low.

The case that is coarsened the most (case 50) has a runtime that is close to the structured, telescopic case that has a cell size of 1x5 m near the well (case 36). This fact shows that implementing an unstructured grid where cells are coarsened at relatively unimportant locations could be used to keep the runtime as low as a structured case with a less fine grid at the extraction well.

Table 3-5. Results of horizontally and vertically unstructured cases (case 46 - 50). Case 37 and 44 are shown for comparison. The runtimes are the average of two separate runs (except case 37).

Case nr	Number of cells	Runtime (min)	Final fresh water volume (m^3)	Solute mass balance error (%)	Volume balance error (%)
37	239904	164	735682	1.74	-0.02
46	204624	118	735557	1.72	-0.02
47	185199	97	735756	1.73	-0.02
48	172179	88	735641	1.75	-0.01
44	144648	95	735693	1.75	-0.01
49	117783	64	735617	1.76	-0.01
50	108058	33	735853	1.76	-0.01

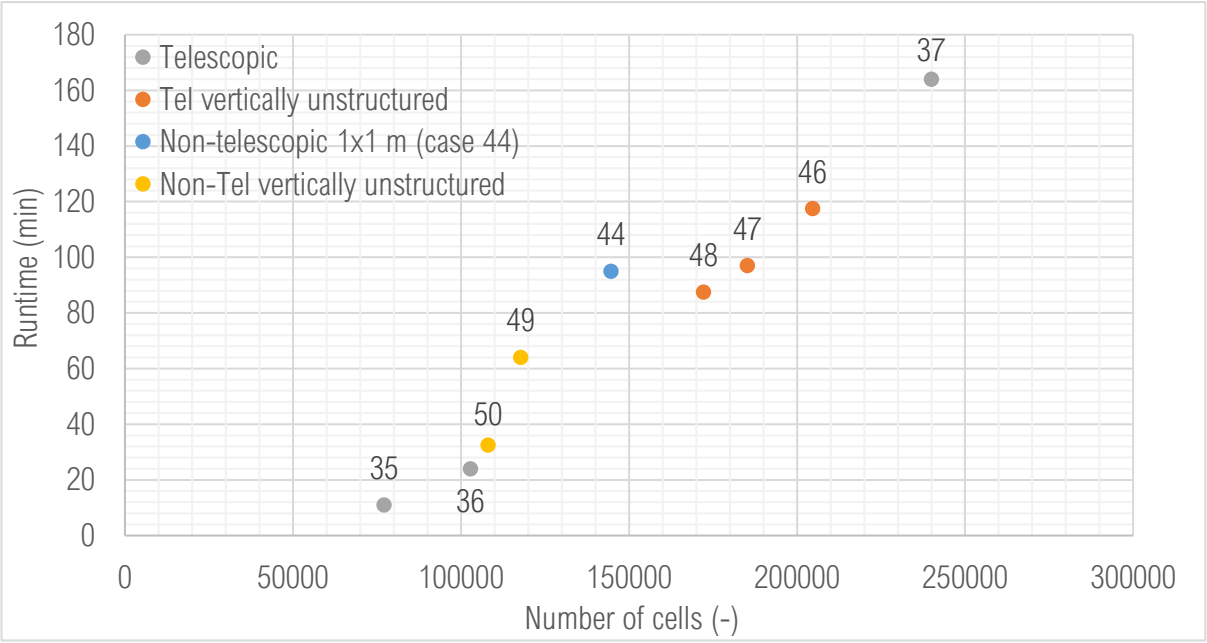

Figure 3-22. Runtimes (minutes) of horizontally and vertically unstructured cases. The data labels show the case number, the color is the model type. The runtimes of the telescopic cases (35 – 37) are shown for comparison.

3.4 Temporal discretization

The number of timesteps within one stress period is varied to determine the influence of the timestep length on the model outcome. The original model has a timestep length of one week, which is one timestep per stress period (NSTP = 1). The results are listed in Table 3-6.

The number of timesteps within one stress period is varied to determine the influence of the timestep length on the model outcome. The original model has a timestep length of one week, which is one timestep per stress period (NSTP = 1). The results are listed in Table 3-6. and Figure 3-24 show that the total incoming and outgoing mass and volume increases when the timestep length decreases: for example the incoming solute mass increases with 1.6% between nstp_1 and nstp_14. The fresh water volume in the system for the last 5 years is shown in Figure 3-25. The fresh water volume of nstp_1 (Δt = 7 days) is consequently lower than the volume for the other cases, but this difference remains small: the final volume of nstp_1 is 0.4% lower than nstp_14. Moreover, every increase in number of timesteps results in a higher fresh water volume in the system, but the difference between cases is getting smaller for every increase. The solute mass balance decreases from 2.5 to about 2.1 %, which suggests that the results become slightly more accurate. The differences in chloride concentration in the well (see Figure 3-26) are slightly more pronounced. Nstp_1 deviates the most from the other runs, the difference with nstp_14 can be more than 0.5 kg Cl^-/m^3. Increasing the number of timesteps yields a concentration plot that converges to one solution.

Table 3-6. Overview of results of cases with varying timestep length

NSTP	Timestep length (days)	Runtime (min)	Final fresh water volume (m^3)	Solute mass balance error (%)	Volume balance error (%)
1	7	27	1233493	2.5	0.01
2	3.5	41	1235980	2.28	0.02
3	2.33	54	1237130	2.24	0.02
4	1.75	66	1237110	2.18	0.02
7	1	109	1237422	2.11	0.01
10	0.7	151	1238073	2.14	0.01
14	0.5	177	1238475	2.17	0.02

The hydraulic heads in the domain are also influenced slightly by the temporal discretization. A comparison between the hydraulic head distribution at t = 39 years for cases nstp_1, nstp_7 and nstp_14 is shown in Figure 3-27. As can be seen, the hydraulic heads increase slightly when the timestep length decreases. A decrease in timestep length will mean that there are more moments at which the flow equation is updated with a new concentration and density distribution. As a result, the velocities are adjusted sooner, which ultimately may lead to an earlier increase in volume fluxes from the sources and sinks. The runtimes are plotted in Figure 3-28. The runtime increases when the number of timesteps per stress period increases, but this increase is much larger for a high number of timesteps per stress period (small timesteps).

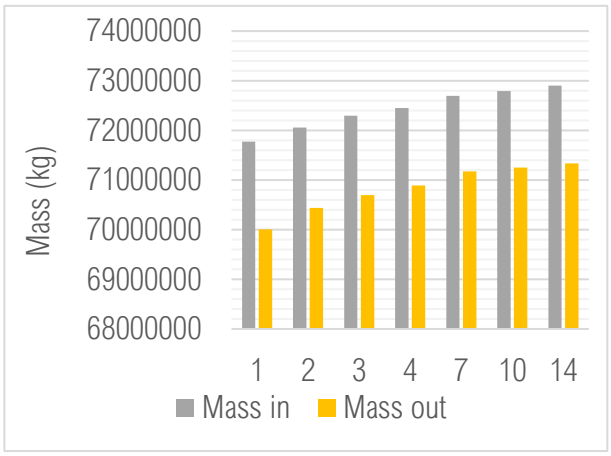

Figure 3-23. Final solute mass balance (kg) for all NSTP values. Note the scale of the y-axis.

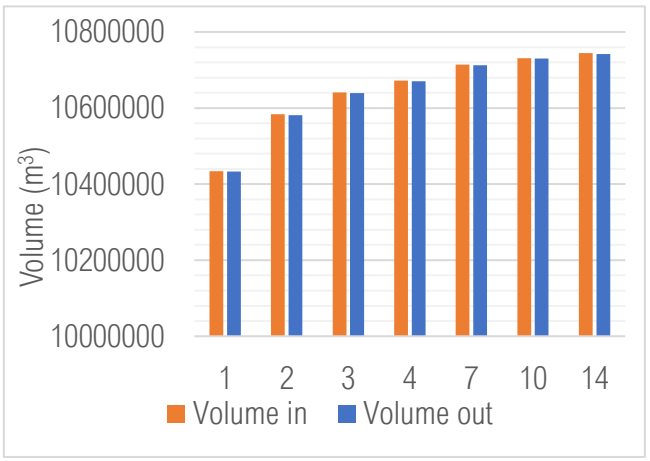

Figure 3-24. Final water volume balance (m3) for all NSTP values. Note the scale of the y-axis.

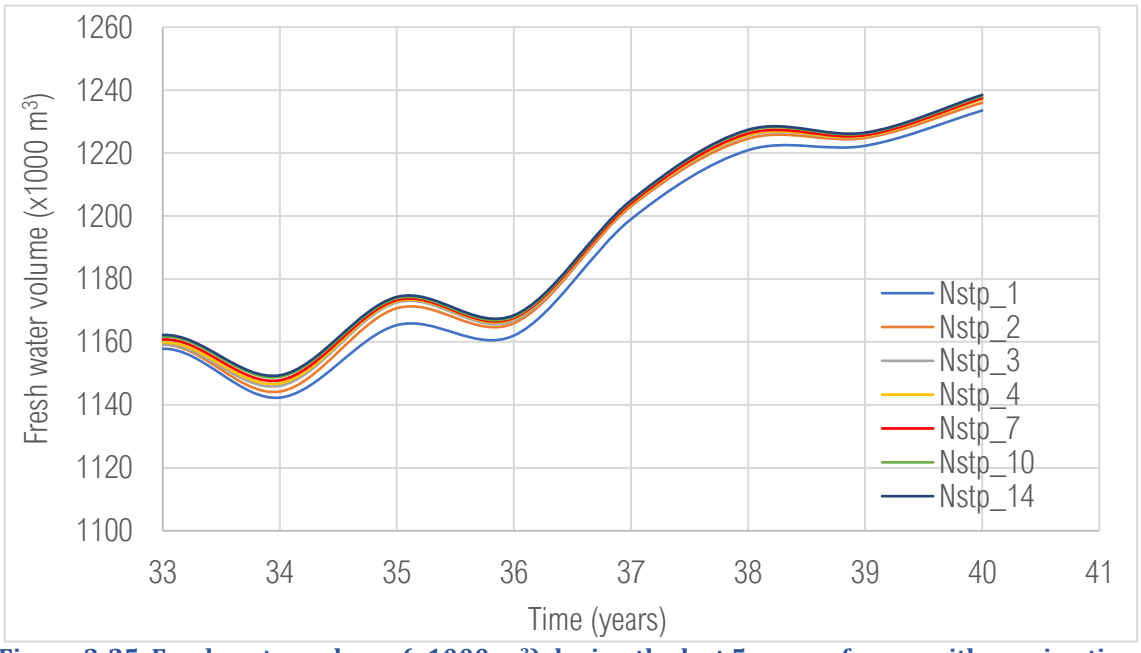

Figure 3-25. Fresh water volume (x1000 m³) during the last 5 years of cases with varying time step length

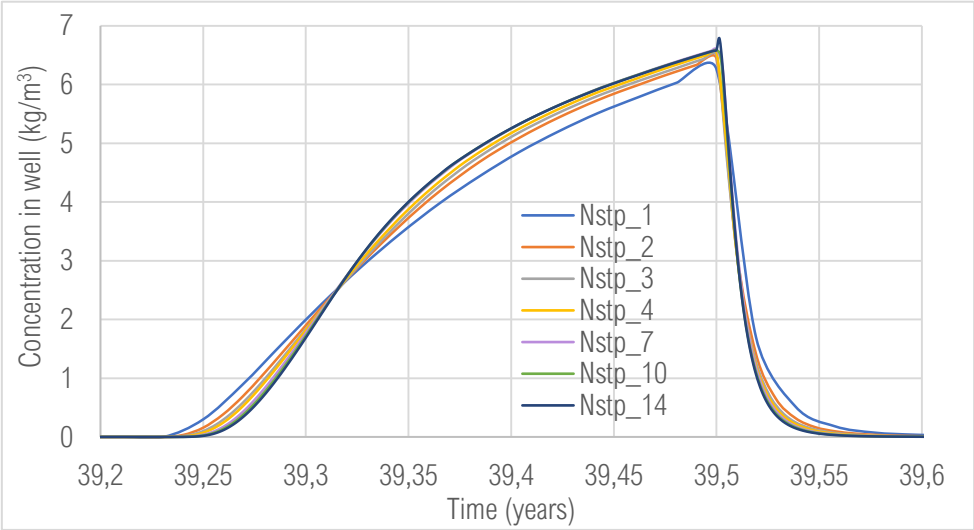

Figure 3-26. Concentration in the well (kg Cl^-/m^3) during the last timesteps

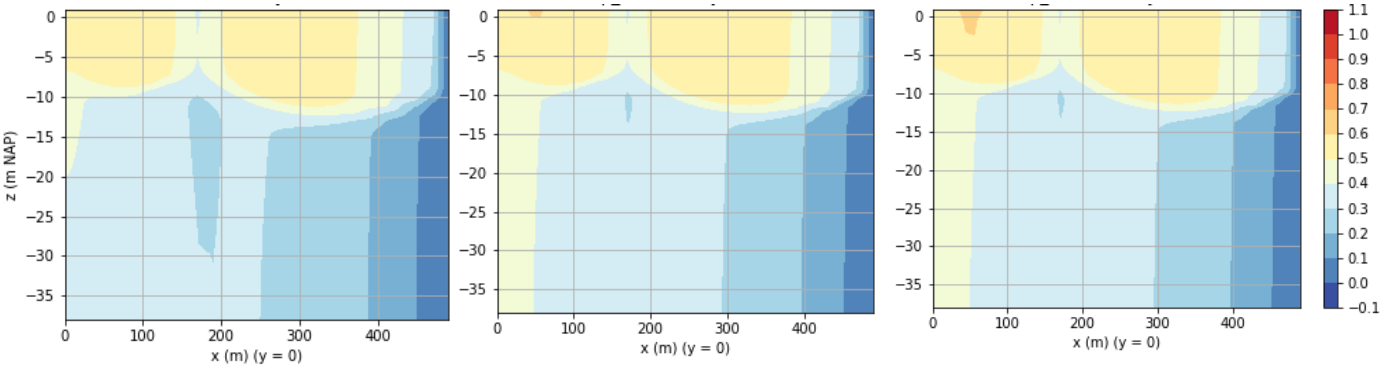

Figure 3-27. Hydraulic head (m) distribution at t = 39 years. From left to right: nstp_1, nstp_7, nstp_14.

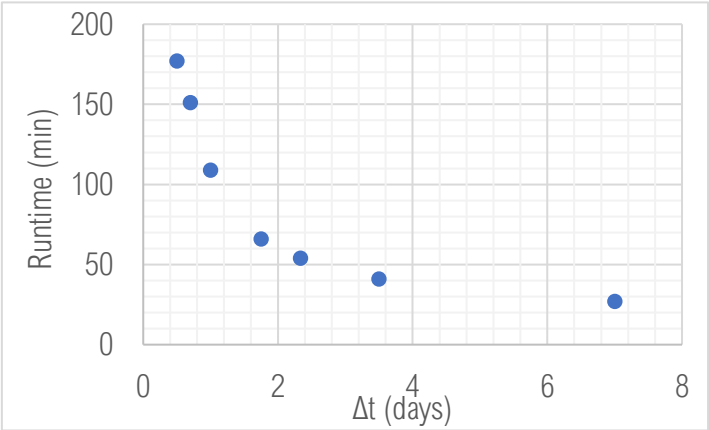

Figure 3-28. Runtimes (minutes) of model runs with varying timestep length (Δt)

4 Discussion

In this chapter, the results of the spatial and temporal discretization tests that are done in MODFLOW 6 will be discussed. The results and discussion of this study are used to get a first impression of the possible (dis)advantages of MODFLOW 6. It should be stressed that only one specific model is used for all these tests, so the results that are presented here can not be applied generally to all variable density groundwater models in MODFLOW 6.

4.1 SEAWAT and MODFLOW 6

Section 3.1 compares the results of the same model that was run with SEAWAT and MODFLOW 6 and it was shown that the results are not the same. The volume of fresh water according to MODFLOW 6 is constantly higher than the volume in the SEAWAT model. This difference is the largest at the beginning of the simulation, but after 40 years the MODFLOW 6 model has a fresh water volume that is still 13% higher compared to the volume in SEAWAT. The plot of the concentration in the well shows that during the first 15 years, SEAWAT over- and undershoots the concentration. When the solution stabilizes, the concentrations are slightly lower than the concentrations in MODFLOW 6. Despite this over- and undershooting, the mass balance error of SEAWAT is 0.0 %, whereas MODFLOW 6 has an error of 2.5 % after 40 years. This error of MODFLOW 6 is remarkably high, especially since the TVD solver should be mass conservative (Zheng and Wang, 1999).

These results show that the model output is influenced by the choice of model software and the solvers and that one should be careful when switching from SEAWAT to MODFLOW 6. Both models use the TVD solver for the advection part of the solute transport equation, so it is remarkable that there are still considerable differences in the results of the models. The implementation of the TVD method could different in both models, but due to the lack of documentation on the transport part of MODFLOW 6 it cannot be said for certain that both models implement the TVD solver differently. Furthermore, other settings like the iteration criteria have also influenced the results. Both the study on the effect of solute transport solvers and iteration criteria is beyond the scope of this research, but it is clear that more research should be done to better understand the effect of solver and iteration settings of MODFLOW 6.

Nevertheless, the results of SEAWAT and MODFLOW 6 are still comparable, meaning that MODFLOW 6 is able to produce appropriate results and that the differences may be attributed to differences in solution techniques and iteration criteria of both models. The fact that MODFLOW 6 does not over- or undershoots the concentrations at the well and is almost twice as fast as SEAWAT (while the number of cells are equal), indicates that MODFLOW 6 is a suitable option for the modelling of variable density groundwater flow and coupled solute transport.

4.2 Spatial discretization

4.2.1 Fully regular grids

One of the aims of this study is to define the effect of implementing coarse, unrefined grids, when implemented in this variable density groundwater model with various hydro(geo)logical stresses. As expected, the results are largely influenced by refining the grids. The cases with cell sizes of 25x25 m and 20x50 m were fast, but inaccurate. The other cases showed a significant increase in runtime, but also the solution improved. The most refined case (case 34, 1x5 m) had a more diffuse transition from fresh to saline groundwater and a lower final volume of fresh water, which may have been caused by a too small region of constant boundary conditions. These three cases show that there will be a choice whether a model should be fast or accurate. Obviously, one of the options to reach a fast, but accurate, solution is to refine the grid only at points of interest, which was the aim of the other cases and will be discussed in the next subsections.

4.2.2 Vertically unstructured grids

The vertically unstructured cases show that it is possible to implement a coarser grid in the lower layers of this model without altering the results. However, there is a limit on the extent to which this coarsening can be carried out, since some cases did produce results that are deviating from the MODFLOW 6 reference case (case R-4). The interface of the fresh-saline groundwater interface is very important for this aspect: when the model is too coarse at the interface between fresh and saline groundwater, the results will change.

The cases that showed a deviation in almost every aspect were all cases from group D: the cases that contain a coarse layer that is coarsened over the full width of the model. This is because the maximum cell sizes in these cases were large (40x50 m or 20x100 m). Better results are shown for the cases that are only coarsened in the telescopic part of the domain. These cases were subdivided into three groups (A, B, E), where the coarsening was in row-, column direction or both. Coarsened layers with wider cells in row direction could be implemented at a higher elevation (less deep) compared to layers with wider cells in column direction. This is because the grids with wider cells in column direction for this model result in a larger displacement of the cell centers. This means that the vertical connection between cells at the transition to the coarse layer are intersecting their connecting face at a higher angle and further from the middle. This is incorrect with respect to the requirements for a 'good' grid (Narasimhan and Witherspoon, 1976; Panday et al., 2013). However, these requirements do not necessarily mean that the model results will be incorrect; the cases in this study have shown that there is no issue when the transition to a coarse layer is far away from the fresh-saline groundwater interface.

A coarse layer that is coarsened in both directions in the telescopic part of the reference case is also implemented in several cases. This discretization occurs in group E, where the transition goes immediately from the fine reference discretization to this coarser layer, and in group C, where this discretization is implemented as an extra coarsening step below an intermediate coarse layer. The cases of group E contain grids where coarse cells are overlain by four fine cells, which is usually better to avoid when creating a grid (Panday et al., 2013). However, the cases of group E have shown that this would not per definition result in inaccurate results, as long as the transition from a fine to coarse discretization is at a distance from important areas, like the fresh-saline interface or the extraction well. The cases in group C show that this coarse discretization can be at higher elevations, but only if there is an intermediate coarse layer in between. This is because in the case of an intermediate coarse layer, all cells are vertically connected to at maximum two cells.

The runtimes of all cases are plotted against the number of cells (see Figure 3-9). This showed that there is a linear correlation between the number of cells and the runtime for the vertical coarsening. According to these tests, a decrease of 10% in number of cells will decrease the runtime with about 16.5%. However, the range in runtime and number of cells in these cases is relatively small: the difference between the fastest and slowest run is only 11 minutes. This relation between runtime and number of cells may therefore change if the cell number or total duration of the model increases. Moreover, not all data points are exactly fitted on the linear trendline in Figure 3-9, which shows that the runtime does not exactly follow this linear relation.

4.2.3 Telescopic refinement

The results of the telescopically refined grids are shown in section 3.3.2. Cases 35 to 38 are refined telescopically near the well and have a base grid cell size of 20x50, cases 39 – 41 have a refined base grid and have a cell size of 1x5 m at the well. Case 38 has a cell size of 0.1x0.1 m at the well, which was the only refinement at the well in both row and column direction. The results of this case were significantly different, meaning that using very fine cells close to the well and refining in the row direction may be necessary to produce accurate results. However, since this was the only case with a high degree of telescopically refinement in row direction, more cases should be run with different cell widths in row

direction (e.g. 0.5 m and 0.2 m) to further prove this. The telescopic cases with a cell size of 1x5 m at the well, but different base grid refinement had a similar effect on the results as was seen in the fully regular cases (see subsections 3.3.1 and 4.2.1). For example, the chloride concentration at the well increases for smaller a base grid cell size.

Increasing the base grid will also influence the runtimes of the model, so both the cell size as the number of cells is an important factor to consider when implementing a grid. However, it is not possible to determine what stability criteria are implemented to solve the transport equation in MODFLOW 6 and to determine the cell that may inhibit a fast simulation. Therefore, the exact influence of cell size and number of cells should be studied further when more information on the transport part of MODFLOW 6 is available.

The difference in relation between runtime and number of cells for cases with a refined base grid compared to cases with a refined telescopic area illustrates that the number of cells of a model will not be a direct measurement of the computational time.

4.2.4 Horizontal unstructured grids

In section 3.3.4, the results of telescopically refined grids were compared to similar horizontally unstructured grids without unnecessary refinement far away from the well. Almost all horizontally unstructured cases had the same results as their telescopic counterparts. The case with cell size 0.125x0.156 near the well (case 45) did not show the change in chloride concentration in the well and fresh water volume that was seen in the telescopic case with cell size 0.1x0.1m near the well. This indicates that only refining a very small part around a point of interest will not directly improve the results. However, the runtime of this case is the highest of all horizontally unstructured cases. This illustrates that implementing a fine grid around a point of interest could have a negative impact on the model outcome, since the results did not improve but the runtime increased significantly. More research should be done to determine a sensible size of the area of refinement.

Implementing a horizontally unstructured grid will significantly decrease the number of cells and runtime, which is shown in Figure 3-21. The only exceptions are the cases with a base grid of 10 x 25 m (cases 43 and 39), where the non-telescopic case with less cells reaches even a slightly larger runtime than its telescopic counterpart. This is unexpected, it may indicate that there were other processes running on the computer that slowed down the simulation. The cases should be rerun to check if this was the case.

When comparing the non-telescopic cases to the telescopic cases, it seems that the runtime increases faster for the same increase in number of cells. For example, case 37 and case 45 have approximately the same number of cells (the difference is only 11424 cells), but case 45 took 50 minutes longer. This is because the cell size near the well of case 37 is 1x1 m, whereas case 45 has a cell size of 0.125x0.126 m. An increase in number of cells for structured (telescopic) cases means that the cell sizes near the well are slightly smaller, but there are also refined cells at relatively unimportant locations. An increase in number of cells in a horizontally unstructured case means that the cell size near the well is decreased significantly, since no extra fine cells are ‘created’ at other locations in the grid. Therefore, the same number of cells for a structured grid and horizontally unstructured grid could only occur if the cell size near the well is smaller. Due to the stability criteria and the relatively high flow velocities near the well, this small cell size will substantially decrease the computational speed. This effect illustrates that the number of cells alone is not a good indicator for the total runtime of the model, but that the smallest cell size near a location with high flow velocities also has a big impact on the runtime.

4.2.5 Horizontal and vertical unstructured grids

To answer the research question whether unstructured grids can be used to implement fine grids at points of interest without drastically increasing the runtime, five unstructured cases (46 – 50) are used (see subsection 3.3.5). These cases are all based on case 37, the structured telescopic case with 1x1 m cells near the well and 20x50 m in the base grid. As was shown in subsection 4.2.3, telescopically refining a grid

significantly increases the runtime. With previous versions of MODFLOW, this was important to consider when building a model grid. Unstructured grids may be the solution to still refine the grid at points of interests, but without significantly increasing the runtime. Therefore, the unstructured cases that will be discussed here all contain 1x1 m cells near the extraction well but have larger cell sizes far away from the well compared to the telescopic case 37.

The runtimes are visualized in Figure 3-22. As was already discussed in the previous sections, the options to refine non-telescopically or to implement coarser grids in lower layers will have a positive impact on the runtime of a model. Cases 46 to 50 further prove this point and show that these measures can also be combined to even further reduce the runtime without altering the results. Besides, these cases have shown that implementing a vertically coarsened grid will have a larger influence on the runtime compared to only implementing a non-telescopic grid, even if the number of cells is lower of the non-telescopic case. This is because vertically coarse grids contain less very fine cells close to the extraction well.

Ultimately, the runtime is reduced from 164 minutes (case 37) to 32.5 minutes (case 50), which is close to the runtime of case 36: a telescopic grid with cell size 1x5 m near the well. It can therefore be concluded that due to the possibility of implementing unstructured grids, MODFLOW-6 can be used to run models with a higher degree of refinement near points of interest compared to SEAWAT.

However, one should be careful with implementing very fine cell sizes without considering the effect on the runtime. As was shown in the previous sections, increasing the number of cells of non-telescopic grids will increase the runtime slightly faster compared to increasing the number of cells in telescopic grids. This means that when a MODFLOW 6 user inserts an extremely fine grid at locations with high flow velocities, like at an extraction well, the runtime could still increase significantly. In that case, a lot of coarsening at other locations is needed to keep the runtimes low enough; it might even be not enough to fully compensate the loss in runtime. However, if the cells at certain locations become too large, there can be a considerable impact on the results. This was shown earlier for example with cases 36 and 39 to 41, where the cell size in the base grid was changed and a change in model results and solute mass balance error was observed. The vertically unstructured cases (case 1-29) also showed that if the coarser discretization is too close to the surface (and therefore close to locations with higher flow velocities and concentration gradients) the results can be become less accurate. This shows that there is a limit on the extent of which coarser cells can be implemented.

To conclude, using unstructured grids can be the solution to keep the model runtimes low for refined cases, but one should be careful that the cell sizes become so small that the runtimes still ‘explode’ and that the subsequent coarsening far away from this point of interest will negatively influence the results. To further research this, the same study should be repeated with cases that have finer cells than 1x1 m near the well, for example cells of 0.5x0.5 m or 0.1x0.1 m.

4.3 Temporal discretization

The length of the model timesteps was varied between 0.5 and 7 days. The results of these tests show that the model output will change when the number of timesteps changes, although these changes are relatively small. However, the runtime increases significantly for a decrease in timestep duration (see Figure 3-28). Both the volume of fresh water in the domain as the concentration of the water in the well show that the results converge when the number of timesteps per stress period increases. The results improve because the flow model and transport model are updated more frequently, and for example the flow velocity distribution will be a better at representing density differences.

Especially the original time step length of 7 days deviates from the rest of the simulations, which indicates that the original timestep length of the reference model is unsuitable to use for these simulations. The cases with the smallest timesteps, 0.5, 0.7 and 1 day, produce results that are very similar. Especially the

plot of the concentration in the well (Figure 3-26) show that these three plots overlay each other almost perfectly.

These results suggest that shorter timesteps produce better results, because the results are converging to a single solution. However, since the runtime increases significantly for the runs with short timesteps, it is best to use an optimal timestep that produces good results with a relatively short runtime. This means that for this simulation the optimal timestep length is 1 day, since the runtime is 1 hour less than nstp_10 and nstp_14, while the difference in results is insignificant.

4.4 Research limitations and future research

In this section, some shortcomings and possible improvements of this research will be discussed. First of all, in section 3.1 the difference in results due to the 'SAVE-SPECIFIC-DISCHARGE' option in the NPF package was shown. Although this option is only used to control the model output, it has an impact on the internal calculations. This demonstrates that the MODFLOW 6 transport model is still under development and there might still be other unresolved numerical problems that have influenced the results.

The grid study with fully regular grids (see section 4.2.1) has shown that cells with an area of 20x50 m are too coarse and influence the results. However, this cell size is still applied as coarse base grid in most cases and in the reference case in SEAWAT. Similarly, the temporal discretization tests have shown that the optimal timestep is one day, while the timestep in all other cases is one week. This timestep and the coarse base grid may have had an impact on the other conclusions from the spatial discretization tests.

Some cases (42, 44, 46, 47, 48, 49 and 50) have been rerun and had a different final runtime. The average runtime is reported here. However, the difference in runtime was significant in some cases. For example, case 49 took 59 and 69 minutes to run, a difference of almost 17%, whereas two runs of case 48 took 87 and 88 minutes; a difference of about 1%. These differences can be explained by the fact that the computational speed of a model may also be influenced by other processes on the computer. This impact is reduced by choosing to run all models on an external server, but this did not fully resolve this issue. This means that the possible impact of spatial or temporal discretization on the runtime may be slightly different than was reported here. However, all second runs of these cases have runtimes that are still in the same order as the first case (the largest difference is 17%), so the general conclusions will not be largely influenced by this.

Lastly, some possibilities to further continue this research will be discussed here. The differences between the results of the same model run in MODFLOW 6 and SEAWAT indicate that the models are different and probably solve the equations for variable density flow and coupled solute transport differently. It is important to determine what exactly caused these differences. It was beyond the scope of this research to look into the effect of solver settings, but the differences between these two models indicate that it should definitely be looked into more detail what these settings are and how they influence the results. The same goes for the Iterative Model Solution (IMS) settings of both the flow (GWF) and the transport (GWT) part of the model in MODFLOW 6.

As was explained in section 1.4, there are some grid characteristics that may indicate whether a grid is 'good', i.e. there will be no numerical instabilities due to the spatial discretization. It was also discussed that the types of unstructured grids that are used for this study do not meet these requirements. To diminish this negative influence of the grid, MODFLOW 6 contains a Ghost Node Correction (GNC) package. It would be interesting to see whether this package could be implemented to improve the model results.

The results have indicated that a very small cell size near the well (case 38, 0.1 x 0.1 m) could influence the results. Most cases only implemented a change in cell size in the column direction (e.g. cases 36, 1x5 m and 37, 1x1 m), whereas this is the only case that also has a refinement in the row direction. More cases should be added to better understand the effect of refining in the row direction.

The effect of spatial discretization is tested with one specific model, which limits the general applicability of the results of this study. For example, the reference model in SEAWAT was already vertically coarsened in the sense that the lower layers have a larger thickness compared to the layers close to the well and the fresh-saline groundwater interface. The layer thicknesses remained the same for all cases in this study. To better understand the possibilities and effects of implementing vertically coarsened grids, the same study (as with cases 1 – 29) could be carried out with models that are not yet vertically coarsened, i.e. models with a small layer thickness in lower layers.

Furthermore, tests could be done with models with a different dimension. Probably, models with much larger dimensions than the model of this study will benefit more from implementing unstructured grids, and a user should be even more careful when such grid is refined. Another option is to focus on models with different hydro(geo)logical stresses. This study solely implemented refinement at the extraction well, but the effect of refinement around the drain instead of around the well is equally important.

5 Conclusion

In this study, the recently released version of MODFLOW (MODFLOW 6) is used to simulate a variable density groundwater model and coupled solute transport. For this, a yet unreleased transport model (that is also implemented in MODFLOW 6) is used. A major difference of MODFLOW 6 compared to earlier versions is the possibility of using unstructured grids. For this, a model that was originally created in SEAWAT is converted to MODFLOW 6. The results of both models were compared first. Next, the possibilities of changing the spatial discretization of MODFLOW 6 were studied. In this study, several cases have been created that use a regular-shaped unstructured grid, i.e. unstructured grids with regular hexahedra. Besides, the temporal discretization of the model was studied.

The spatial discretization tests have been subdivided into five different groups: fully regular grids (that consist of one cell size), telescopically refined grids (that are refined around the extractions well in the way that was originally done in earlier MODFLOW versions), vertically unstructured grids (where lower layers contain a telescopically refined grid that is coarser compared to the top layers), horizontally unstructured grids (where the grid is refined at the extraction well without any unnecessary fine cells) and grids that are both horizontally and vertically unstructured.

The comparison between SEAWAT and MODFLOW 6 showed that the results of the same model will change if different software is used. The exact method of solving the groundwater flow and solute transport equations will differ for both models and also the solver settings will not be exactly the same. However, it is clear that MODFLOW 6 is suitable for running variable density groundwater models. Moreover, MODFLOW 6 has a significantly lower computational time and shows less issues with over- and undershooting of the concentration. More research should be done to better understand why the results differ from the results of the same case that was run with SEAWAT.

There are three research questions regarding the spatial discretization of MODFLOW 6. The first is to determine whether a regular grid without refinement can produce accurate results with an acceptable runtime. The fully regular cases in this study have shown that implementing a relatively coarse grid in the whole domain will strongly influence the results. Decreasing the cell size produces better results with very low mass balance errors, but also sharply increases the runtime. This shows that refining a grid only locally (either telescopically or with an unstructured grid) is necessary to improve the results of models that have a high runtime.

The second research question concerns the effect of implementing a regular unstructured grid compared to using a structured grid. The tests with vertically unstructured grids have indicated that it is possible to use coarser grids at low layers without influencing the results and that this will positively influence the model runtime. It will be best to implement a coarser layer in such way that cells are vertically connected to at maximum two cells and to only coarsen cells in the telescopic part of the grid. Coarsening a telescopic grid over the full area will lead to very large cells in the base grid of the coarse layers, with negatively impacts the results. Moreover, the coarse layers need to be at a distance from areas with high flow velocities and/or concentration gradients. Besides, the comparison between structured and horizontally unstructured grids have showed that the number of cells and therefore the runtime could decrease significantly if a non-telescopic discretization is used instead of a telescopic discretization. This further emphasizes the fact that using unstructured grids in MODFLOW 6 could be a relatively easy way to improve the runtime of a variable density groundwater flow and transport model.

The last question regarding spatial discretization was to determine if unstructured grids can be used to refine around a point of interest, without drastically increasing the model runtime. The tests of this study have shown that refining the grid near the extraction well will strongly increase the model runtime. This is because stability criteria (like the Courant criterion) that are generally implemented in solute transport models will decrease the timestep size when the cell dimension is small at locations with high velocities.

The runtime also increases when the grid is refined at other locations, but at a lower rate. Therefore, the number of cells is not the only indication for the expected runtime of a model, but the cell sizes and the location of small cells is equally influential.

The increase in runtime due to the implementation of a fine grid can be partially undone by increasing the cell sizes at other, less relevant, locations. However, this should be done carefully. First of all because the results of this study have shown repetitively that too coarse cells can negatively influence the model results, hence there is a limit on the degree of coarsening that can be done to decrease the runtime. Second of all, implementing very fine cells at a location with high flow velocities will drastically increase the runtime. This means that, even though MODFLOW 6 can use unstructured grids, it remains equally important as for previous solute transport models to consider the effect of small cell sizes at locations with high velocities.

The last aim of this study was to determine the relation between the temporal discretization and the results and runtime of a variable density model. Decreasing the timestep length will decrease the model runtime but also slightly decrease the accuracy of the results, since the flow and transport part of the simulation are updated less frequently. However, the spatial discretization tests lead to assume that the transport model of MODFLOW 6 breaks down the timesteps into smaller steps, as was the case with MT3DMS and SEAWAT. More research should be done to further understand this mechanism in MODFLOW 6.

6 References

Amtec Engineering, Inc. (2003) Tecplot User's Manual. Version 10. Bellevue, Washington, U.S.A.

Baxter, G.P. and Wallace, C.C. (1916) Changes in volume upon solution in water of halogen salts of alkali metals: IX. American Chemical Society Journal, 38, p. 70-104.

Guo, W. and Langevin, C.D. (2002) User's Guide to SEAWAT: A Computer Program for Simulation of Three-Dimensional Variable-Density Ground-Water Flow. U.S. Geological Survey Techniques of Water-Resources Investigations 6-A7, 77 p.

Harbaugh, A.W. (2005) MODFLOW-2005, The U.S. Geological Survey modular groundwater model—the Groundwater Flow Process: U.S. Geological Survey Techniques and Methods 6-A16

Hughes, J.D., Langevin, C.D., and Banta, E.R. (2017) Documentation for the MODFLOW 6 framework: U.S. Geological Survey Techniques and Methods, book 6, chap. A57, 42 p.

Langevin, C.D., Hughes, J.D., Banta, E.R., Niswonger, R.G., Panday, S., and Provost, A.M. (2017) Documentation for the MODFLOW 6 Groundwater Flow Model: U.S. Geological Survey Techniques and Methods, book 6, chap. A55, 197 p.

Narasimhan, T.N., and Witherspoon, P.A. (1976) An integrated finite-difference method for analyzing fluid flow in porous media. Water Resources Research, v. 12.1, 7 p.

Oude Essink, G.H.P. (2000) Groundwater Modelling. Lecture notes. Institute of Earth Sciences, Utrecht University.

Oude Essink, G.H.P. (2001) Density Dependent Groundwater Flow: Salt Water Intrusion and Heat Transport. Lecture notes Hydrological Transport Processes/Groundwater Modelling II. Institute of Earth Sciences, Utrecht University.

Oude Essink, G.H.P., Pauw, P.S. (2018) Evaluatie en verdiepend onderzoek naar grondwateronttrekkingsregels in de provincie Zeeland. Deltares Rapport 1231011-001, 157p.

Panday, S., Langevin, C.D., Niswonger, R.G., Ibaraki, M., and Hughes, J.D. (2013) MODFLOW–USG version 1: An unstructured grid version of MODFLOW for simulating groundwater flow and tightly coupled processes using a control volume finite-difference formulation. U.S. Geological Survey Techniques and Methods, book 6, chap. A45, 66 p.

Provincie Zeeland. (2002). Samen omgaan met (grondwater). Grondwaterbeheersplan 2002-2007.

Scheldestromen, W. (2013). Beleidsnota grondwater Versie: 3.5.

Stuyfzand, P. J. (1993) Hydrochemistry and hydrology of the coastal dune area of the Western Netherlands. Vrije Universiteit Amsterdam.

Van Baaren E.S., Oude Essink G.H.P., Janssen G.M.G.M., de Louw P.G.B., Heerdink R., Goes B. 2016. Verzoeting en verzilting van het grondwater in de Provincie Zeeland; Regionaal 3D model voor zoet-zout grondwater, Deltares rapport 1220185-000, 86 p.

Voss, C.I. (1984) SUTRA – A finite element simulation for saturated-unsaturated, fluid-density-dependent ground-water flow with energy transport or chemically reactive single-species solute transport. U.S.G.S. Water-Resources Investigations Report 84-4369, 409 p.

Zheng, C., and Wang, P. P. (1999). MT3DMS: A modular three-dimensional multispecies transport model for simulation of advection, dispersion, and chemical reactions of contaminants in groundwater systems; documentation and user's guide. Contract Report SERDP-99-1, U.S. Army Engineer Research and Development Center, Vicksburg, MS.

Appendix A *Create_DISU* script

Two parts of the *Create_DISU* Python script are shown here. The full script contains 481 lines, which is too long to fully show here.

The first part that is shown here (line 209 – 255) shows the lines where for every cell the neighboring cells on the top and sides are found:

```
209 #Finding vertical connections with over(!)lying cells:
210 n_min=0
211 for l in range(0,nlay-1): #nlay-1 because one layer doesnt have overlying cells
212     for icell in range(ncell_lay_cum[l],ncell_lay_cum[l+1]): #starting with cells in the second layer
213         x_min=coords[icell][0]
214         y_min=coords[icell][1]
215         y_max=coords[icell][2]
216         x_max=coords[icell][3]
217 
218         n_max=ncell_lay_cum[l]
219 
220         for i in range(n_min,n_max):#cells in only the layer on top of the lower layer
221             if coords[i][0] < x_max and coords[i][1] < y_max and coords[i][2] > y_min and coords[i][3] > x_min: #check of cell dimensions overlap with cells in the layer above
222                 cell_connect[icell].append(i+1)
223                 connect_dir[icell].append(0)
224 
225     n_min=n_min+ncell_lay[l]
226 
227 #create a list for every cell that contains all cells it is horizontally connected to:
228 class OrderedCounter(Counter, OrderedDict):
229     pass
230 
231 n=0
232 icell=0
233 lmin=0
234 for l in range (0,nlay):
235     for icell in range(lmin,ncell_lay_cum[l]):
236         vertexnr1=nodes[icell]
237         for i in range(icell+1,ncell_lay_cum[l]):
238             vertexnr2=nodes[i]
239             while n<len(vertexnr1):
240                 if vertexnr2.count(vertexnr1[n])>0 and icell!=i: #the cells have overlapping nodes and aren't the same cell
241                     overlap[icell].append(i+1)
242                     overlap[i].append(icell+1)
243                 n=n+1
244             n=0
245             if len(overlap[icell]) == ((len(nodes[icell])*2)+4): #all the cells with overlapping nodes are found
246                 break
247         neighbor = [k for k, v in OrderedCounter(overlap[icell]).items() if v > 1] #only contains every cell that has multiple connecting nodes with current cell
248         cell_connect[icell].extend(neighbor) #add the horizontally connected cells
249         x=0
250         while x < len(neighbor):
251             connect_dir[icell].append(1)
252             x+=1
253         cell_connect[icell].insert(0,icell+1)#add the cellid of current cell
254         connect_dir[icell].insert(0,1) #note!! a horizontal connection direction (IHC=1) is added to the connection between the cell and itself.
255     lmin=ncell_lay_cum[l]
```

The second part (lines 278 – 329) shows how, for every connection between cells, the distance of the connection, the width of the connection (for horizontal connections) or area of the connection (for vertical connections) and the angle of the connection is found:

```
278 #find distance to neighboring face and width/area of the flow connection
279 def overlap(min1, max1, min2, max2): #function to find the overlap between 2 lines
280     return max(0, min(max1, max2) - max(min1, min2))
281 
282 def connection_information(connect_right,distance,angle):
283     connect_dist[icell].append(distance)
284     angldegx[icell].append(angle)
285 
286 
287 for icell in range(0,ncell):
288     connect_nodes=[]
289     cell_connections=cell_connect[icell] #list with all connected cells for cell i
290 
291     x_min_base = coords[icell][0]
292     y_min_base = coords[icell][1]
293     y_max_base = coords[icell][2]
294     x_max_base = coords[icell][3]
295 
296     x_dist=(abs(x_max_base-x_min_base))/2
297     y_dist=(abs(y_max_base-y_min_base))/2
298     vert_dist=(abs(top[icell]-bot[icell]))/2
299 
300     connection_number=1
301     while connection_number < len(cell_connections):
302 
303         x_min_connect = coords[cell_connections[connection_number]-1][0] #dimensions of the connected cell
304         y_min_connect = coords[cell_connections[connection_number]-1][1]
305         y_max_connect = coords[cell_connections[connection_number]-1][2]
306         x_max_connect = coords[cell_connections[connection_number]-1][3]
307 
308         if top[cell_connections[connection_number]-1] != top[icell]: #This means that the cells are not horizontally connected
309             flow_area1 = overlap(y_min_base,y_max_base,y_min_connect,y_max_connect) #y-side of the overlapping area
310             flow_area2 = overlap(x_min_base,x_max_base,x_min_connect,x_max_connect) #x-side of the overlapping area
311             flow_width[icell].append(flow_area1*flow_area2) #overlapping area between vertically connected cells.
312             connection_information('no',vert_dist,0.0)
313         elif x_min_connect == x_max_base: #horizontal connection at right side of cell
314             flow_width[icell].append(overlap(y_min_base,y_max_base,y_min_connect,y_max_connect))
315             connection_information('yes',x_dist,0.0)
316         elif x_max_connect == x_min_base: #horizontal connection at left side of cell
317             flow_width[icell].append(overlap(y_min_base,y_max_base,y_min_connect,y_max_connect))
318             connection_information('no',x_dist,180.0)
319         elif y_min_connect == y_max_base: #connection in direction of positive y-axis
320             flow_width[icell].append(overlap(x_min_base,x_max_base,x_min_connect,x_max_connect))
321             connection_information('no',y_dist,90.0)
322         elif y_max_connect == y_min_base: #connection in direction of negative y-axis
323             flow_width[icell].append(overlap(x_min_base,x_max_base,x_min_connect,x_max_connect))
324             connection_information('no',y_dist,270.0)
325         connection_number=connection_number+1
326 
327     connect_dist[icell].insert(0,0.0) #the first value in connect_dist contains distance from cell center to itself, so this is always 0.
328     flow_width[icell].insert(0,0.0) #the first value in flow_width is for flow from cell to itself, so always 0.
329     angldegx[icell].insert(0,0)
```

Appendix B *mf2005_to_MF6_DISU* script

Two parts of the Python script that is used to translate the input files from older MODFLOW versions or SEAWAT to MODFLOW 6. First, the part of the script where the grids are 'linked' is shown:

```
def closest(X, p):
    disp = X - p
    return np.argmin((disp*disp).sum(1)) #np.argmin returns the indices of the minimum values along an axis
```

```
# Build horizontal coordinates for structured and unstructured cells
col_index = []
row_index = []
coord_structured = []
for row in range(0, nrow):
    for col in range(0, ncol):
        col_index.append(col)
        row_index.append(row)
        if col == 0 and row == 0:
            coord_structured.append([0.5*delr[0], 0.5*delc[0]])
        elif col == 0:
            coord_structured.append([0.5*delr[0], sum(delc[0:row]) + 0.5*delc[row]])
        elif row == 0:
            coord_structured.append([sum(delr[0:col]) + 0.5*delr[col], 0.5*delc[row]])
        else:
            coord_structured.append([sum(delr[0:col]) + 0.5*delr[col], sum(delc[0:row]) + 0.5*delc[row]])

coord_unstructured = []
of = open(root + "cell_centers.txt", "r")
for lines in of:
    line = lines.split()
    coord_unstructured.append([float(line[0]), float(line[1])])

of.close()

# For each unstructured cell, get index of closest structured cell
Index_closest_structured_cell = []
for i in range(0, ncell_unstructured):
    Index_closest_structured_cell.append(closest(np.array(coord_unstructured[i]), coord_structured))

# DISU:
# For each structured cell in a layer, get index of closest unstructured cell
Index_closest_DISU_cell = []
coord_unstructured_lay = []

for lay in range (0,nlay):
    Index_closest_DISU_cell.append([])
nmin=0
nmax=0
for lay in range (0,nlay):
    nmax+=cellay[lay]
    for n in range(nmin,nmax):
        coord_unstructured_lay.append(coord_unstructured[n])#only look into one unstructured layer

    for i in range(0, ncell_structured):
        Index_closest_DISU_cell[lay].append(closest(coord_structured[i], np.array(coord_unstructured_lay))+nmin)
    nmin=nmax
    coord_unstructured_lay = []
```

Below is the part of the script where the input for the input for the well file is read from a SEAWAT input file and written to a MODFLOW 6 file.

```
#Read WEL file
with open(welfile_temp, "w") as f:
    f.write("BEGIN OPTIONS\n")
    f.write("END OPTIONS\n\n")
    f.write("BEGIN DIMENSIONS\n")
    f.write(" MAXBOUND 40 \n") # ES: note these are the cells that contain wells
    f.write("END DIMENSIONS\n\n")
    for i in range(0,2):
        welfile.readline()
    nwel=6134 #number of lines in welfile
    per = 0
    for i in range(0,nwel):
        line = welfile.readline()
        if line[13:17] == 'week':
            per = per+1
            if per != 1:
                f.write("END PERIOD\n\n")
                f.write("BEGIN PERIOD " + str(per) + "\n")
            else:
                f.write("BEGIN PERIOD " + str(per) + "\n")
        elif line[8:10] == '10':
            wel_lay = int(line.split()[0])
            wel_row = int(line.split()[1])
            wel_col = int(line.split()[2])
            wel_dis = float(line.split()[3])
            wel_usg_min = (wel_lay-1)*cellay_fine
            wel_usg_max = (wel_lay)*cellay_fine
            index_structured_gridcell = (wel_row-1)*ncol + wel_col - 1
            res_list = [k for k, value in enumerate(Index_closest_structured_cell[wel_usg_min:wel_usg_max]) if value == index_structured_gridcell]
            x_wel = coord_unstructured[res_list[0]][0]
            rows=[]
            for j in range(0,len(res_list)):
                if coord_unstructured[res_list[j]][0] == x_wel and coord_unstructured[res_list[j]][1] < 40.0:
                    rows.append(res_list[j]+wel_usg_min+1)
            wel_dis_fine=wel_dis/len(rows)
            for k in range(0,len(rows)):
                f.write("   " + str(rows[k]) + " " + str(wel_dis_fine) + "\n")
    f.close()
```

Appendix C Iterative Model Solution (IMS) settings

The IMS file of both the GWF (left) as the GWT (right) model:

```
BEGIN OPTIONS
 COMPLEXITY COMPLEX
END OPTIONS

BEGIN NONLINEAR
 OUTER_HCLOSE 0.001
 OUTER_MAXIMUM 1000
END NONLINEAR

BEGIN LINEAR
 INNER_MAXIMUM 30
 INNER_HCLOSE 0.0001
 INNER_RCLOSE 1.
 LINEAR_ACCELERATION CG
 RELAXATION_FACTOR 0.97
END LINEAR
```

```
BEGIN OPTIONS
 COMPLEXITY COMPLEX
END OPTIONS
```

Appendix D Vertically unstructured grids

The grids that are used in the study of the implementation of a vertical unstructured grid are shown here. The original spatial discretization of the SEAWAT model is used as fine grid in the top layers of these cases. This grid is shown in the middle picture in Figure D-1, the left and right pictures show respectively a coarsened grid along the x-axis and along the y-axis. These grids are implemented in separate cases at the layers below the fine grid. The result is a transition where one cell of the fine grid overlays two cells in the coarse grid.

Besides coarsening over the whole area, the coarsening is also done in only the telescopic area, which is shown in Figure D-2. The pictures shown are the 100 x 100 m area of the whole grid. The grid far away from the well at x,y = 0,0 keeps the same cell sizes. For these cases, the grid is coarsened in x- or y-direction (top left and bottom right in Figure D-2) or in both directions together (top right).

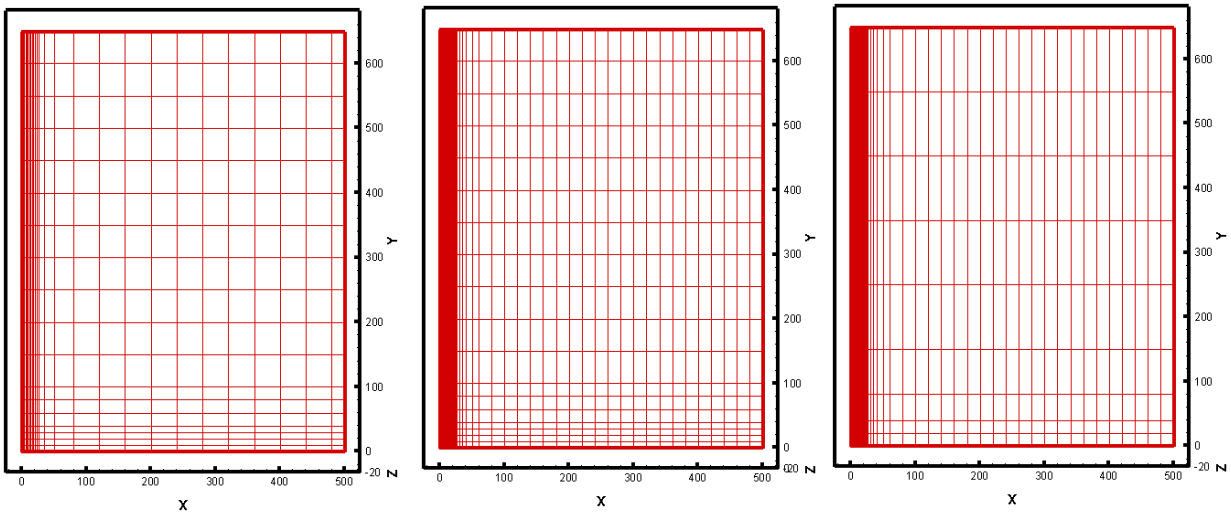

Figure D-1. Fully coarsened grids. Left: coarsening along cell width (x-direction). Middle: original fine discretization. Right: coarsening along cell length

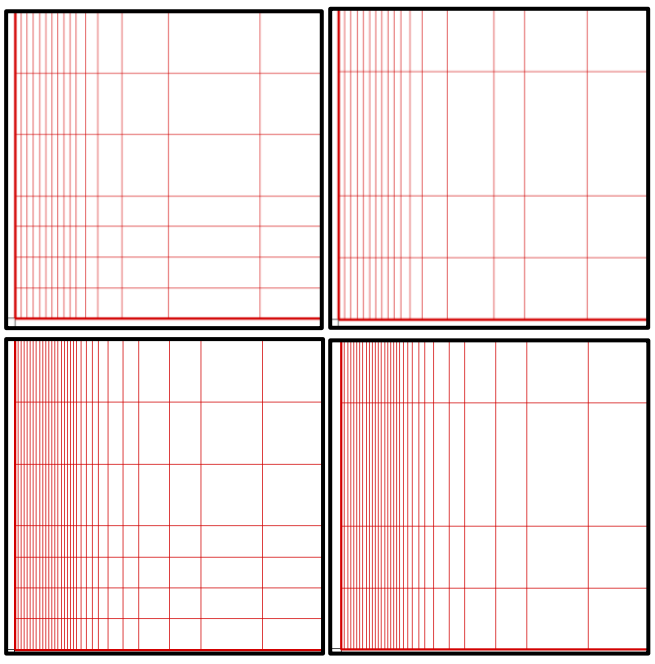

Figure D-2. 100x100 m of partly coarsened grids. Top left: coarsened along cell width (x-direction). Bottom left: original fine discretization. Top right: coarsened along cell width and length (x- & y-direction). Bottom right: coarsened along cell length (y-direction)

Appendix E Telescopically refined grids

In this appendix, the telescopically refined grids are shown. Figure E-1, Figure E-2 and Figure E-3 show the grids that have the same base mesh of 20 x 50 m far away from the well, but have a different fine cell size close to the well. The figures only show 100 x 100 m of the full grid. The whole grid of case 36 is shown in Figure E-4.

Figure E-4 and Figure E-7 show the cases where the cell size at the well is constant (1x5 m), but the base grid cell size is varied.

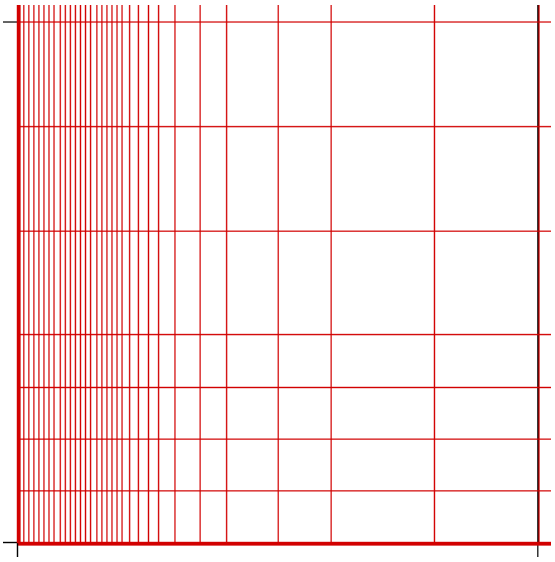

Figure E-1. Cut-out of case 35.
Cell size = 1 x 10 m at the well.

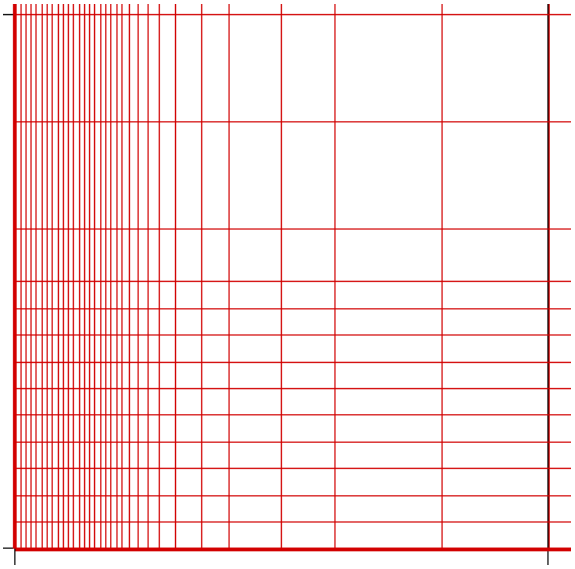

Figure E-2. Cut-out of case 36.
Cell size = 1 x 5 m at the well

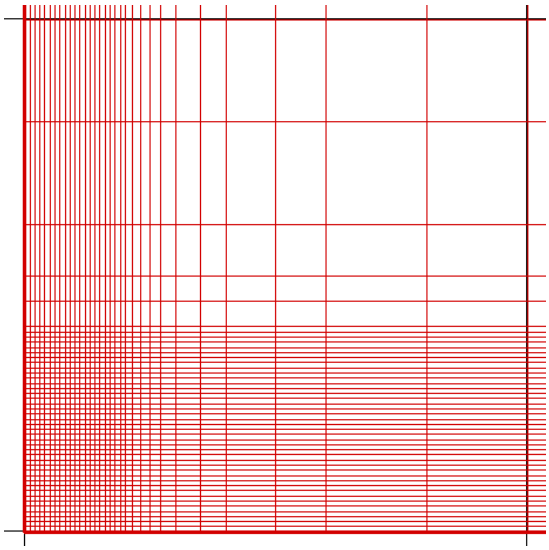

Figure E-3. Cut-out of case 37.
Cell size = 1 x 1 m at the well

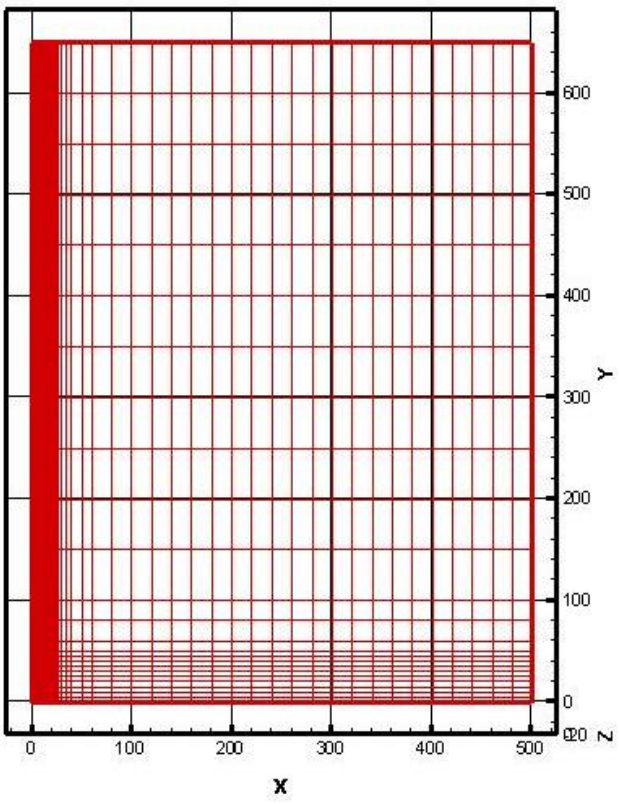

Figure E-4. Case 36. Cell size base grid = 20 x 50 m

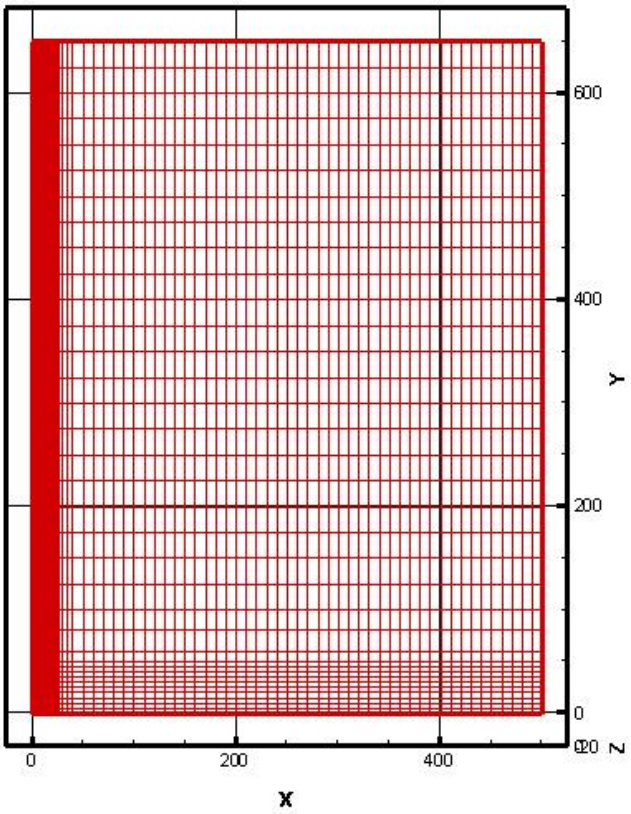

Figure E-5. Case 39. Cell size base grid = 10 x 25 m

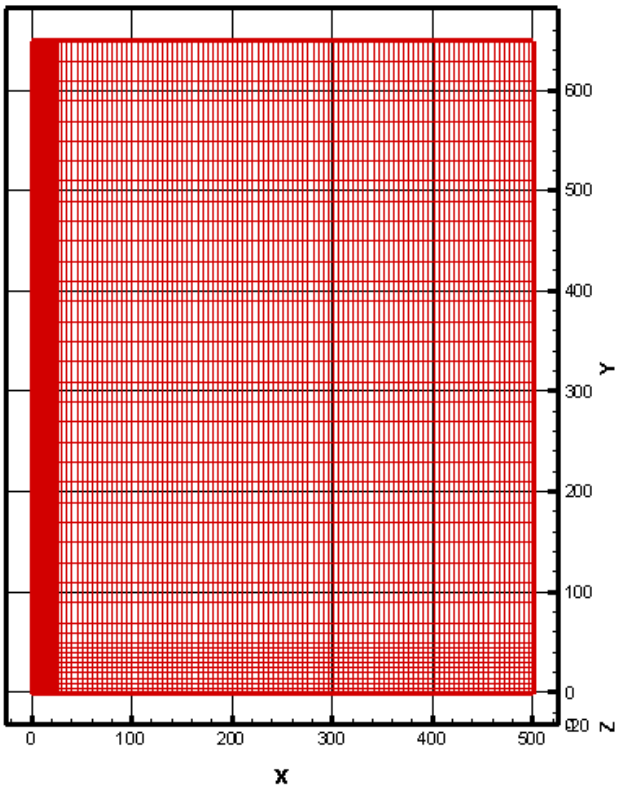

Figure E-6. Case 40. Cell size base grid = 5 x 20 m

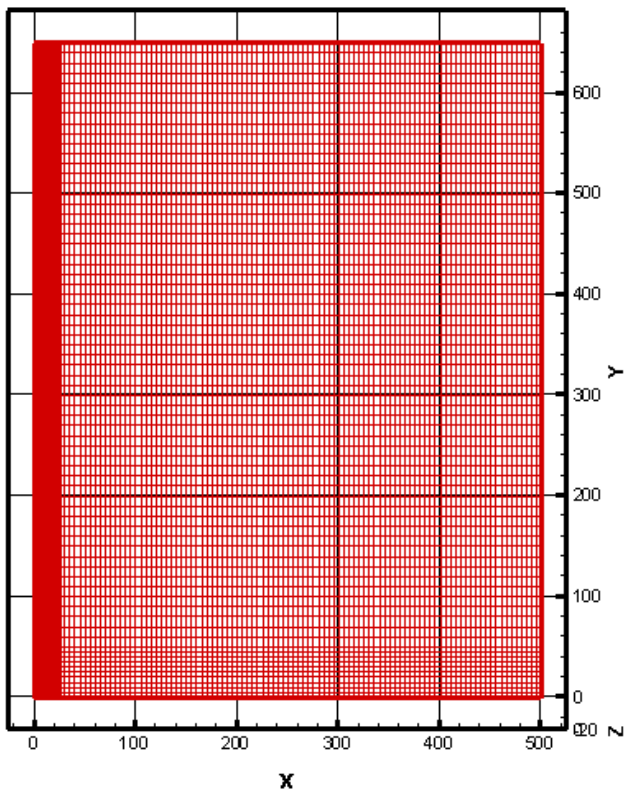

Figure E-7. Case 41 Cell size base grid = 5 x 10 m

Appendix F Horizontally unstructured grids

This appendix shows the comparison between the grids of telescopic and non-telescopic counterparts. The grids are drawn in Microsoft Excel. The first three figures show 500x350 m of the grid, the same cell sizes continue in the rest of the grid.

The last figure (Figure F-4) shows a cutout of the grid of case 45, the part that is shown is only 20 x 80 m, so the scale of this figure is different from the others.

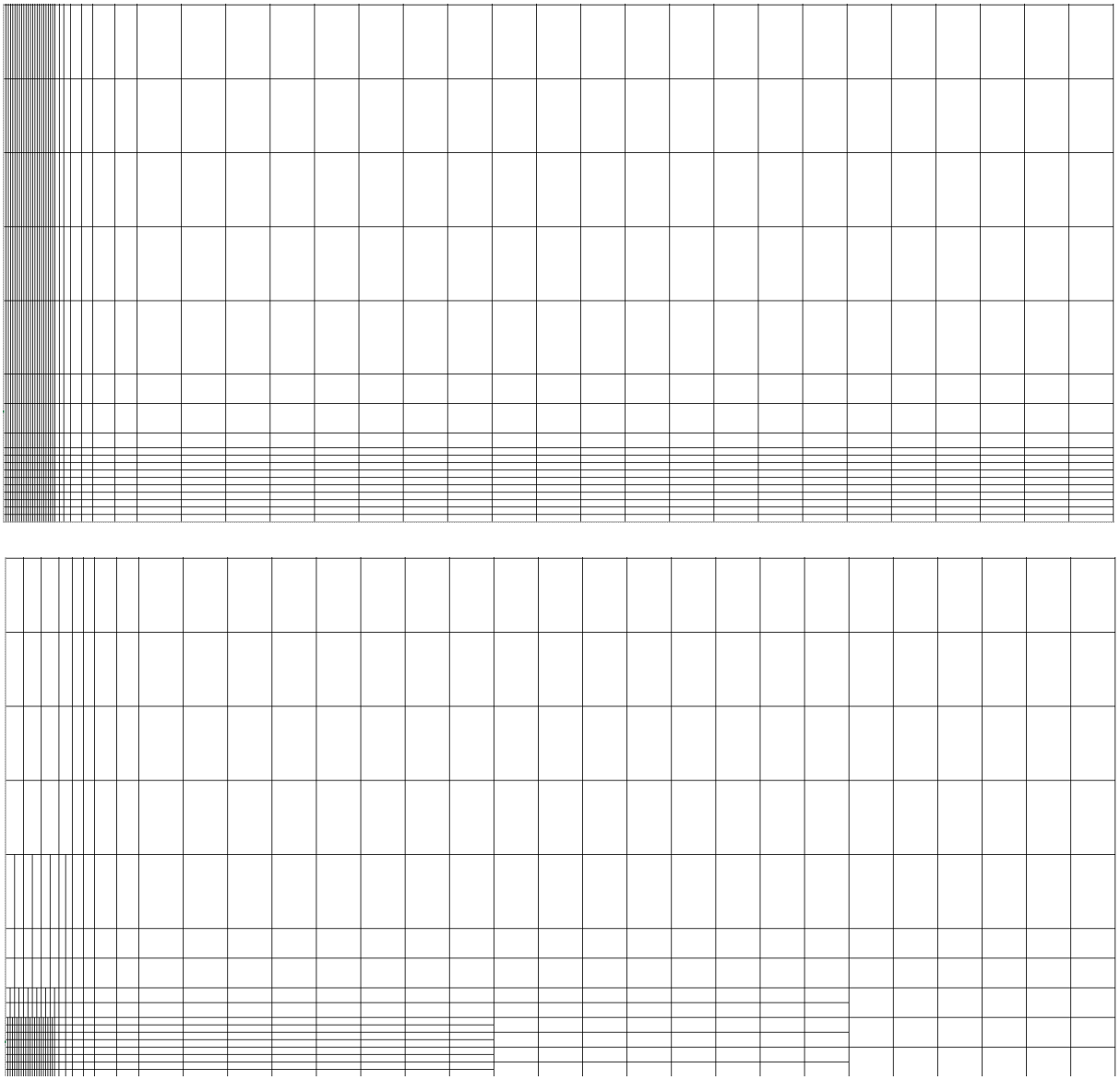

Figure F-1. Grids of case 36 (top) and case 42 (bottom). Both cases have cell size 1x5 m at the extraction well (bottom left) and cell size 20x50 m in the base grid (top right). Both grids are in plan view and show 500x350 m (so not the full grid)

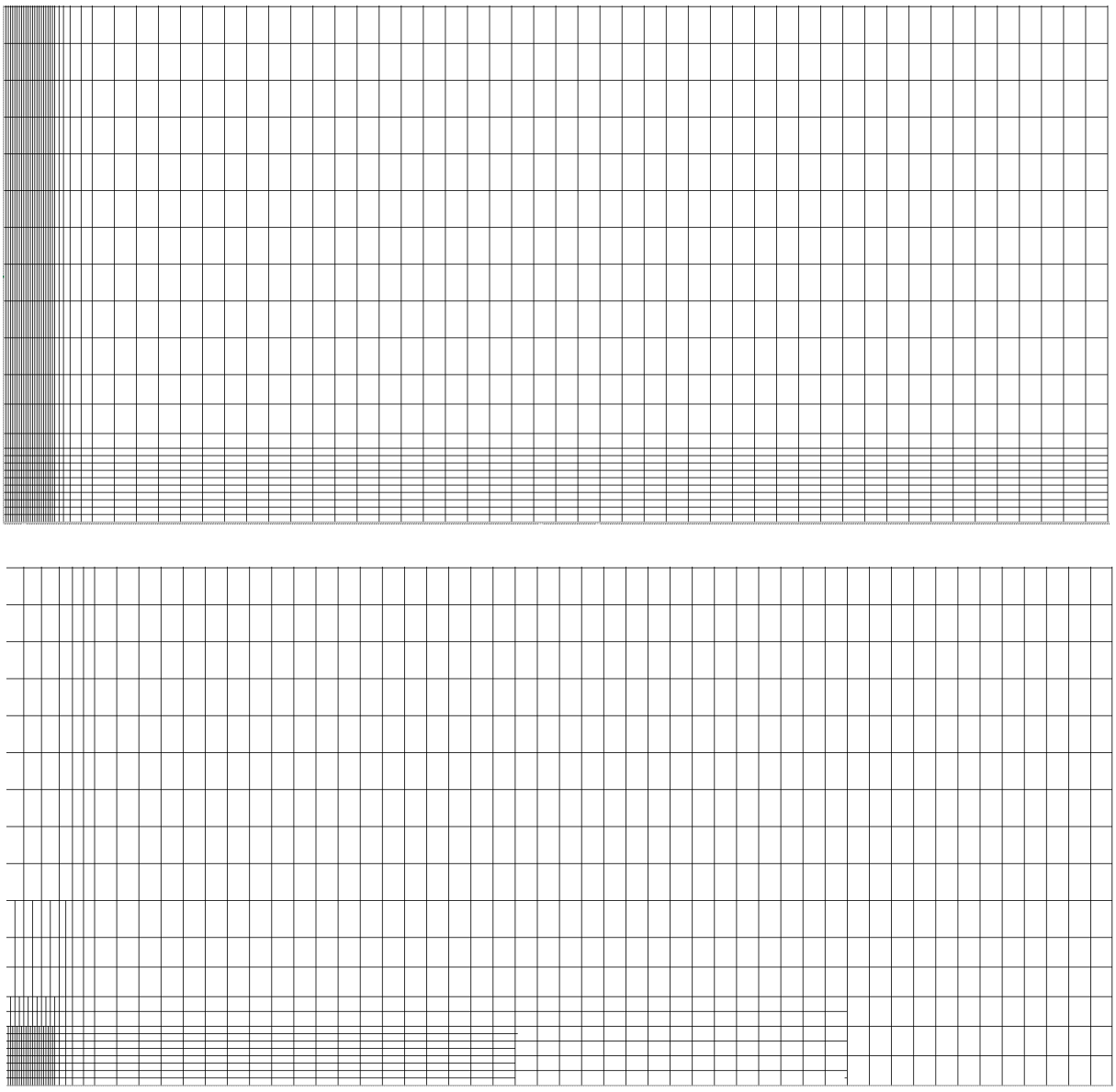

Figure F-2. Grids of case 39 (top) and case 43 (bottom). Both cases have cell size 1x5 m at the extraction well (bottom left) and cell size 10x25 m in the base grid (top right). Both grids are in plan view and show 500x350 m (so not the full grid)

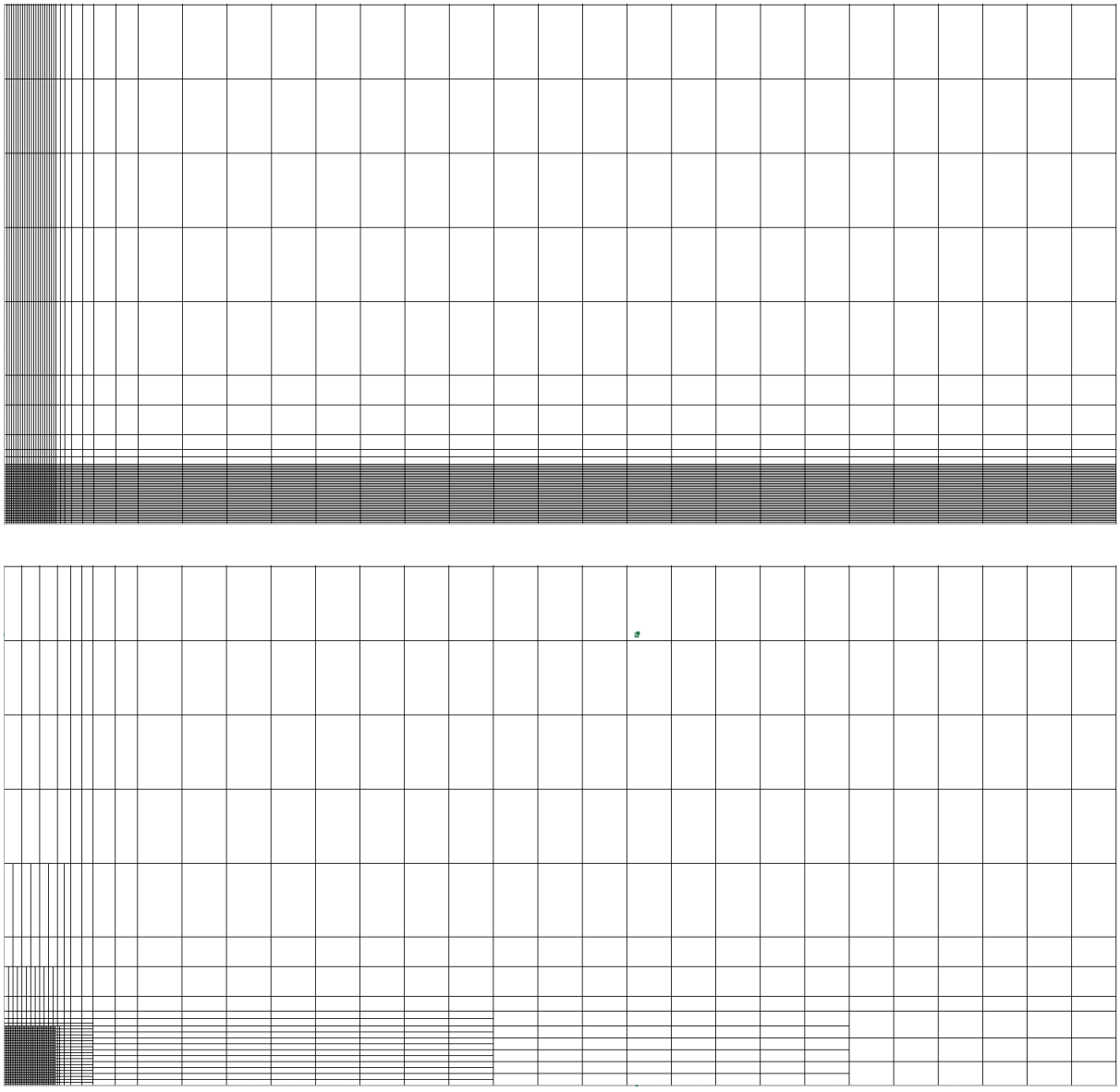

Figure F-3. Grids of case 37 (top) and case 44 (bottom). Both cases have cell size 1x1 m at the extraction well (bottom left) and cell size 20x50 m in the base grid (top right). Both grids are in plan view and show 500x350 m (so not the full grid)

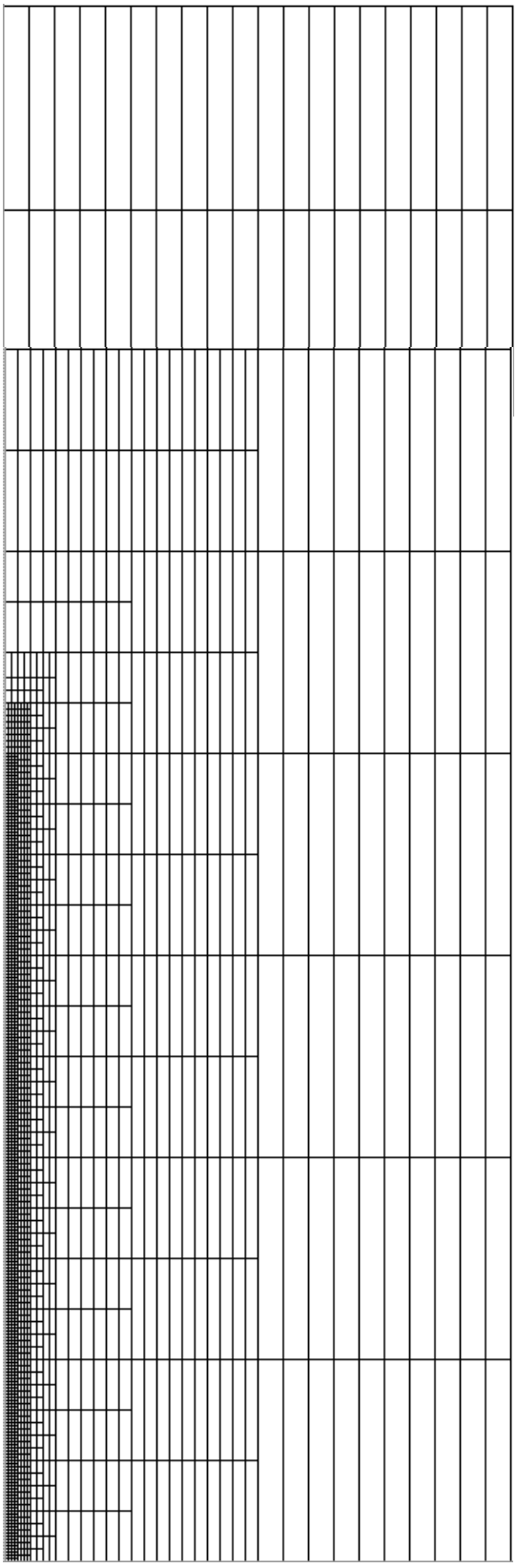

Figure F-4. Cutout (20 x 80 m) of case 45. The cell size close to the extraction well is 0.125 x 0.15625 m, the largest cells in this image are 1 x 10 m. The rest of the grid (that is not shown here) is equal to the grid of the reference cases (e.g. case 36, shown in Figure E-4).

Appendix G Save specific discharge

Below, the results of the SEAWAT model are compared to case R-1 of MODFLOW 6, that is both run with and without the SAVE-SPECIFIC-DISCHARGE setting in the NPF package.

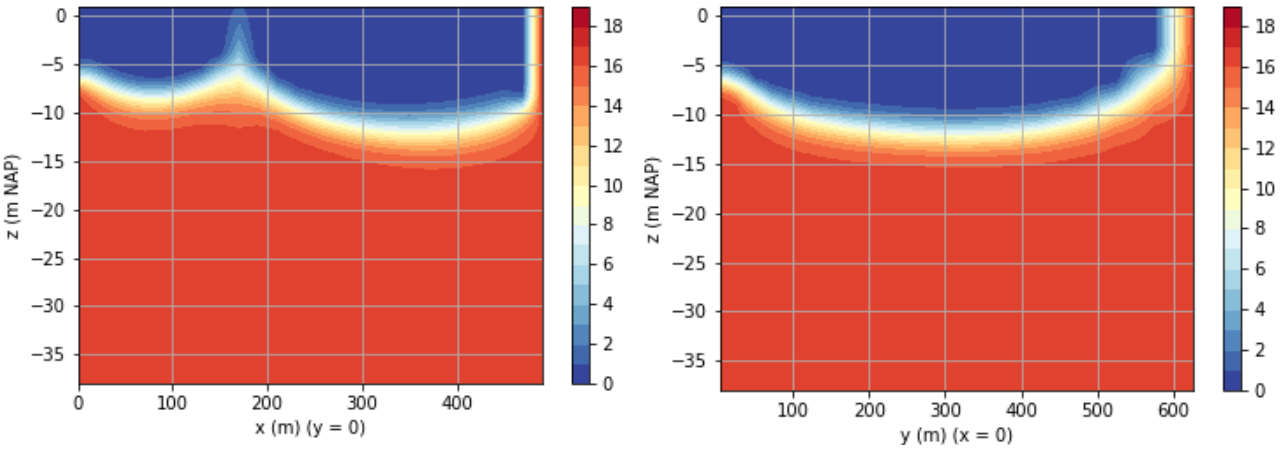

Figure G-2. Chloride concentration (kg/m³) distribution in the SEAWAT model (case R-1) after 39 years.

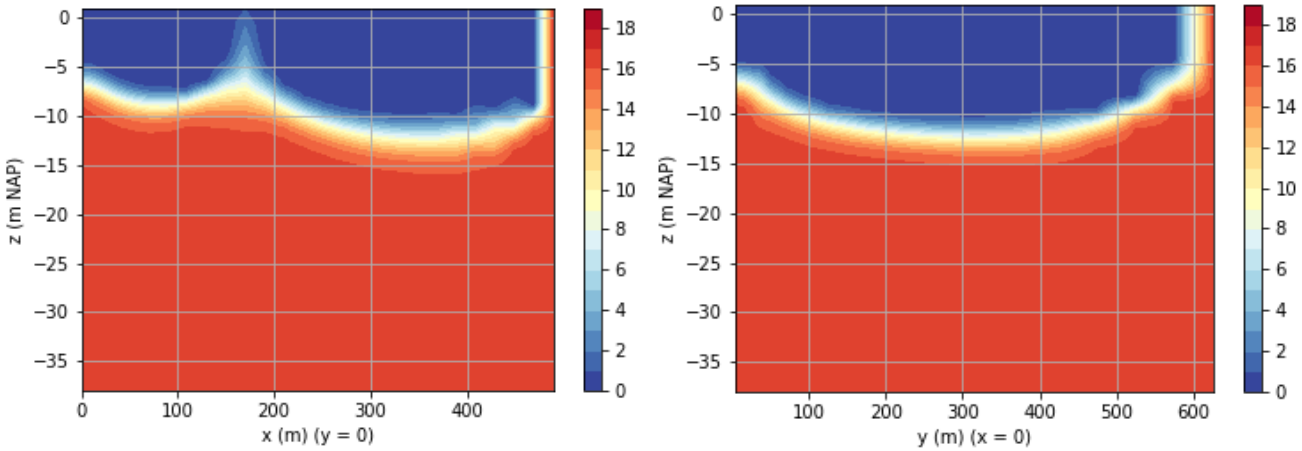

Figure G-1. Chloride concentration (kg/m3) distribution in the MODFLOW 6 model (case R-2) that contains the SAVE-SPECIFIC-DISCHARGE option, after 39 years.

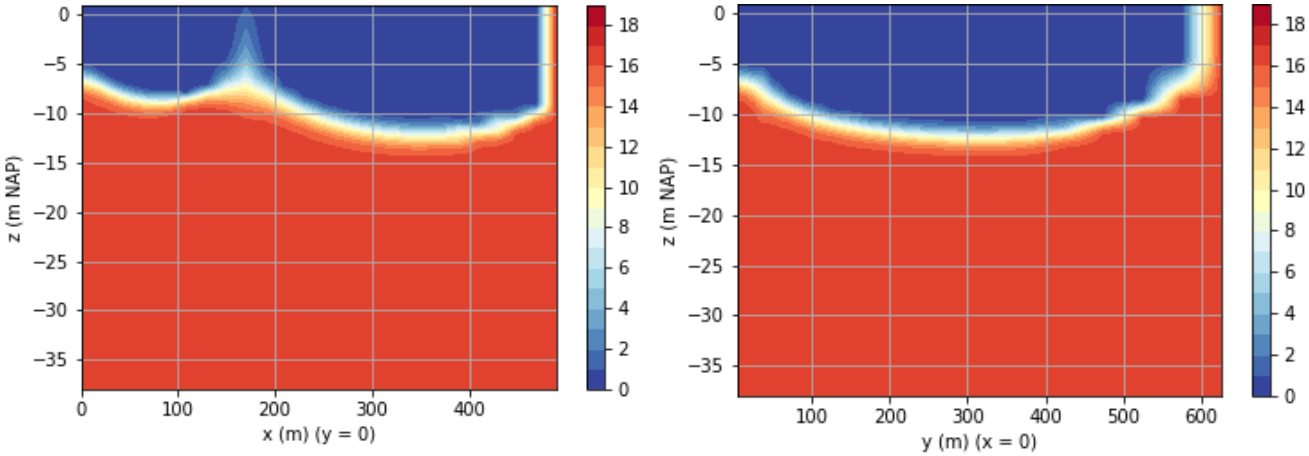

Figure G-3. Chloride concentration (kg/m3) distribution in the MODFLOW 6 model (case R-2) that NOT contains the SSD option, after 39 years.