Assessment of the usability of OpenGeoSys tools for Aquifer Thermal Energy Storage (ATES) simulations

by

Haoyue Liu
Student number: 4769902
Water Management

Delft University of Technology
A thesis presented for the degree of
Master of Science

Thesis committee:
Prof.dr.ir. M.Bakker, TU Delft, Chair Committee
Dr.ir. J.M.Bloemendal, TU Delft, Committee
Dr. D.V.Voskov, TU Delft, Committee

Wednesday 17th February, 2021

Preface

Even though 2020 was a tough year, I overcame all kinds of obstacles and accomplished this research. To be honest, the process was torturing. Learning software, building models, running simulations occupied my daily life. But because of this, this experience helped increase my anti-pressure ability.

I chose this topic based on my interest of the alternative heating technologies which was introduced in a geohydrology class. To help mitigate the green house effect, many more alternative technologies that makes the energy production more sustainable are needed. This research helped me to have a more clear understanding of how ATES systems work and how numerical tools help to design an ATES system.

I want to thank my members of my graduation committee. Thanks to their help and feedbacks, I am able to achieve this result. And I would like to express my special gratitude to my daily supervisors Mark Bakker and Martin Bloemendal. Thanks for your constructive suggestions when I met difficulties. Furthermore, I'd like to thank my parents and my friends for their supportings in both finance and spirit.

Haoyue Liu

Chengdu, China, February 2021

Abstract

Aquifer thermal energy storage (ATES) is an energy efficient technology to temporarily store groundwater of different temperatures in an aquifer. The basic idea behind ATES is to store thermal energy in warm and cold wells so that the energy can be used for heating and cooling of buildings in the next season.

Design and planning of ATES systems requires numerical simulation tools such as SEAWAT, COMSOL, and FEFLOW. COMSOL and FEFLOW are expensive commercial products. SEAWAT is a free computer program based on MODFLOW and MT3DMS. The flow field is modelled with the finite difference method while there are several methods to simulate the heat transport including the finite difference (FD) method and the total variation diminishing (TVD) method. OpenGeoSys (OGS) is an open-source alternative based on the finite element method which is able to simulate groundwater flow and heat transport processes and can potentially be used for design and planning of ATES systems.

There are very few applications of OGS to simulate ATES systems. The main goal of this research is to assess the usability of OpenGeoSys in the simulation of ATES systems. The numerical solutions of OGS and SEAWAT (both the TVD scheme and FD scheme) are compared to the analytical solutions for a single well that injects water with a constant flow rate of water and constant temperature. For a doublet with a cold well and a warm well, the OGS solution is compared to the SEAWAT solution for both a single cycle system and a multiple cycles system with constant flow rates and temperatures. Finally, a field case of a doublet with varying flow rates and injection temperatures is studied to compare the performance of OGS to SEAWAT.

The main findings of this study are as follows. Both OGS and SEAWAT can reproduce the heat transport process of the analytical solutions for a single well injecting warm or cold water. There are some deviations between the temperature distributions computed by the numerical models and the analytical solutions which are primarily caused by numerical dispersion. The TVD scheme results in the smallest numerical dispersion while OGS shows slightly more numerical dispersion when applying the same number of cells/nodes. Numerical dispersion can be reduced by application of finer spatial resolution

for the SEAWAT since the time step is adjusted automatically. For OGS, the time step must be reduced manually when the grid is refined.
Both OGS and SEAWAT show energy balance errors smaller than 1%. Numerical dispersion has an effect similar to physical dispersion so that the thermal radius increases when the numerical dispersion is larger. The interaction between the warm well and cold well of a doublet increases when the thermal radius increases, which may result in an increase of the energy balance error.
According to the analytical solution for the model where the heat transport between the aquifer and aquitards is included, OGS shows the smallest overestimation of thermal energy that remains in the aquifer part compared to the two solution schemes of SEAWAT. The vertical temperature distribution in aquitards simulated with OGS is closer to that of the analytical solution compared to SEAWAT.
Both numerical dispersion and the overestimation of thermal energy in the aquifer have an influence on thermal recovery efficiency of an ATES system. Larger numerical dispersion indicates a longer heat transport distance, resulting in more time needed for thermal energy to transport back to the wells during the extraction period. Similarly, the overestimation of the thermal energy in the aquifer provides more thermal energy during the extraction period. There are some overshoots/undershoots when simulating an ATES system with varying flow rates and injection temperatures by the TVD scheme. This can be eliminated by refining the spatial resolution.
It is more complicated to simulate an ATES system with OGS than SEAWAT. The computational time is much longer for OGS compared with SEAWAT by a factor of around 10 for the same number of cells/nodes. Despite these drawbacks, OGS is suitable for the simulation of ATES systems as the simulated temperature distribution in both the aquifer and aquitards compares well to the analytical solution, and there are no problems of overshoots/undershoots in the simulation of varying flow rates and injection temperatures.

Keywords— OGS - SEAWAT - ATES - Analytical Solution

Table of Contents

1 Introduction **1**

1.1 Aquifer Thermal Energy Storage System 1

1.2 Aim and Objectives 2

1.3 Thesis Outline 4

2 Material and Method **5**

2.1 Governing Processes 5

2.1.1 Groundwater Flow 5

2.1.2 Heat Transport 5

2.2 Software 6

2.2.1 OpenGeoSys Tools 6

2.2.2 SEAWAT Tools 8

2.3 Application of OpenGeoSys in coupled groundwater flow and heat transport processes 10

2.4 Analytical Solutions 11

2.4.1 Groundwater Flow Steady-state Analytical Solution 11

2.4.2 Heat Transport Analytical Solutions for Different Models . 12

2.4.2.1 Sand Model 12

2.4.2.2 Clay-sand-clay Model 12

2.5 Numerical Methods 15

2.6 Scenarios 23

2.6.1 Sand model 23

2.6.2 Clay-sand-clay model 24

2.7 Numerical Model Implementations 26

2.7.1 OGS 28

2.7.2 SEAWAT 36

2.8 Integration methods 39

2.9 Assessment Framework 43

3 Results **45**

3.1 Compare the performance of OGS and SEAWAT in one-well model with the analytical solution 45
3.1.1 Sand model 45
3.1.2 Clay-sand-clay model 50
3.2 Compare the performance of OGS and SEAWAT in the single-cycle ATES system 55
3.3 Compare the performance of OGS with SEAWAT when applied in the multi-cycle ATES system 62
3.4 Compare the performance of OGS with SEAWAT in field case . . 65
3.4.1 Data Aggregation 65
3.4.2 Results of the temperature at the doublet 67

4 Discussion **70**

5 Conclusions **74**
5.1 Conclusions 74
5.2 Recommendations 75

Bibliography **77**

Appendix A Implementing a doublet in OpenGeoSys **81**

Appendix B Spatial and temporal resolution adjustment in clay-sand-clay model **83**

List of Figures

1.1 Basic working principle of ATES systems. Direct cooling mode is applied in summer. In winter, a heating mode supported by a heat pump is used. 1

2.1 Mathematical framework of coupled THMC modelling. 7
2.2 Clay-sand-clay model layout. 13
2.3 Typical finite element mesh components. 17
2.4 Three-dimensional element: hexahedron and its representation in the local coordinate system. 17
2.5 The index system used for the finite-difference grid. 19
2.6 Illustration of the nodal points involved in the ULTIMATE scheme. 20
2.7 Example meshgrid. 22
2.8 Sand model mesh top view. 23
2.9 Clay-sand-clay one-layer aquifer model mesh cross-section view. . 25
2.10 Clay-sand-clay multi-layer aquifer model mesh cross-section view. 26
2.11 Main idea behind the Simpson's rule. 40
2.12 Sum-up method example. 42
2.13 Illustration of numerical dispersion. 44

3.1 Head distribution along the radial direction at t=180 days. 45
3.2 Temperature distribution along the radial direction of the analytical solution and numerical solutions at different time (black-50 days, red-100days, blue-180days); A(analytical), T(TVD), F(Finite Difference), O(OGS). 46
3.3 Temperature distribution at t=180 days for the exact solution and numerical solutions for different spatial resolutions at a constant time step of 5 days for OGS; A (analytical), T (TVD), F (Finite Difference), O (OGS). 48
3.4 Temperature distribution at t=180 days for the exact solution and OGS for different time steps and a horizontal spatial resolution of 5m$\times$5m; A (analytical), O (OGS). 49

3.5 Temperature distribution after one-time-step simulation at the first clay layer above the aquifer in SEAWAT environment (isotropic dispersivity). 51
3.6 Temperature distribution after one-time-step simulation at the first clay layer above the aquifer in SEAWAT environment (anisotropic dispersivity). 51
3.7 Temperature distribution along the radial direction in the aquifer of the analytical solution and numerical solutions at t = 180 days. . 52
3.8 Vertical temperature distribution in the aquitard with different distances to the wellbore after 180 days. 53
3.9 Wells location in the doublet model. 56
3.10 Temperature distribution along observation line in the aquifer after 360 days. 57
3.11 Temperature distribution along observation line in the aquitard after 360 days. 57
3.12 Temperature of the warm well at different operation time. 58
3.13 Temperature of the warm well at different operation time when applying the temporal and spatial resolution adjustment in both SEAWAT and OGS. 60
3.14 Temperature of wells at different operation time in multi-cycle model in SEAWAT and OGS environment. 63
3.15 Recovery efficiency variation against the operation cycles. 64
3.16 Energy balance error variation against the operation cycles. 65
3.17 a)flow rate at the warm well (positive represents the injection while negative represents the extraction); b)measuring temperature at warm/cold well 66
3.18 The numerical solutions of the doublet temperature variation in operation at 3M building. 68
3.19 The numerical solutions of the doublet temperature variation in operation at 3M building(finer cell size for SEAWAT). 69

B.1 Temperature distribution along the observation line after temporal resolution adjustment in OGS. 83

B.2 Temperature distribution along the observation line after spatial resolution adjustment in SEAWAT. 84
B.3 Temperature distribution for numerical models after spatial/temporal resolution adjustment. 85

List of Tables

2.1 OGS input files in ATES systems simulations. 8
2.2 MODFLOW and MT3DMS packages in ATES systems simulations 10
2.3 Model parameters applied in the Sand model. 24
2.4 Model parameters applied in the Clay-sand-clay model. 27

3.1 Energy balance error (%) for different models in Sand model. . . . 50
3.2 Energy balance error (%) for different models in Clay-sand-clay model. 54
3.3 Energy balance error (%) for different models in doublet model . . 61

List of Abbreviations

ATES . . . . . Aquifer Thermal Energy Storage System

VTK . . . . . Visualization Toolkit

OGS . . . . . OpenGeoSys

OO . . . . . Object-oriented

1 | Introduction

1.1 Aquifer Thermal Energy Storage System

Climate change has become a global issue due to the increase of greenhouse gases produced by fossil fuels consumed by heating, cooling and other processes. In order to mitigate this phenomenon, some sustainable technologies need to be implemented in buildings. Geothermal energy is regarded as a promising alternative energy source as it is suited properly for base-load energy supply which can replace the fossil fuels power generation (Kolditz et al., 2013). In this study, a sustainable technology based on geothermal energy is studied.
Aquifer thermal energy storage (ATES) is a technology that takes advantage of the groundwater stored at different temperatures to provide energy efficient heating and cooling for buildings (Bloemendal & Hartog, 2018). This technology is implemented in temperate climates since both the heating and cooling are needed throughout a year (Bloemendal et al., 2015).

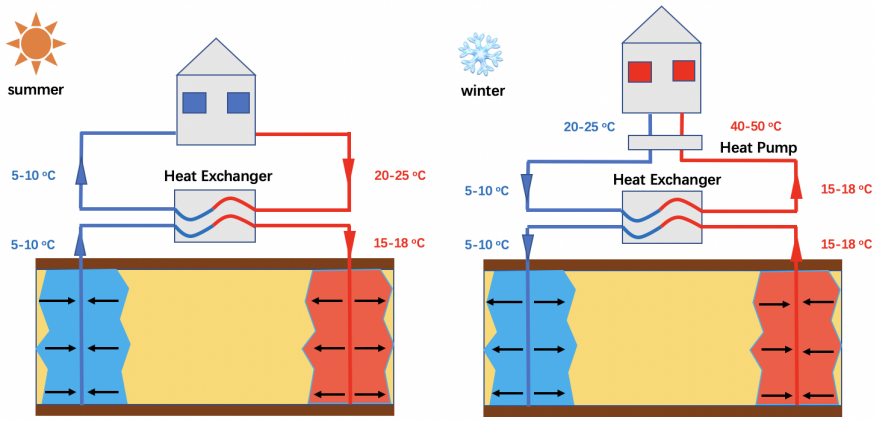


Figure 1.1: Basic working principle of ATES systems. Direct cooling mode is applied in summer. In winter, a heating mode supported by a heat pump is used.

Generally, ATES systems include two separate wells (a doublet) which are used for seasonal storage for cold and warm water in aquifers. The warm well and the cold well are installed in the same aquifer with sufficient distance between them to mitigate the energy loss (Figure 1.1). In ATES systems, the water is infiltrated

into the aquifer and extracted in next season. The water is stored at 6-12^{o}C in the cold well and 12-25^{o}C in the warm well. A large amount of thermal energy can be stored in the aquifer due to the high specific heat capacity of water. During summer time, the water is extracted from the cold well for cooling buildings. The thermal energy is transported from the groundwater to the water for buildings by a heat exchanger. Meanwhile, the heated groundwater is injected back to the warm well to store the heating energy. In winter time, the flow direction is reversed and the warm water is extracted from the warm well to heat buildings and the cooled water is injected to cold well. A heat pump often operates together to heat the water to a sufficiently high temperature since the building heat systems cannot be operated on the low temperature of the warm well.

1.2 Aim and Objectives

Design and planning of ATES systems requires numerical simulation tools. The increase of demanding of this sustainable technology in various places in the world requires easy access to such simulation tools. Currently, SEAWAT, COMSOL and FEFLOW are commonly used in ATES systems simulations (Seibt & Kellner, 2003; Zeghici et al., 2015; Dharma, 2009; Bozkaya et al., 2017). However, COMSOL and FEFLOW are expensive commercial products. Numerical modeling based on the FEM is recommended by Aravena et al.(2016) for geothermal potential estimation as it provides a rigorous way to evaluate and understand conceptual models and the system heat transfer mechanisms. Compared to the FDM which is relying on structured grids, the local grid refinement is feasible in FEM. As an open-source software based on FEM that is able to simulate groundwater flow and heat transport processes, OpenGeoSys (OGS) has the potential to be available additional code for design and planning of ATES systems. SEAWAT only allows the simulation of the groundwater system within ATES systems, while OGS allows for incorporation of building climate installation and thus integrated assessment of ATES systems. Besides, OGS allows better possibilities for combined simulation of ATES and Borehole Heat Exchangers (BHE). Compared to FEFLOW (Trefry & Muffels, 2007) which is a commercial software, OGS 5 is a

fully open source software. As regards SEAWAT (Hughes et al., 2017), OGS 5 is a finite element solver and can handle complex geometries naturally by irregular Finite Element Method meshes. Even though the newest version of the groundwater flow solver within SEAWAT has mitigated some on this limitation, the meshes do still fall short when compared with the full flexibility of a FEM mesh (Hughes et al., 2017). Such meshes that built by using the Delaunay triangulation, use triangles or tetrahedra of appropriate sizes that can conform to the often irregular geometry of geological structures (Müller et al., 2020). In addition, the FEM is a general paradigm for partial differential equations and OGS 5 can consequently solve a series of subsurface problems, like saturated and unsaturated flow, solute transport and heat transport; all using the same numerical paradigm (Müller et al., 2020). OGS offers a promising alternative for ATES design and evaluation, however, application of ATES in OGS environment is not yet done. The main objective of this thesis is to assess the usability of OGS tools for ATES simulations. And this leads to the main research question:

How does OGS perform when simulating the temperature distribution and energy conservation of ATES systems compared with analytical solutions and other numerical methods?

The main question is answered by the following approach:

1) ***Build one-well models that consider a constant injection flow rate with a constant temperature in the OGS environment and compare the results with the SEAWAT solutions for the same models and the corresponding analytical solutions.***

2) ***Build the generic ATES model with constant flow rate and temperature that includes one cycle in both OGS and SEAWAT environments and compare the solutions.***

3) ***Consider more cycles in the generic ATES model while keep other conditions the same.***

4) ***Apply the field data with varying flow rate and temperature in the generic ATES model.***

1.3 Thesis Outline

The remainder of this report is organised as follows:

Chapter 2 — introduces the methodology applied in this study. The analytical solutions are presented as well as the model implementations. Besides, some methods used for energy computation are included.

Chapter 3 — shows how different numerical models perform in the one-well and doublet models. In addition, the exact solutions are included in the one-well models.

Chapter 4 — gives some discussions based on the results of this study.

Chapter 5 — concludes the findings for this study and gives some recommendations for future study.

2 | Material and Method

In this chapter, the research approach applied in this study is presented. The governing processes for groundwater flow and heat transport are described as well as the analytical solutions for single aquifer and aquifer/aquitard systems. The numerical method aiming to approximate the definite integral value is introduced. The integrating variables process by Paraview Tools is explained as well. In addition, the parameters utilized to assess models performance is introduced in this chapter. Moreover, a brief introduction of OGS set-up is given as well as SEAWAT.

2.1 Governing Processes

In an ATES system, the groundwater is extracted from one well and injected into another well but at different temperature. Simulation of an ATES system requires the simulation of the coupled processes of groundwater flow and heat transport.

2.1.1 Groundwater Flow

In the aquifer, both pumping and injecting influence the hydraulic head leads to a groundwater flow that affects the advection process for heat transport. A three-dimension partial-differential equation 2.8 (Harbaugh, 2005) for the movement of groundwater in the porous medium is described.

$$\frac{\partial}{\partial x}(K_{xx}\frac{\partial h}{\partial x}) + \frac{\partial}{\partial y}(K_{yy}\frac{\partial h}{\partial y}) + \frac{\partial}{\partial z}(K_{zz}\frac{\partial h}{\partial z}) + W = S_s\frac{\partial h}{\partial t} \tag{2.1}$$

where K_{**} represents the hydraulic conductivity in the x, y, z direction [m/d], respectively; W is the sources/sinks of water[d^{-1}]; S_s is the specific storage for the porous medium [m^{-1}]; h the head [m]; t the time [d].

2.1.2 Heat Transport

Heat transport process includes heat advection, heat diffusion and heat dispersion. The heat advection and diffusion result from heat transport by fluid motion

and molecular activity, respectively. The porous medium structure results in the heat dispersion. The heat transport equation in a porous medium is described by equation 2.3 (Langevin et al., 2008).

$$(1+\frac{1-\theta}{\theta}\frac{\rho_s}{\rho_f}\frac{c_{P,s}}{c_{P,f}})\frac{\partial(\theta T))}{\partial t} = \nabla[\theta(\frac{k_{T,bulk}}{\theta\rho_f c_{P,f}}+\alpha\frac{q}{\theta})\nabla T] - \nabla(qT) - q_s' T_s \quad (2.2)$$

with

$$k_{T,bulk} = \theta k_{T,f} + (1-\theta)k_{T,s} \quad (2.3)$$

where θ is the porosity of the porous media [-], ρ_s is the solid density [kg/m^3], ρ_f is the fluid density [kg/m^3], $c_{P,s}$ is the specific heat capacity of the solid phase [J/kg/K], $c_{P,f}$ is the specific heat capacity of the fluid phase [J/kg/K], $k_{T,f}$ is the fluid thermal conductivity [W/m^2], $k_{T,s}$ is the solid thermal conductivity [W/m^2], α is the dispersivity tensor [m], q is the specific discharge [m/d], q_s is the specific discharge of sources/sinks [m/d], T is the temperature [^{o}C], T_s is the temperature of sources/sinks [^{o}C].

2.2 Software

The groundwater flow and heat transport processes described in section 3.1 are solved using numerical softwares. In this study, OpenGeoSys and SEAWAT are utilized for solving the equations, respectively.

2.2.1 OpenGeoSys Tools

OpenGeoSys(OGS) is a scientific open-source initiative for numerical simulation of thermo-hydro-mechanical/chemical (THMC) processes in porous and fractured media (Kolditz et al., 2012). Figure 2.1 shows the mathematical framework to be solved for THMC processes in porous media.
Initially, the OGS code aimed at applications in environmental geoscience. For instance, the OGS has been applied in the fields of contaminant hydrology, waste deposits, water resources management and geothermal systems, but it has also been applied to new fields in energy storage. The numerical processes within

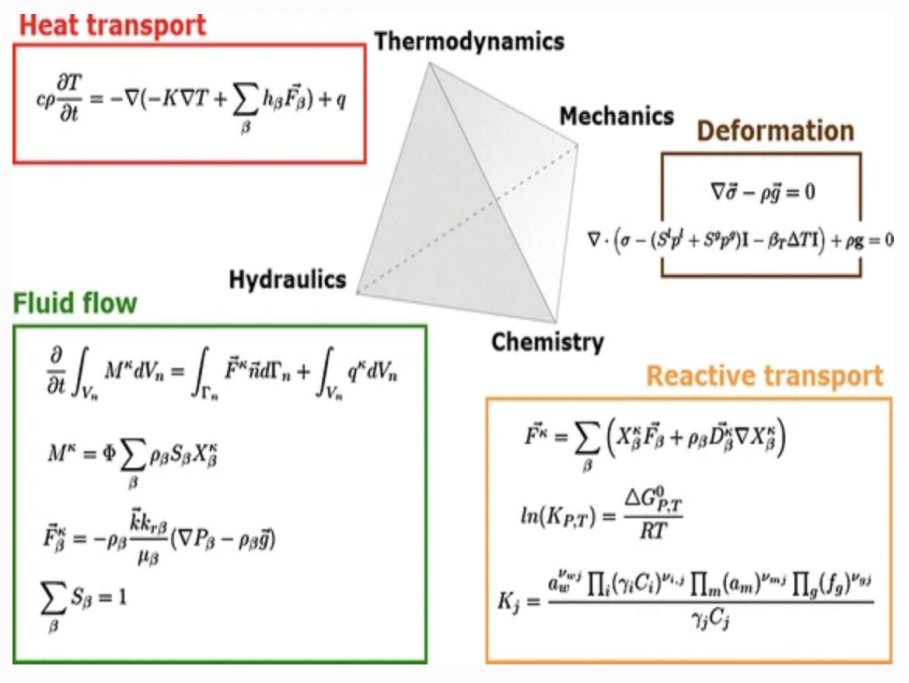


Figure 2.1: Mathematical framework of coupled THMC modelling. (Kolditz et al., 2012)

OGS are implemented in an object-oriented (OO) framework. The basic idea behind the concepts is that the solution consists of procedure steps. The procedure steps are calculation for element contributions, assembly for equation system, solution for equation systems, linear methods and calculation for other variables, that are independent for the specific process (Kolditz et al., 2012).

In this study, ogs5py is applied to build OGS models, which is a Python package that provides a Python representation of the full modeling process of OGS 5 (Müller et al., 2020). In the OGS environment, the directory structure for the basic heat transport problem (constant source/sink term) is given in table 2.1.
Generally, the process (PCS) object is defined at the beginning of building a model. Separate definitions of PCS object for different processes are required in a coupled modelling, e.g. the heat transport process and the groundwater flow process are defined separately for an ATES system model. The PCS object is designed to administrate references to mesh (MSH) objects (including mesh nodes,

Table 2.1: OGS input files in ATES systems simulations.

Object	File	Explaination
GEO	file.gli	system geometry
MSH	file.msh	finite element mesh
PCS	file.pcs	process definition
NUM	file.num	numerical solver properties
TIM	file.tim	time discretization
IC	file.ic	initial condition
BC	file.bc	boundary condition
ST	file.st	source/sink term
MFP	file.mfp	fluid properties
MSP	file.msp	solid properties
MMP	file.mmp	medium properties
OUT	file.out	output configuration

elements and topology); geometric (GEO) objects (including points, polylines, surfaces and volumes), initial conditions (IC) and boundary conditions (BC) as well as source/sink terms (ST); material data for porous data (including solid (MSP), fluid (MFP) phases, medium (MMP) and chemical properties), parameters for different numerical methods (NUM) (Kolditz et al., 2012). Eventually, the model is built after specifying a temporal discretization in time object (TIM).

2.2.2 SEAWAT Tools

SEAWAT is a software, combining MODFLOW and MT3DMS, which is designed to simulate groundwater flow and species transport. By treating heat as the solute species, SEAWAT can also simulate the heat transport. Python is used as the interface to construct the input files and to process SEAWAT output using the flopy package(Bakker et al., 2016).

MODFLOW

MODFLOW is a computer program that solves three-dimensional groundwater flow on a rectangular finite difference grid by using finite difference method (Harbaugh et al., 2000) which is written in FORTRAN. It is able to simulate the groundwater flow in multi-layer aquifer systems with adjustable grid in horizontal plane and the

same grid for each layer. The partial differential equation for groundwater flow process in the ATES systems are solved by MODFLOW. And MODFLOW2005 is applied in this study.

MT3DMS

MT3DMS is designed to solve the three-dimensional multi-species transport equation numerically. Basically, MT3DMS simulates advection, dispersion, diffusion, sorption and reaction of solute species in groundwater(Langevin et al., 2008) by finite difference method. Additionally, many solution options could be selected to solve the advection part.

In this study, the main focus is on the simulation of heat transport in the soil. In order to simulate the heat transport, some parameters are required to be adjusted as shown in equation 2.4. The K_{dtemp} is the soprtion parameter and D_{mtemp} is the effective molecular diffusion coefficient.

$$(1+\frac{\rho_b K_{dtemp}}{\theta})\frac{\partial(\theta T)}{\partial t} = \nabla[\theta(D_{mtemp}+\alpha\frac{q}{\theta})\nabla T] - \nabla(qT) - q_s' T_s \tag{2.4}$$

with

$$\rho_b = (1-\theta)\rho_s \tag{2.5}$$

$$K_{dtemp} = \frac{c_{P,s}}{\rho_f c_{P,s}} \tag{2.6}$$

$$D_{mtemp} = \frac{k_{T,bulk}}{\theta \rho_f c_{P,f}} \tag{2.7}$$

Both MODFLOW and MT3DMS utilize the 'package' approach in which features and options may be turned on or off by users depending on the requirements of the specific problem. Table 2.2 lists the packages used in this study. To get to know more information about each package, more details are provided by flopy coding on GitHub (https://github.com/modflowpy/flopy/tree/develop/flopy).

Table 2.2: MODFLOW and MT3DMS packages in ATES systems simulations (Zheng et al., 1999; Harbaugh, 2005)

Acronym	Program	Package
BAS	MODFLOW	Basic
WEL	MODFLOW	Well
LPF	MODFLOW	Layer-Property Flow Package
DIS	MODFLOW	Discretization File
PCG	MODFLOW	Preconditioned-Conjugate Gradient Package
OC	MODFLOW	Output Control Option
BTN	MT3DMS	Basic Transport Package
ADV	MT3DMS	Advection Package
DSP	MT3DMS	Dispersion Package
GCG	MT3DMS	Generalized Conjugate Gradient Solver Package
RCT	MT3DMS	Chemical Reaction Package
SSM	MT3DMS	Sink and Source Mixing Package

2.3 Application of OpenGeoSys in coupled groundwater flow and heat transport processes

Some simplified models based on OpenGeoSys have been developed in order to provide basic ideas. Beyer et al.(2016) applied OpenGeoSys in a hypothetical high-temperature medium scale ATES system to investigate the effect of convective heat transport on the storage efficiency and found that higher injection temperature leads to a decreasing of thermal recovery. A 3-D flow and heat transport simulation with inclined faults is implemented by Cherubini et al.(2013) to assess the impact of faults on fluid and heat transfer.

OpenGeoSys has been applied in many case studies as well. Meng et al.(2018) constructed a 2-D ground source heat pump(GSHP) system model considering both thermal convection and conduction in a Germany case study. Giordano et al.(2015) utilized OGS in a model of the borehole thermal energy storage (BTES) system using an array of borehole heat exchangers (BHEs), focusing on how changing water content as well as advective flow affect the heat propagation within a porous medium. Nordbeck et al.(2017) proposed that the numerical cooling be-

haviour could be fitted closely to the experimentally measured data after adapting the insulation parameters in a scalable modular geothermal heat storage system. Kong et al.(2017) developed a procedure in optimizing the distance between production and re-injection wells for the exploitation of geothermal reservoirs. OpenGeoSys has also been applied in an estimation of the geothermal potential in Nevado Del Ruiz (NDR) volcano (Colombia) (Vélez et al., 2018). Volpi et al.(2018) built an hydrothermal model to recreate the convetice behaviour of the Castel Giorgio-Torre Alfina (CG-TA, central Italy) reservior and simulated the exploitation of the geothermal filed.

2.4 Analytical Solutions

In this section, analytical solutions for groundwater flow process and the heat transport process are described.

2.4.1 Groundwater Flow Steady-state Analytical Solution

Both water injection and extraction at wells lead to the change in the head in the aquifer. The steady-state exact solution of the hydraulic head in the aquifer can be computed by equation 2.8.

$$h(x,y) = \sum \frac{Q_i}{2\pi K b} \ln(r_i) + C \tag{2.8}$$

and then

$$r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2} \tag{2.9}$$

with Q_i the flow rate at well i [m^3/d], K the hydraulic conductivity [m/d], b the aquifer thickness [m], r_i the distance between the point (x, y) where the head is computed and the well i [m] (x_i, y_i) and C is the constant value based on the reference head at a specific distance [m]. A positive flow rate indicates an extraction while an injection is represented by a negative value.

2.4.2 Heat Transport Analytical Solutions for Different Models

2.4.2.1 Sand Model

Consider a two dimensional flow in a confined aquifer. The water and the aquifer are approximated as incompressable. A well starts injecting water with a temperature $T_{injected}$ at 16 ^{o}C and a flow rate $Q_{injected}$ (storage volume = 250,000 m^3) at time $t = 0$. The thickness of the aquifer (b) is 30 m and there is no background flow. A modified analytical solution for the radial heat transport simulations from an injection well in transient conditions, introduced by Guimerà et al.(Guimerà et al., 2007), which assumes a continuous line-source and no background groundwater flow is described as follows:

$$\frac{\Theta_{(x,y,t)}}{\Theta_{injected}} = \frac{1}{2}\text{erfc}\left\{\frac{r^2 - r^{*2}}{2[(\frac{4}{3}\alpha)(r^*)^3 + (\frac{k_{T,bulk}}{A_T C_{P,bulk}})(r^*)^4]^{\frac{1}{2}}}\right\} \tag{2.10}$$

with:

$$C_{P,bulk} = \theta c_{P,f}\rho_f + (1-\theta)c_{P,s}\rho_s \tag{2.11}$$

$$\Theta = T - T_{ambient} \tag{2.12}$$

$$r^* = (2A_T t)^{\frac{1}{2}} \tag{2.13}$$

$$A_T = \frac{1}{R}\frac{Q_{injected}}{2\pi\theta b} \tag{2.14}$$

$$R = \frac{C_{P,bulk}}{\theta\rho_f c_{P,f}} \tag{2.15}$$

where $T_{ambient}$ is the ambient temperature [^{o}C], r is the distance to the wellbore and r^* represents the frontal position which is defined as a hypothetical thermal front position when neglecting the dispersion and diffusion phenomenon [m] and A_T denotes the retarded velocity.

2.4.2.2 Clay-sand-clay Model

In this model, the aquifer is confined by two clay layers which are approximated as impermeable. The model is introduced by Lin et al. (Lin et al., 2019), assuming

an aquifer is homogeneous and isotropic which is confined by two impermeable aquitards with no water flowing inside. The model layout is depicted in figure 2.2.

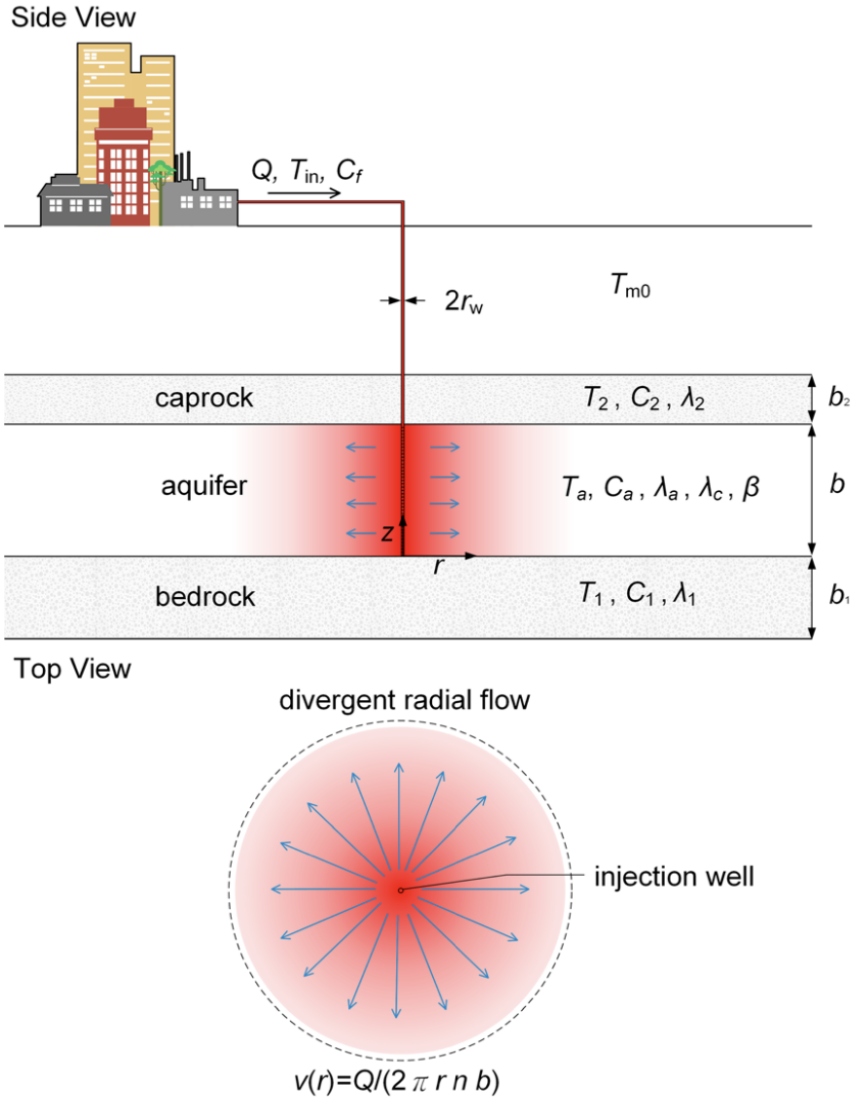


Figure 2.2: Clay-sand-clay model layout.
(Lin et al., 2019)

Identically, the settings are the same to the Sand model apart from the aquitards part. In this analytical model, thermal conduction, dispersion and convection contribute to the heat transfer in the aquifer while only thermal conduction is taken

into account in aquitards. Besides, the top boundary for the upper aquitard and the bottom boundary for the lower aquitard are set to a constant temperature condition of 12 ^oC. Separate equations are presented for the aquifer part and the aquitard part.

$$\text{aquifer} : \bar{\Theta}_a = \frac{\Theta_{injected}}{p} \frac{e^{(r_w - r)\sqrt{\Lambda}}(r + \alpha Pe)^{Pe}(r_w + \alpha Pe)^{-Pe} U(\gamma, \eta, \xi(r))}{U(\gamma, \eta, \xi(r_w))} \tag{2.16}$$

$$\text{bedrock} : \bar{\Theta}_1 = \bar{\Theta}_a \frac{e^{2\phi_1 b_1 + \phi_1 z} - e^{-\phi_1 z}}{e^{2\phi_1 b_1} - 1} \tag{2.17}$$

$$\text{caprock} : \bar{\Theta}_2 = \bar{\Theta}_a \frac{e^{\phi_2(z-b)} - e^{(2\phi_2 b_2 + \phi_2 b - \phi_2 z)}}{1 - e^{2\phi_2 b_2}} \tag{2.18}$$

with

$$\gamma = \frac{(1 - Pe - \alpha Pe\sqrt{\Lambda})}{2} \tag{2.19}$$

$$\eta = 1 + Pe \tag{2.20}$$

$$\xi(r) = 2\sqrt{\Lambda}(r + \alpha Pe) \tag{2.21}$$

$$\Lambda = \frac{(\rho_f \theta c_{P,f} + \rho_s(1-\theta)c_{P,s})p}{k_{T,bulk,aq}} + \frac{k_{T,bulk,bed}\phi_1}{coth(b_1\phi_1)bk_{T,bulk,aq}} + \frac{k_{T,bulk,cap}\phi_2}{coth(b_2\phi_2)bk_{T,bulk,aq}} \tag{2.22}$$

$$\phi = \sqrt{(\rho_f \theta c_{P,f} + \rho_s(1-\theta)c_{P,s})pk_{T,bulk,**}} \tag{2.23}$$

where b is the aquifer thickness[m]; b_* is the aquitard thickness[m]; p is Laplace parameter; $U()$ is the confluent hyper-geometric function; the over-bar represents the function in the Laplace domain; Pe is the Peclet number which is defined as $\rho_f c_{P,f} Q/(2\pi b \lambda_c)$.

Based on the equations above, the temperature solution can be expressed as:

$$T_a = T_{ambient} + \mathcal{L}^{-1}\left\{\bar{\Theta}_a\right\} \tag{2.24}$$

$$T_1 = T_{ambient} + \mathcal{L}^{-1}\left\{\bar{\Theta}_1\right\} \tag{2.25}$$

$$T_2 = T_{ambient} + \mathcal{L}^{-1}\left\{\bar{\Theta}_2\right\} \tag{2.26}$$

where $\mathcal{L}^{-1}$ is the inverse Laplace transform operator. The time domain for equations 2.24, 2.25 and 2.26 can be inverted by the numerical inverse Laplace trans-

form algorithm of 'The de Hoog, Knight, and Stokes' which is the most robust method compared to 'Talbot' and 'Stehfest' (Kuhlman, 2013). According to the equations described above, the function that computes the temperature in the aquifer at a distance of r to the wellbore is defined as fp(r, p). The inverse Laplace transform operator in equation 2.24 is inverted by calling invertlaplace function from mpmath and specifying the time and the algorithm 'deHoog'. Based on the defined function ft(r, t), the temperature at distance r at time t is calculated.

```
from mpmath import *
def ft(r,t):
    Tambient = 12
    ap = lambda p: fp(r,p)
    return invertlaplace(ap,t,method='deHoog')
    + Tambient
```

Similarly, the temperature in the aquitards is computed according to the defined ft_aqt function by specifying the distance r, the z coordination and the time t. It is worth noting that the datum line for z coordination is at the bottom of the aquifer in this model.

```
def ft_aqt(r,z,t):
    Tambient = 12
    ap = lambda p: fp_aqt(r,z,p)
    return invertlaplace(ap,t,method='deHoog')
    + Tambient
```

2.5 Numerical Methods

The description of the laws of physics for space and time-dependent problems are usually expressed by partial differential equations (PDEs) (COSMOL, n.d.). For the vast majority geometries and problems, these PDEs cannot be solved directly by analytical methods. Instead, an approximation of equations can be constructed, typically based upon different types of discritizations. These discretization meth-

ods approximate the PDEs with numerical model equations, which can be solved using numerical methods. The finite element method applied in OGS and finite difference method applied in SEAWAT are both used to compute approximations to the real solution to the PDEs. MT3DMS includes several solution options that treat the dispersion, sink/source and reaction terms in exactly the same fashion, using the block-centered finite-difference method, either explicitly or implicitly (Zheng et al., 1999). Once the GCG solver package is included in the simulation, the user is telling MT3DMS to do the following (Christopher J. Neville, n.d.):

1) All terms in the governing equation are represented with implicit-in-time weighted finite-difference approximations when the FD scheme is used.

2) The implicit-in-time weighted finite-difference approximations is applied when the particle-based methods and the TVD method are used. However, the time steps for TVD will still be subject to a time constraint-only the advection term.

Finite Element Method (OGS)

Within OGS environment, Crank-Nicolson method is used to approximate time derivative for the heat equation which is an implicit finite difference method and the FEM is employed for spatial discritization(Bottcher N, 2015). The solution for a continuum problem by FEM is approximated by the following steps: discretize the continuum, select the interpolation or shape functions, form element equations, assemble the element equations to obtain a system of simultaneous equations, solve the system of equations (Lewis et al., 2004). The solution region is divided into non-overlapping elements with each element formed by connection of a certain number of nodes as shown in figure 2.3. The functions employed to represent the nature of the solution within each element are called shape functions, or interpolating functions, or basis functions which are usually in polynomial type (Lewis et al., 2004). In this study, the hexahedron elements are applied. The bilinear interpolation function for the hexahedron element is

$$T = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 z + \alpha_5 xy + \alpha_6 yz + \alpha_7 zx + \alpha_8 xyz \tag{2.27}$$

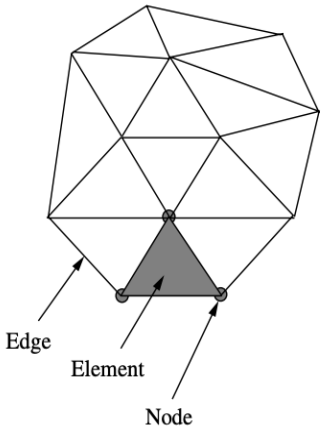


Figure 2.3: Typical finite element mesh components.
(Lewis et al., 2004)

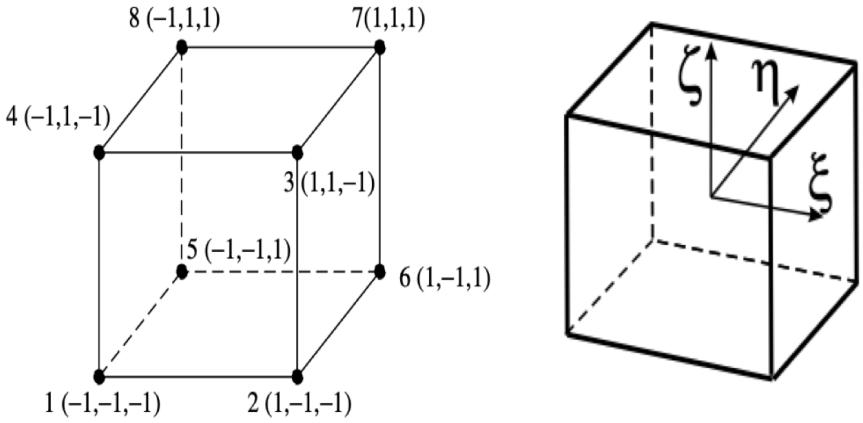


Figure 2.4: Three-dimensional element: hexahedron and its representation in the local coordinate system.
(Lewis et al., 2004)

which can be written as

$$T = \sum_{i=1}^{8} N_i T_i \tag{2.28}$$

where

$$N_i = \frac{1}{8}(1 + \xi\xi_i)(1 + \eta\eta_i)(1 + \zeta\zeta_i) \tag{2.29}$$

where ξ_i, η_i and ζ_i are the local coordinates (Lewis et al., 2004).
The next step is to determine the elements characteristics which is the relation between the nodal unknowns and the corresponding loads/forces in the form of the matrix equation, namely,

$$[K]\{T\} = \{f\} \tag{2.30}$$

where where $[K]$ is the thermal stiffness matrix, $\{T\}$ is the vector of unknown temperatures and $\{f\}$ is the thermal load, or forcing vector (Lewis et al., 2004). To convert the continuum partial differential equations (PDEs) as well as auxiliary conditions into a discrete system of algebraic equations, the Garlerkin method of weighted residuals is used to provide a weak formulation of PDEs (Bottcher N, 2015). To solve the algebraic equations, a linear solver called biconjugate gradient stabilized method (SpBICGSTAB) with Jacobi preconditioner is used.

Finite Difference Method (SEAWAT)

The general heat transport equation can be written as,

$$R\theta\frac{\partial T}{\partial t} = -\frac{\partial}{\partial x}(\theta v_x T) - \frac{\partial}{\partial y}(\theta v_y T) - \frac{\partial}{\partial z}(\theta v_z T) + L(T) \tag{2.31}$$

where R is the retardation factor and $L(T)$ is the non-advection terms (Zheng et al., 1999). The first partial derivatives representing the three components of the advection term at any finite difference cell can be approximated by the temperature values at the cell interfaces, see figure 2.5, as given below:

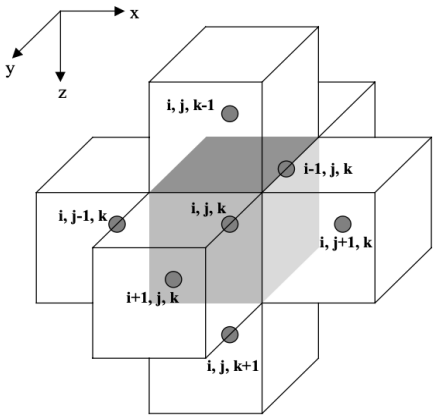


Figure 2.5: The index system used for the finite-difference grid. (Zheng et al., 1999)

$$\begin{aligned} &\frac{\partial}{\partial x}\left(\theta v_x T\right)+\frac{\partial}{\partial y}\left(\theta v_y T\right)+\frac{\partial}{\partial z}\left(\theta v_z T\right) \\ =&\frac{q_{x(i,j+1/2,k)}T_{i,j+1/2,k}-q_{x(i,j-1/2,k)}T_{i,j-1/2,k}}{\Delta x_j} \\ &+\frac{q_{y(i+1/2,j,k)}T_{i+1/2,j,k}-q_{y(i-1/2,j,k)}T_{i-1/2,j,k}}{\Delta y_i} \\ &+\frac{q_{z(i,j,k+1/2)}T_{i,j,k+1/2}-q_{x(i,j,k-1/2)}T_{i,j,k-1/2}}{\Delta z_k} \end{aligned} \tag{2.32}$$

where Δx_j , Δy_i, Δz_k are the dimensions of cell (i, j, k) in the x, y, and z directions, respectively; and $j + 1/2$, $i + 1/2$, and $k + 1/2$ denote the cell interfaces normal to the x, y, and z directions, respectively. The way to determine the interface temperature is what distinguishes different solution techniques.

In this study, an upstream weighting finite-difference method is applied where the interface temperature between two neighbouring nodes in a particular direction is equal to the temperature at the upstream node along the same direction as the example shown in equation 2.33.

$$T_{i,j-1/2,k}=\begin{cases} T_{i,j-1,k}, \text{if } q_{x(i,j-1/2,k)}>0 \\ T_{i,j,k}, \text{if } q_{x(i,j-1/2,k)}<0 \end{cases} \tag{2.33}$$

The upstream weighting scheme results in an oscillation-free solutions (Zheng et al., 1999). However, the solution for the advection that is only accurate to the first order leads to significant numerical dispersion when applied to the advection-dominated problems, since the truncation error resulting from the advection solution is of the same order and could overwhelm the second-derivative physical dispersion term (Bennett et al., 1995).

Third-order TVD Method (SEAWAT)

The TVD scheme utilized in MT3DMS for solving the advection term is based on the ULTIMATE algorithm (Universal Limiter for Transient Interpolation Modeling of the Advective Transport Equations) (Leonard-Barton & Deschamps, 1988). Within the ULTIMATE scheme, the interface temperatures are determined through a third-order polynomial interpolation of nodal temperatures, supplemented by a universal flux limiting procedure to minimize un-physical oscillations which may occur if sharp temperature fronts are involved (Zheng et al., 1999). The basic idea of the ULTIMATE scheme in one dimension are elaborated below. Suppose

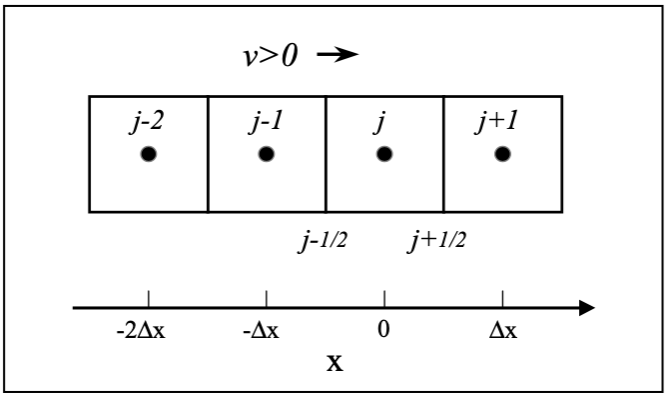


Figure 2.6: Illustration of the nodal points involved in the ULTIMATE scheme. (Zheng et al., 1999)

a finite difference grid with regular intervals superimposed by a local coordinate system with node j as the origin as depicted in figure 2.6. The velocity v is assume to be positive from left to right. The temperature at the node $j(x = 0)$ at time level

$n+1$ due to advection alone can be directly written as

$$T_j^{n+1} = T(0, \Delta t) = T(-v\Delta t, 0) \tag{2.34}$$

where Δt is the transport step size between the old time level $n(t = 0)$ and the new time level $n+1(t = \Delta t)$ (Zheng et al., 1999). As $T(-v\Delta t, 0)$ at the old time level does not coincide with a nodal point, the interpolation from the nearby nodes temperatures is required to determine this. A general third-order polynomial formula can be used for this purpose (Zheng et al., 1999), which, in 1-D form, can be written as

$$T(x, 0) = a + bx + cx^2 + dx^3 \tag{2.35}$$

where a, b, c, d are coefficients that can be related to the nodal temperatures by noting that

$$\begin{aligned} T(0, 0) &= T_j, \text{at } x = 0 \\ T(-2\Delta x, 0) &= T_{j-2}, \text{at } x = -2\Delta x \\ T(-\Delta x, 0) &= T_{j-1}, \text{at } x = -\Delta x \\ T(\Delta x, 0) &= T_{j+1}, \text{at } x = \Delta x \end{aligned} \tag{2.36}$$

The coefficients can be determined by substituting the equation 2.36 into 2.35. The solution for equation 2.34 can be obtained as

$$\begin{aligned} T_j^{n+1} &= T(0, \Delta t) = T(-v\Delta t, 0) = a + b(-v\Delta t) + c(-v\Delta t)^2 + d(-v\Delta t)^3 \\ &= T_j^n - C_r \left(\frac{T_{j+1}^n}{3} + \frac{T_j^n}{2} - T_{j-1}^n + \frac{T_{j-2}^n}{6} \right) + C_r^2 \left(\frac{T_{j+1}^n - 2T_j^n + T_{j-1}^n}{2} \right) \\ &- C_r^3 \left(\frac{T_{j+1}^n - 3T_j^n + 3T_{j-1}^n - T_{j-2}^n}{6} \right) \end{aligned} \tag{2.37}$$

where $C_r = v\Delta t/\Delta x$. Compared with the FD scheme in terms of the interface temperatures,

$$T_j^{n+1} = T_j^n - C_r \left(T_{j+1/2}^n - T_{j-1/2}^n \right) \tag{2.38}$$

the interface temperatures of ULTIMATE scheme in one dimension are determined by

$$T_{j+1/2} = \frac{T_{j+1} + T_j}{2} + C_r \frac{T_{j+1} - T_j}{2} - (1 - C_r^2) \frac{T_{j+1} - T_j + T_{j-1}}{6} \tag{2.39}$$

and

$$T_{j-1/2} = \frac{T_j + T_{j-1}}{2} + C_r \frac{T_j - T_{j-1}}{2} - (1 - C_r^2) \frac{T_j - T_{j-1} + T_{j-2}}{6} \tag{2.40}$$

where the first term corresponds to the standard FD solution with the central-in-space weighting, the second term is referred as the GRADIENT term and the third term as the CURVATURE term (Zheng et al., 1999). The basic idea remains the same for two-dimension and three-dimension. TVD scheme is mass conservative without excessive numerical dispersion but requires more computational effort than FD scheme. In addition, the interface temperatures computed by equation 2.39 and 2.40 can lead to nonphysical oscillations if sharp temperature fronts are involved. To circumvent this problem, an universal flux limiter is employed by ULTIMATE scheme to adjust the interface temperatures after they are determined by the interpolation (Leonard & Niknafs, 1990).

Figure 2.7 depicts an example meshgrid in order to explain the difference between OpenGeoSys and SEAWAT in the calculation process. As described in the numerical methods above, SEAWAT calculates the properties at the center of the cell (black dots) while OpenGeoSys calculates the properties for nodes (red dots). Thus, OpenGeoSys is able to calculate the head and temperature at nodes at the boundary of two layers.

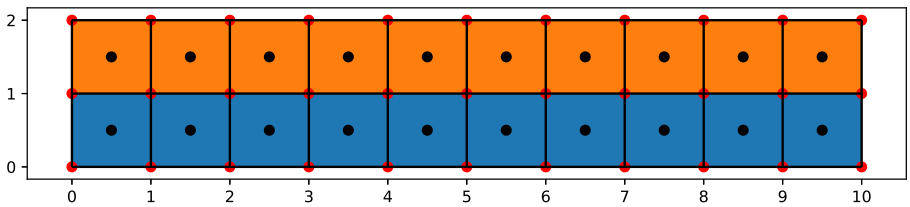


Figure 2.7: Example meshgrid.

2.6 Scenarios

2.6.1 Sand model

Model Set-up

Consider the problem in section 2.4.2.1, there is only the aquifer part in the Sand model. To make the energy calculation easy, a regular-grid mesh is applied for both OGS and SEAWAT, see figure 2.8. As the thermal radius is shorter than

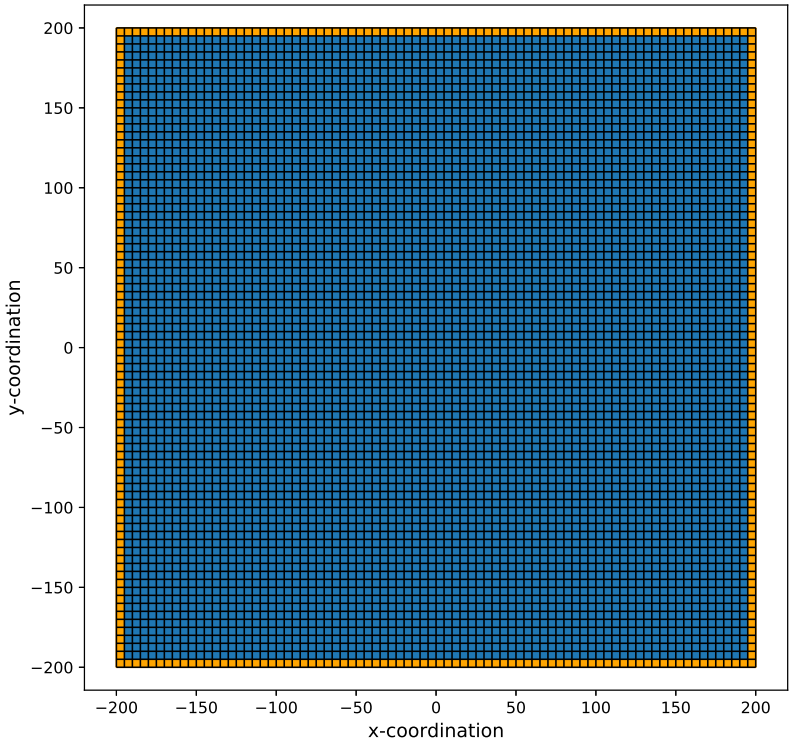


Figure 2.8: Sand model mesh top view.

100 meters, the mesh domain is set to be 400×400 meters with a thickness of 30 meters which is large enough to make sure that the boundary temperature is not affected by the injection of water. Besides, the hydraulic conductivity and

dispersivity are regarded as isotropic. In addition, the head and temperature of the orange cells in the figure 2.8 are set to be 0 meter and 12 ^oC that is constant during the simulation to expedite the convergence. The well is implemented at the middle of the mesh with injecting a constant flow (250,000 m^3 per 180 days) in a constant temperature (16 ^{o}C) for half a year (180 days). A cell size of 5×5 meters and a time step of 5 days are applied in OGS and SEAWAT (both FD and TVD scheme).

Model Parameters

The parameters utilized in this model is shown in table 2.3. And the converted parameters for SEAWAT are computed as well.

Table 2.3: Model parameters applied in the Sand model.

Parameter	**Symbol**	**Value**
Hydraulic conductivity [m/d]	$K_{h,s}$	12
Porosity[-]	θ	0.3
Fluid phase thermal conductivity[W/m/K]	$k_{T,f}$	0.58
Solid phase thermal conductivity[W/m/K]	$k_{T,s}$	3
Fluid phase specific heat capacity[J/kg/K]	$c_{P,f}$	4183
Solid phase specific heat capacity[J/kg/K]	$c_{P,s}$	710
Fluid density[kg/m^3]	ρ_f	1000
Solid density[kg/m^3]	ρ_s	2640
Longitudinal dispersivity[m]	α_L	0.5
Calculated parameter for SEAWAT	**Symbol**	**Value**
Bulk density[kg/m^3]	ρ_b	1848
Thermal conductivity for aquifer[W/m/K]	$k_{T,aq}$	2.274
Thermal distribution coefficient[kg/m^3]	k_{dist}	0.00017
Diffusion coefficient for aquifer[m^3/kg]	$k_{dtemp,aq}$	0.0131

2.6.2 Clay-sand-clay model

Model Set-up

Similarly, the regular-grid mesh is utilized in this model as well. As the mesh domain is symmetric, half of the domain is generated in SEAWAT and OGS sim-

ulations environments to save the computational time. In the aquifer part, the hydraulic conductivity stays the same (12 m/d) while in the aquitard part, the hydraulic conductivity is set to be 0.001 m/d to simulate the impermeable aquitard condition. Furthermore, the dispersivity in the aquifer is set to be 0 as only diffusion contributes to the heat transport in the clay layer in the exact solution. Beside the boundary condition mentioned in the sand model, a temperature boundary condition is added at the most top clay layer.

One-layer-aquifer model

Initially, the aquifer layer is regarded as one layer in the meshgrid because the analytical solution for the aquifer is independent to the depth. Since the aquifer thickness is 30 meters and the clay layer is set to be 20 meters which is thick enough to avoid the influence of the top boundary condition, there are 21 layers in total with 20 layers for the aquitard and 1 layer (half of the aquifer thickness) for the aquifer, see figure 2.9. The aquifer dispersivity is regarded as isotropic as only one dispersivity value is mentioned in the analytical solution.

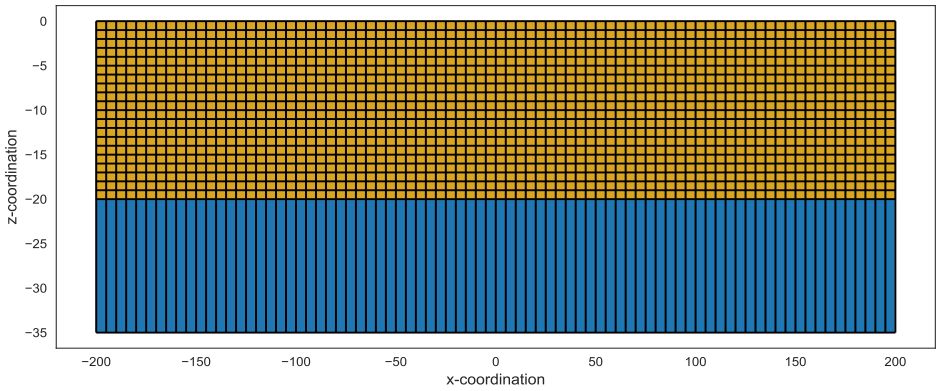


Figure 2.9: Clay-sand-clay one-layer aquifer model mesh cross-section view.

Multi-layers-aquifer model

Considering that there is still a temperature difference at different depth in the aquifer, the meshgrid for aquifer is set to be 1 meter thick to make sure the layer thickness is the same for the whole domain. Thus, there are 35 layers in total in the multi-layer-aquifer model with 20 layers for aquitard and 15 layers for aquifer, see figure 2.10.

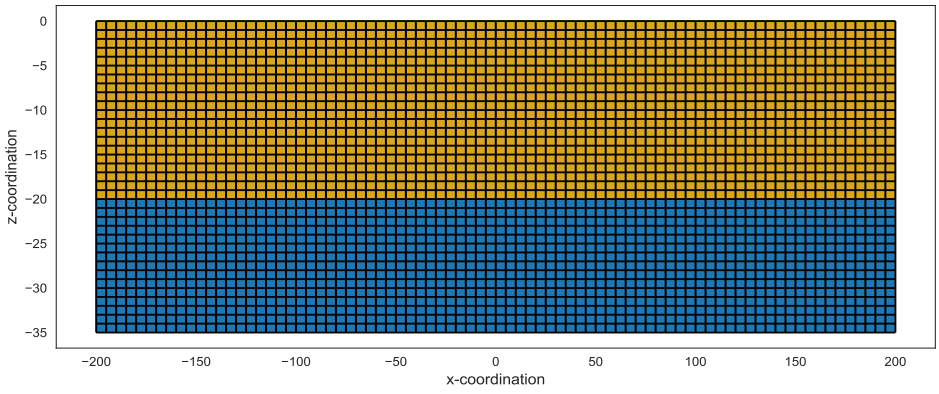


Figure 2.10: Clay-sand-clay multi-layer aquifer model mesh cross-section view.

Model Parameters

Since the aquitards are taken into account in the Clay-sand-clay model, the corresponding parameters are included apart from that of aquifer, see table 2.4. Due to that the analytical solution is not able to converge with the storage volume of 250,000m^3/year, a slightly adjusted volume of 220,000m^3/year is used in this model.

2.7 Numerical Model Implementations

The effects of temperature on viscosity and density are not taken into account as the temperature variation is not large in this study. This section mainly introduces comparison of the method of applying groundwater flow and heat transport in OGS and SEAWAT. In order to elaborate the model implementations in both OpenGeoSys and SEAWAT environments, the sand model is taken as an example.

Table 2.4: Model parameters applied in the Clay-sand-clay model.

Parameter	**Symbol**	**Value**
Hydraulic conductivity (aquifer) [m/d]	$K_{h,s}$	12
Hydraulic conductivity (aquitard) [m/d]	$K_{h,c}$	0.001
Porosity[-]	θ	0.3
Fluid phase thermal conductivity[W/m/K]	$k_{T,f}$	0.58
Solid phase thermal conductivity(aquifer)[W/m/K]	$K_{T,s}$	3
Solid phase thermal conductivity(aquitard)[W/m/K]	$K_{T,c}$	1
Fluid phase specific heat capacity[J/kg/K]	$c_{P,f}$	4183
Solid phase specific heat capacity[J/kg/K]	$c_{P,s}$	710
Fluid density[kg/m^3]	ρ_f	1000
Solid density[kg/m^3]	ρ_s	2640
Longitudinal Dispersivity[m]	α_L	0.5
Aquifer transverse horizontal dispersivity[m]	α_h	0.5
Aquifer transverse vertical dispersivity[m]	α_v	0.5
Calculated parameter for SEAWAT	**Symbol**	**Value**
Bulk density[kg/m^3]	ρ_b	1848
Thermal conductivity for aquifer[W/m/K]	$K_{T,aq}$	2.274
Thermal conductivity for aquitard[W/m/K]	$K_{T,aqt}$	0.874
Thermal distribution coefficient[kg/m^3]	K_{dist}	0.00017
Diffusion coefficient for aquifer[m^3/kg]	$K_{dtemp,aq}$	0.0131
Diffusion coefficient for aquitard[m^3/kg]	$K_{dtemp,aqt}$	0.0050

2.7.1 OGS

Spatial and Temporal Discretizing

Discretizing the spatial and temporal domain to a numerical mesh is of great importance before solving the differential equations for the whole domain when applying numerical solving methods. In this study, the simplified meshgrid with regular cell is applied. Since there are different models in this research, a brief introduction for generating the meshgrid for one layer is described below.

The PCS object is specified individually for different processes. The default of the TIM_TYPE is transient, the steady state can be set by specifying STEADY in the TIM_TYPE.

```
model.pcs.add_block(
    main_key="PROCESS",
    PCS_TYPE="GROUNDWATER_FLOW",
    #specified process
    TIM_TYPE = 'STEADY',
    #specified state)
)
model.pcs.add_block(
    main_key="PROCESS",
    PCS_TYPE="HEAT_TRANSPORT",
    #specified process)
)
```

In OGS, the meshgrid can be defined by many ways such as a range of internal methods or providing close integration with the pygmsh package which is a python tool of mesh generation. More details of internal methods can be found in https://github.com/GeoStat-Framework/ogs5py/blob/master/ogs5py/fileclasses/msh/generator.py. In this study, 3-D meshgrid with regular cell size is required.

```
from ogs5py import MSH
mesh = MSH()
```

```
mesh.generate (
    "rectangular",
     #Generate a rectangular grid in 2D or 3D.
     dim=3,
     #Dimension of the resulting mesh.
     mesh_origin=(-200.0, -200.0, -30.0),
     #Origin coordinate of the mesh.
     element_no=(80, 80, 1),
     #Number of elements in each direction.
     element_size=(5.0, 5.0, 30.0),
     #Size of an element in each direction.
)
```

Here, a meshgrid includes 6400 elements (cubes with size of $5\times5\times30$) and 13122 nodes is created. In the created mesh file, the coordinate of each node is included as well as the elements information. In this example, there is only one group of mesh, thus, the generated mesh can be directly applied in the numerical simulation. However, the meshgrid should be created separately if there are more than one group of mesh. It is worth noting that objects of OGS are associated with the material ID, especially fluid properties object, medium properties object and solid properties object. For instance, the hydraulic properties and thermal properties are different for different kinds of soils. The default material ID is 0 and it can be defined by using the command:

```
mesh.MATERIAL_ID = 0
#define the material id for the mesh
```

After generating each mesh group, the mesh can be saved by:

```
mesh.save("")
```

In order to acquire a whole domain, the different groups of mesh can be merged by:

```
mesh.combine_mesh('filename.msh')
#combine the current mesh with another mesh
```

As for the temporal discretizing, OpenGeoSys provides the TIM object to define the time step which is shown below:

```
model.tim.add_block(
        main_key='TIME_STEPPING',
        PCS_TYPE='GROUNDWATER_FLOW',
        TIME_STEPS=[36, \
        5$$\times$24$$\times$3600],
        TIME_END=1e99,
        TIME_START=0.0,
)
```

```
model.tim.add_block(
        main_key='TIME_STEPPING',
        PCS_TYPE='HEAT_TRANSPORT',
        TIME_STEPS=[36, \
        5*24*3600],
        TIME_END=1e99,
        TIME_START=0.0,
)
```

In this example, two processes are included. Since the groundwater flow process is set to be steady-state, only the temporal discretizing for heat transport process is defined. There are 36 time steps in total with each step of 5 days. If the value of TIME_END is longer than the total time, the simulation would end at the total time. Otherwise, the simulation would stop at the value of TIME_END.

Geometry Definition

```
import ogs5py as OGS
```

```
model = OGS(task_root="",
            task_id="",
            output_dir="")
model.msh.read_file("")
model.gli.add_polyline("well", [[0, 0, 0], \
                        [0, 0, -30]])
model.gli.add_polyline('Layer1', [[0, 0, 0],\
                        [100, 0, 0]])
model.gli.add_polyline('Layer2', [[0, 0, -30],\
                        [100, 0, -30]])
model.gli.add_polyline('F', [[-200, -200, 0],\
          [200, -200, 0], [200, -200, -30], \
          [-200, -200, -30], [-200, -200, 0]])
model.gli.add_polyline('B', [[-200, 200, 0], \
            [200, 200, 0], [200, 200, -30], \
           [-200, 200, -30], [-200, 200, 0]])
model.gli.add_polyline('L', [[-200, -200, 0],\
           [-200, 200, 0], [-200, 200, -30],\
         [-200, -200, -30], [-200, -200, 0]])
model.gli.add_polyline('R', [[200, 200, 0], \
        [200, 200, -30], [200, -200, -30], \
              [200, -200, 0], [200, 200, 0]])
model.gli.add_surface('Front', ['F'])
model.gli.add_surface('Back', ['B'])
model.gli.add_surface('Left', ['L'])
model.gli.add_surface('Right', ['R'])

)
```

Initially, a file is created in order to save the settings and outputs by calling OGS and specifying the file name as well as the direction. OGS could directly read the mesh file if it is given, otherwise users have to create the mesh manually. Then, geometric objects are required to be specified for wells and surfaces since there

are sink/source terms and the Initial/Boundary conditions. As shown above, the first three polylines are specified which include one for well and two for observations. In addition, four polylines that compose to four boundary surfaces are defined as well as the four surfaces.

Initial/Boundary Condition

The initial head for the whole domain is set to be 0 and 12^{o}C for the initial temperature. In addition, there are boundary conditions for both head and temperature at the edges of domain.

OpenGeoSys provides IC and BC objects for defining the initial and boundary conditions by specifying the geometry type defined in GLI object.

```
model.ic.add_block(
    main_key = "INITIAL_CONDITION",
    PCS_TYPE = "GROUNDWATER_FLOW",
    PRIMARY_VARIABLE="HEAD",
    GEO_TYPE = ['DOMAIN'],
    DIS_TYPE = ["CONSTANT", 0],
)
model.ic.add_block(
    main_key = "INITIAL_CONDITION",
    PCS_TYPE = "HEAT_TRANSPORT",
    PRIMARY_VARIABLE="TEMPERATURE1",
    GEO_TYPE = ['DOMAIN'],
    DIS_TYPE = ["CONSTANT", 12],
)
for i in np.array(['Front', 'Back', \
                   'Left', 'Right']):
    model.bc.add_block(
    main_key = "BOUNDARY_CONDITION",
    PCS_TYPE = "GROUNDWATER_FLOW",
    PRIMARY_VARIABLE = "HEAD",
    GEO_TYPE = ["SURFACE", i],
```

```
22      DIS_TYPE = ["CONSTANT", 0],
23      )
24      model.bc.add_block(
25          main_key = "BOUNDARY_CONDITION",
26          PCS_TYPE = "HEAT_TRANSPORT",
27          PRIMARY_VARIABLE = "TEMPERATURE1",
28          GEO_TYPE = ["SURFACE", i],
29          DIS_TYPE = ["CONSTANT", 12],
30      )
```

Parameters setting

The mmp, mfp and msp objects specify the medium properties such as porosity, storage, hydraulic conductivity as well as the heat dispersion; the fluid properties such as fluid type (liquid or gas), density, specific heat capacity and the heat conductivity; the solid properties such as density, thermal capacity as well as the thermal conductivity. The first value in the bracket indicates the type of the parameters. In this study, all parameters are assumed to be stable for the whole simulation. Thus, value 1 is used indicating a steady variable value. In addition, the permeability_tensor in mmp object means hydraulic conductivity when applying in groundwater flow process and can be either isotropic and anisotropic. When specifying an anisotropic condition, three values are required. Besides, two values in heat dispersion in mmp object represent the longitudinal dispersivity and the transverse dispersivity, respectively.

```
1  model.mmp.add_block(
2      main_key='MEDIUM_PROPERTIES',
3      GEOMETRY_DIMENSION=3,
4      POROSITY=[1, value],
5      STORAGE=[1, value],
6      PERMEABILITY_TENSOR=['ISOTROPIC', value],
7      HEAT_DISPERSION=[1, value1, value2],
8  )
9  model.mfp.add_block(
```

```
10        main_key='FLUID_PROPERTIES',
11        FLUID_TYPE='LIQUID',
12        PCS_TYPE='HEAD',
13        DENSITY=[1, value],
14        SPECIFIC_HEAT_CAPACITY=[1, value],
15        HEAT_CONDUCTIVITY=[1, value],
16    )
17
18    model.msp.add_block(
19        main_key='SOLID_PROPERTIES',
20        DENSITY=[1, value],
21        THERMAL=[
22        ['CAPACITY:'],
23        [1, value],
24        ['CONDUCTIVITY:'],
25        [1, value],
26        ],
27    )
```

Source/Sink Term

As for the source/sink terms, there is a constant amount of water injected into the well. Within the block, the distribution types could be either CONSTANT or CONSTANT_NEUMANN. Once the CONSTANT is specified, the value is assigns the same value to every node belonging to the given geometric object ("warmwell") whereas CONSTANT_NEUMANN assigns the value multiplied by node area/node length to each node. Since the CONSTANT_NEUMANN is selected, the constant flow rate is required to be devided by the length of the well-bore.

```
1   model.st.add_block(   # source term
2       main_key='SOURCE_TERM',
3       PCS_TYPE="GROUNDWATER_FLOW",
4       PRIMARY_VARIABLE="HEAD",
```

```
    GEO_TYPE=["POLYLINE", "well"],
    DIS_TYPE=["CONSTANT_NEUMANN", flow rate/30],
    #the flow rate is in unit of m3/s
)
```

In OpenGeoSys, the temperature of the injected water is set to be the boundary condition as shown below:

```
model.bc.add_block(
    main_key = "BOUNDARY_CONDITION",
    PCS_TYPE = "HEAT_TRANSPORT",
    PRIMARY_VARIABLE="TEMPERATURE1",
    GEO_TYPE=["POLYLINE", 'well'],
    DIS_TYPE = ["CONSTANT", 16.0],
)
```

Numerical Solver

The details about the values in the linear solver can be found in https://ogs5-keywords.netlify.app/ogs/wiki/public/doc-auto/by_ext/num/s_linear_solver.

```
model.num.add_block(
    main_key='NUMERICS',
    PCS_TYPE="GROUNDWATER_FLOW",
    LINEAR_SOLVER=[2, 5, 1e-5, 1000,\
    1.0, 1, 2],
)
model.num.add_block(
    main_key='NUMERICS',
    PCS_TYPE="HEAT_TRANSPORT",
    LINEAR_SOLVER=[2, 5, 1e-7, 1000,\
    1.0, 1, 2],
)
```

After the steps described above, the model can be simulated by:

```
    model.write_input()
    model.run_model(ogs_exe =
    "/Users/liuhaoyue/Desktop/model/ogs")
```

2.7.2 SEAWAT

To compare the modelling implementation for different software, same model is applied in SEAWAT environment and described below.

Spatial and Temporal Discretizing

In the SEAWAT environment, the number of model layers, rows and columns are specified. In addition, the spacings along a row and a column are required. The height of a cell can be calculated by a subtraction of given top and bottom elevations. Both spatial and temporal discretizing can be defined within the ModflowDis package as shown below:

```
mswtf = flopy.seawat.Seawat(name, \
exe_name=swtexe_name, model_ws=dirs[0])
#define the output file for SEAWAT
dis = flopy.modflow.ModflowDis(
    nlay=1,
    #the number of layers
    nrow=80,
    #the number of rows
    ncol=80,
    #the number of columns
    delr=5,
    #the spacings along a row
    delc=5,
    #the spacings along a column
    top=0,
```

```
16      #the top location of the cell
17      botm=-30,
18      #the bottom location of the cell
19      perlen=5,
20      #length of the period
21      nstp=36,
22      #step of the period
23      steady=True,
24      )
```

The temporal discretizing for the heat transport is not required since MT3DMS would automatically adjust the time step.

Initial/Boundary Condition

In SEAWAT, the initial condition is specified by defining array (nlay, nrow, ncol) for head and temperature separately. Besides, in order to set the boundary conditions, SEAWAT provides the boundary array. The default value is 1 for the whole array which indicates that there is no boundary condition. By specifying the value -1 for a certain location, the boundary condition can be defined. The PEFF within Mt3dBtn package indicates the porosity in SEAWAT which would be specified in medium properties object in OGS.

```
1  head = np.zeros((nlay, nrow, ncol))
2  temp = np.ones((nlay, nrow, ncol)) * 12
3  ibound[:, 0, :] = -1
4  ibound[:, -1, :] = -1
5  ibound[:, :, 0] = -1
6  ibound[:, :, -1] = -1
7  ibound1[:, 0, :] = -1
8  ibound1[:, -1, :] = -1
9  ibound1[:, :, 0] = -1
10 ibound1[:, :, -1] = -1
11 bas = flopy.modflow.ModflowBas(mswtf, \
```

```
12                 ibound=ibound, strt=head)
13 btn = flopy.mt3dms.Mt3dBtn(mswtf, \
14 icbund=ibound1, prsity=PEFF, sconc=temp)
```

Parameters setting

The hydraulic conductivity is specified in ModflowLpf package. In addition, the dispersivity is defined within the Mt3dDsp package. The other thermal properties for fluid and solid are converted into bulk properties according to the porosity and then applied in SEAWAT. The calculation method is given in section 3.1.2.

```
1      lpf = mf.ModflowLpf(mswtf, hk=12, \
2      vka = 12, ss = ss)
3      #hk-horizontal hydraulic conductivity
4      #vka-vertical hydraulic conductivity
5      #ss-storge coefficient
6      dsp = flopy.Mt3dms.Mt3dDsp(mswtf, \
7      al=al, trpt=1, trpv=1, dmcoef=kT_aq)
8      #al-longitudinal dispersivity
9      #trpt-ratio of tranverse horizontal\
10     #dispersivity to the logitudinal one
11     #trpv-ratio of tranverse vertical\
12     #dispersivity to the logitudinal one
13     #dmcoef-effective molecular diffusion\
14     coefficient
15     rct = mt3.Mt3dRct(mswtf, isothm=1, \
16     ireact=0, igetsc=0, rhob=rho_b, \
17     sp1 = Kdist)
18     #rho_b-bulk density
19     #Kdist-sorption parameter
```

Source/Sink Term

Unlike OpenGeoSys, SEAWAT assigns both injected flow and the temperature as the source terms by specifying the location of the well. The well is located in the

center of the domain, thus the corresponding row and column are both 40.

```
ssm_data = {0:[[0, 40, 40, 16, \
itype['WEL']]]}
well_LRCQ_list = {0:[[0, 40, 40, \
flow rate]]}
#the flow rate is in unit of m3/d
wel = flopy.modflow.ModflowWel(mswtf, \
stress_period_data=well_LRCQ_list)
ssm = flopy.mt3dms.Mt3dSsm(mswtf, \
stress_period_data=ssm_data)
```

Numerical Solver

SEAWAT provides ModflowPcg and Mt3dGcg packages for specifying the numerical solving setting for groundwater flow and heat transport, respectively. The default error tolerance for ModflowPcg is 1e-5.

```
pcg = flopy.modflow.ModflowPcg(mswtf)
gcg = flopy.mt3dms.Mt3dGcg(mswtf, \
iter1=1000, mxiter=1, isolve=1, \
cclose=1e-7)
adv = mt3.Mt3dAdv(mswtf, mixelm=-1)
#-1 for TVD scheme and 0 for FD scheme
```

Similarly, the model can be simulated by:

```
mswtf.write_input()
m = mswtf.run_model()
```

2.8 Integration methods

Simpson's Rule

To calculate the extracted energy from the well, a numerical integration method

called Simpson's Rule that approximates definite integrals is utilized in this study. Simpson's Rule is corresponding to the 3-point Newton-Cotes quadrature rule (Simpson(Formula), 2001). The basic equation is given below which computes the integration of the curve between a and b by using the parabolas to approximate each part of the curve.

$$\int_a^b f(x)dx \approx \frac{b-a}{6}(f(a) + 4f(\frac{a+b}{2}) + f(b)) \tag{2.41}$$

In figure 2.11, the area is divided into n equal segments of width Δx where n has to be an even number. By substituting $f(x_8)$, $f(x_9)$ and $f(x_{10})$ into equation 2.41, the blue area is computed.

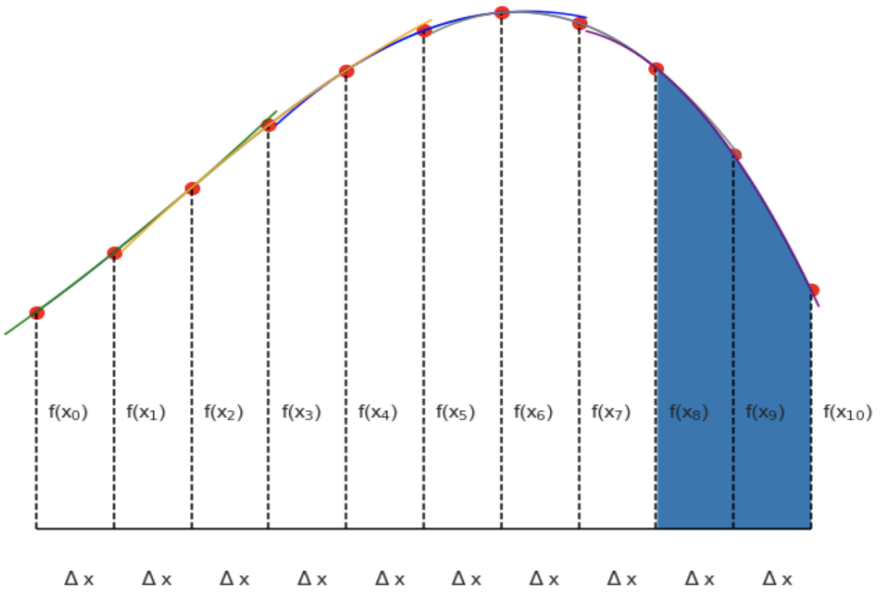


Figure 2.11: Main idea behind the Simpson's rule.

The approximate area is given by the following.

$$\int_a^b f(x)dx \approx \frac{\Delta x}{3}(f(x_0)+4f(x_1)+2f(x_2)+4f(x_3)+2f(x_4)+\cdots+4f(x_9)+f(x_{10})) \tag{2.42}$$

Gaussian Quadrature

Unlike the numerical solution, exact solutions are continuum. The analytical solution cannot be integrated or difficult to integrate analytically. Thus, some numerical integration methods are alternatives. In numerical analysis, a quadrature rule is an approximation of the definite integral of a function, usually stated as a weighted sum of function values at specified points within the domain of integration (Wikipedia contributors, 2020). An n-point Gaussian quadrature rule constructed to yield an exact result for polynomials of degree $2n-1$ or less by a suitable choice of the nodes x_i and weights w_i for i = 1, ..., n (Gauss, 1815) and the most common domain of the integration is taken as [-1, 1], so the rule is stated as

$$\int_{-1}^{1} f(x)dx \approx \sum_{i=1}^{n} w_i f(x_i), \tag{2.43}$$

which is exact for polynomials of degree $2n-1$ or less. The interval over [a, b] must be changed into an integral over [-1, 1] before applying the Gaussian quadrature rule and can be done in the following way (Wikipedia contributors, 2020):

$$\int_a^b f(x)dx = \frac{b-a}{2}\int_{-1}^{1} f(\frac{b-a}{2}\xi+\frac{a+b}{2})d\xi \approx \frac{b-a}{2}\sum_{i=1}^{n} w_i f(\frac{b-a}{2}\xi_i+\frac{a+b}{2}) \tag{2.44}$$

A function called scipy.integrate.quad is supported by Python to compute the definite integrate using a technique from the FORTRAN library QUADPACK (Scipy, 2020) based on the quadrature rule. Once given the function required to be integrated and the upper, lower limit of the integration, the integrated result can be calculated. The return value is a tuple, with the first element holding the estimated value of the integral and the second element holding an upper bound on the error .

Paraview Tools

Paraview is an open-source and multi-plate-form data analysis and visualisation software (Paraview, 2020). In this study, since the output files for the domain are in VTK format, Paraview is required to do the further analysis for the results to cal-

culate the thermal energy stored in the system. Initially, import the output file that includes all data in the Paraview. Secondly, use **POINT-DATA-TO-CELL-DATA** in the tools to transfer the data type. Then, compute the temperature deviation by **CALCULATOR**. Finally, **INTEGRATE VARIABLE** is utilized to integrate the temperature in the whole domain. According to the temperature integration, the energy in the system can be computed by multiplying the volumetric heat capacity.

Sum-up Method

This method aims to calculate the mean value for the cell based on the point-wise data. An example is given in figure 2.12 where only the visible points are presented. Each cell has the same size and each point represent one value. In order to integrate the value for the whole domain, the contributions for each node should be considered. For orange nodes, they only contribute one quarter to the whole domain and green nodes contribute half to the whole domain. Only the node in the center contribute to whole domain.

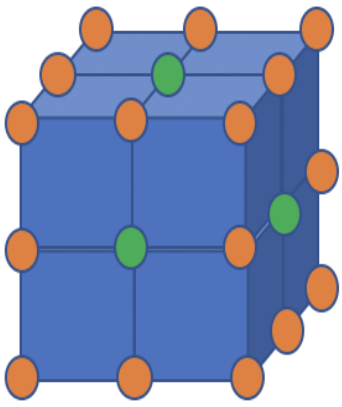


Figure 2.12: Sum-up method example.

2.9 Assessment Framework

The models performance is assessed based on several criteria which is listed below.

Thermal Recovery Efficiency

The thermal recovery efficiency is defined as the percentage of the thermal energy injected into the system that is extracted in the next season. The higher the thermal recovery efficiency, the better the ATES system performs.

$$\eta = \frac{E_{out}}{E_{in}} \tag{2.45}$$

Energy Balance Error

The energy balance error is computed by equation 2.46:

$$\epsilon = \frac{(E_{remained} + E_{out} - E_{in})}{E_{in}} 100 \tag{2.46}$$

with $E_{remained}$ the calculated energy remained in numerical models [J]; E_{out} the energy extracted from the aquifer during extraction period [J]; E_{in} the energy input. The energy is calculated by equations below:

$$E_{remained} = \sum_{i=1}^{n} T_i V_{bulk} C_{p,bulk} \tag{2.47}$$

$$E_{in} = V \rho_f c_{P,f} \Delta T \tag{2.48}$$

with n is the total number of cells [-], T_i the temperature at the cell i [^{o}C], V_{bulk} the volume of the cell [m^3] and V the injected amount of water into the well [m^3]. Since the output of OGS is point-wise, the sum-up method is needed to calculate the remained energy in OGS models. In addition, the extracted energy is calculated by Simpson's rule based on the extraction temperature of wells. The extraction temperature is of vital importance when analysing the system perfor-

mance in ATES systems since it determines the heating/cooling capacity of ATES systems.

Temperature Distribution

The temperature distribution is a straightforward way to compare the numerical solutions with the exact solution as it shows the matching extent, indicating which one performs better. The closer the numerical solution to the analytical solution, the better it performs.

Numerical Dispersion

For advection-dominated problem, numerical dispersion is a common error which is caused by truncation error and has the similar effect to the physical dispersion (Zheng et al., 1999). It becomes a serious problem once the physical dispersion is negligible and can be distinguished when the concentration front has a smearing effect where it is supposed to have a sharp change as depicted in figure 2.13.

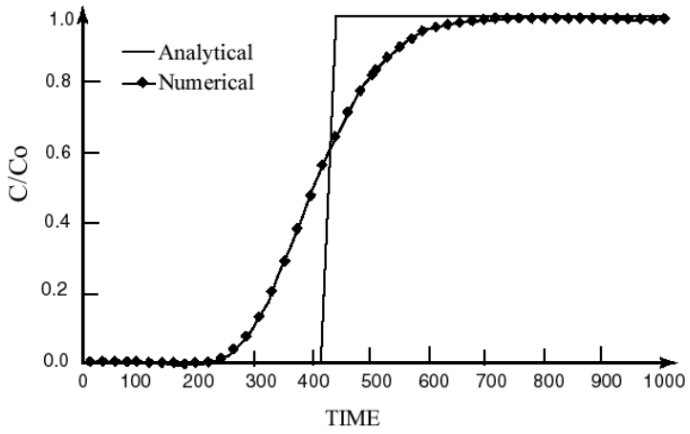


Figure 2.13: Illustration of numerical dispersion.
(Zheng et al., 1999)

3 | Results

3.1 Compare the performance of OGS and SEAWAT in one-well model with the analytical solution

As stated in section 2.1.2, the analytical solution for coupled processes is only feasible for one-well-injection model. In this study, two models described in 2.4.2 are generated respectively to compare the performances of OGS and SEAWAT.

3.1.1 Sand model

The head distribution around the well is plotted, see figure 3.1. The analytical solution for the head at the well is not included since there is no specific radius for the well in OGS simulation. Despite this, head distribution of three numerical models in steady-state matches the exact solution away from the well. Since SEAWAT applies finite difference method in simulating the groundwater flow process in cells regardless of the different advection options, the results are the same. While in OGS, a finite element solver is utilized and the well is represented by a polyline, resulting in a higher head at the well than that of SEAWAT.

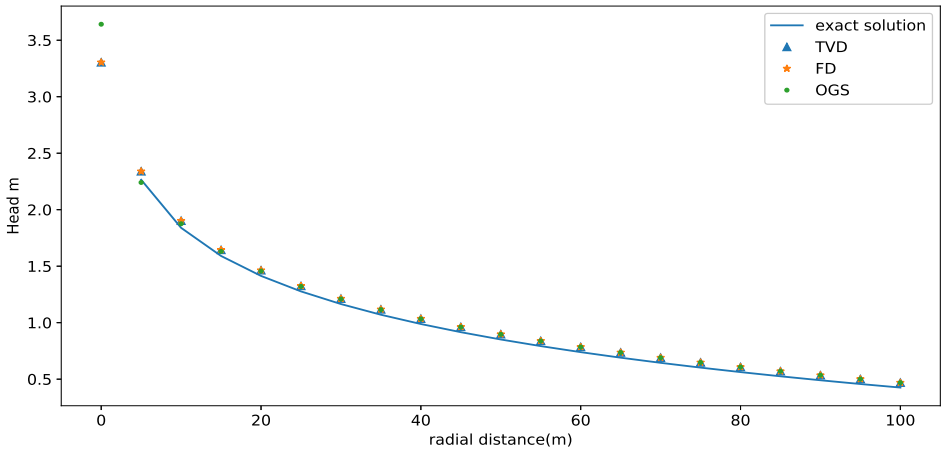


Figure 3.1: Head distribution along the radial direction at t=180 days.

Figure 3.2 depicts the temperature distribution in the radial direction. The exact solution is represented as the solid lines while the numerical solutions for OGS, FD and TVD are represented by dots, stars and triangles, respectively.

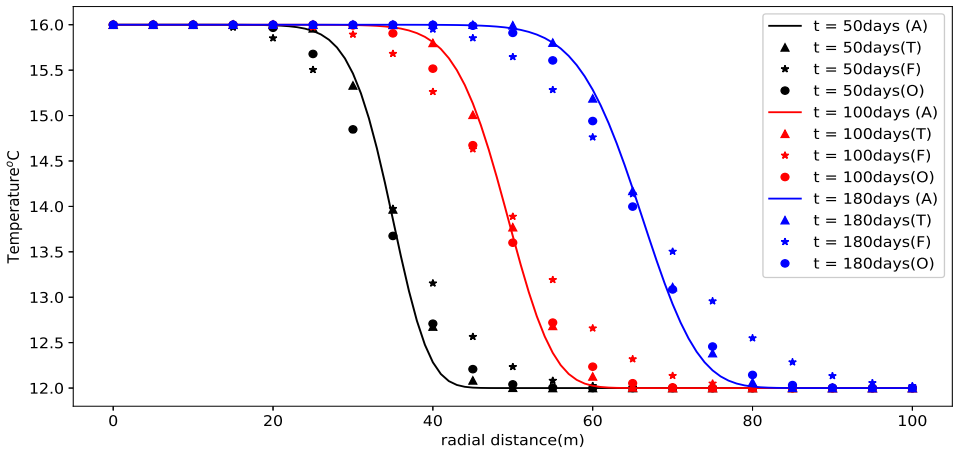


Figure 3.2: Temperature distribution along the radial direction of the analytical solution and numerical solutions at different time (black-50 days, red-100days, blue-180days); A(analytical), T(TVD), F(Finite Difference), O(OGS).

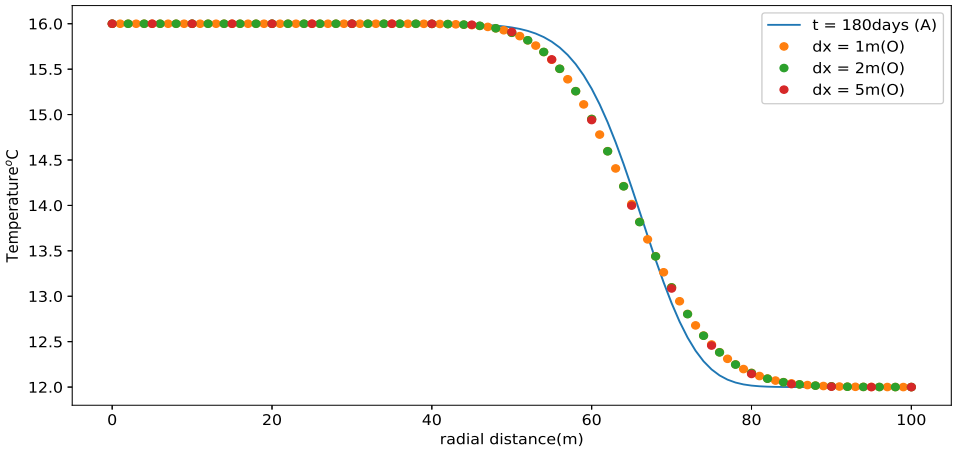
As shown in the figure, although TVD performs better among these solutions, neither of them shows a satisfying fitting extent. In the front part of the heat transport front, TVD solution fits the exact solution well while there is an overestimation of temperature at the tail part. Both OGS and FD show an underestimation in the front part and an overestimation in the tail part. The overestimation and underestimation are mainly caused by numerical dispersion. The temperature distribution of the model with FD scheme presents the largest numerical dispersion.
In order to reduce the numerical dispersion and to improve the models performances, both finer cell and shorter time step are applied in OGS while only finer cell is applied in SEAWAT because it automatically adjusts the time step for heat transport process simulations.

Different cell size

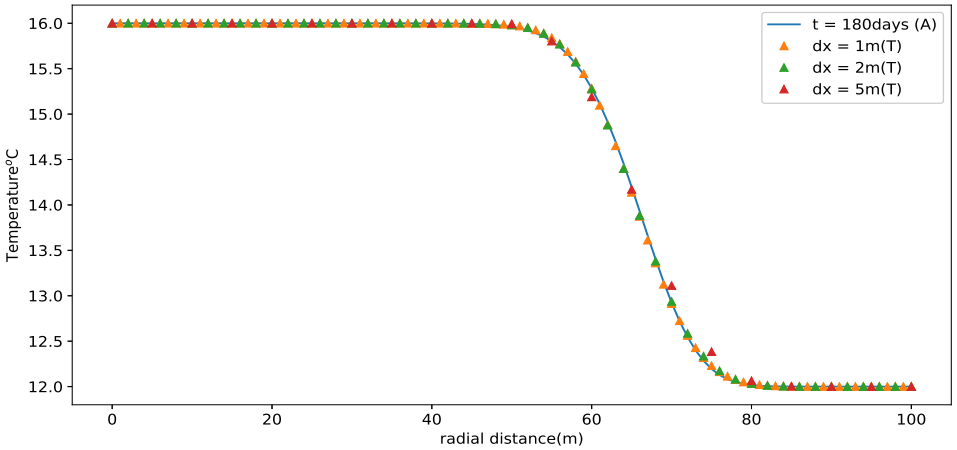
Figure 3.3 depicts how different spatial resolutions affect the model performance

with cell sizes of 5×5, 2×2 and 1×1, respectively while keeping the time step at 5 days. According to the horizontal temperature distribution, finer spatial resolution affect the model performance significantly when applying finite difference scheme as the advection solution. However, the numerical dispersion still exists even applying the finest cell size of 1×1. Moreover, the numerical solution of the model with TVD scheme is able to fit the exact solution very well after adopting a finer cell size (2×2). However, the temperature distribution for the OGS stays the same regardless of applying different spatial resolutions.

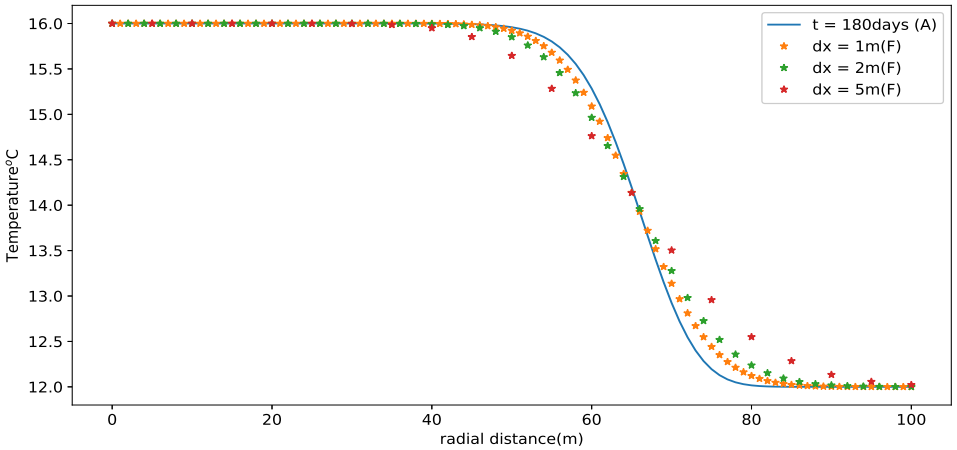
Since SEAWAT automatically adapts the time step for the heat transport process, it's meaningless to compare the OGS result with SEAWAT while the comparison between TVD and FD is feasible. Based on figure (b) and (c) in figure 3.3, it is clear that the model performance of temperature distribution for TVD is much better than FD.



(a) OGS environment(Horizontal spatial resolution: 5m×5m;2m×2m;1m×1m)



(b) TVD scheme (Horizontal spatial resolution: 5m×5m;2m×2m;1m×1m)



(c) FD scheme (Horizontal spatial resolution: 5m×5m;2m×2m;1m×1m)

Figure 3.3: Temperature distribution at t=180 days for the exact solution and numerical solutions for different spatial resolutions at a constant time step of 5 days for OGS; A (analytical), T (TVD), F (Finite Difference), O (OGS).

Different time step

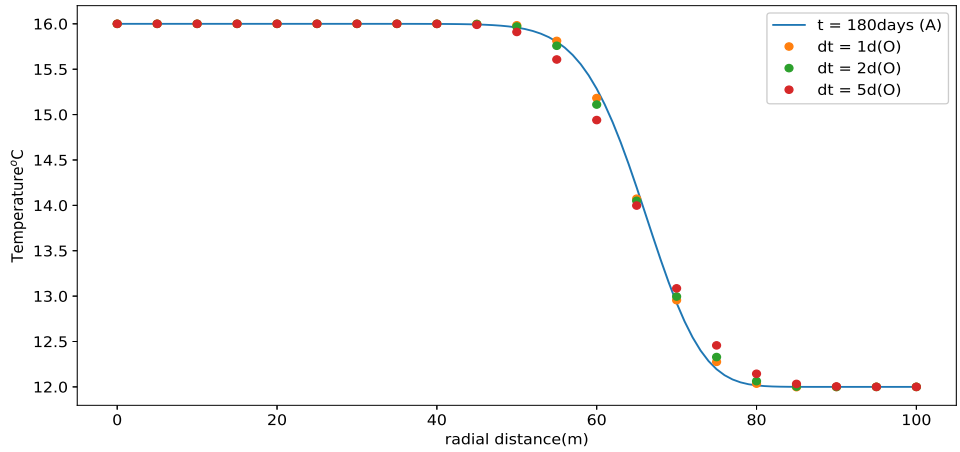


Figure 3.4: Temperature distribution at t=180 days for the exact solution and OGS for different time steps and a horizontal spatial resolution of 5m$\times$5m; A (analytical), O (OGS).

The temperature distribution along the radial direction for different time steps in the OGS environment is given in figure 3.4, depicting that the curve fitting between the numerical solution and the exact solution is improved by adopting larger temporal resolution (shorter time step).

Energy balance

Beside the temperature distribution, energy balance is also an important factor to assess a model's performance. Thus, the energy balance for all models shown above has been computed and presented in table 3.1.

The energy balance error for the analytical solution is computed by Guassian quadrature and gives a result of 5.83×10^{-10}. Since the mesh is in a regular shape, the energy stored in the system is computed by both Paraview and the sum-up method for OGS, showing the same result which confirms that Paraview works for this model. The energy balance error for all models is smaller than 1%. The adjusting of temporal resolution and spatial resolution has a small influence on the energy balance. There is always an overestimation of energy in both TVD and

Table 3.1: Energy balance error (%) for different models in Sand model.

Time step		5 days	
Cell size	5m×5m	2m×2m	1m×1m
OGS	-0.0105	0.0110	0.0128
TVD	0.00131	0.00136	0.00136
FD	0.00112	0.00127	0.00132
Cell size		5m×5m	
Time step	5 days	2 days	1 day
OGS	-0.0105	-0.0111	-0.0117

FD schemes while for OGS, there is mostly an underestimation of energy. In general, SEAWAT shows a better energy conservation than OGS based on the smaller energy balance error but the difference is small.

3.1.2 Clay-sand-clay model

After running the model with isotropic dispersivity in the aquifer, some oscillations occurred at the well in the aquitard after one-time-step simulation in SEAWAT environment despite that aquifer performed normally during the modelling process as shown in 3.5. The temperature at the cell next to the center cell is much higher than that of the center cell while this does not happen in OGS. To get rid of this oscillation, a spatial resolution adjustment is applied but failed. The oscillation always occurred at the cell next to the center one no matter how fine the spatial resolution is, indicating that this oscillation has nothing to do with the spatial resolution.

Considering all parameters set used in this model, the only parameter that might cause the oscillation is the vertical dispersivity since the dispersivity is regarded as isotropic in the aquifer which is not the case in reality and OGS does not take the vertical dispersivity into account. As suggested in Mt3dms, the vertical dispersivity is usually 100 times smaller than the longitudinal one. Thus, an anisotropic dispersivity is applied and the oscillation is eliminated , see figure 3.6.

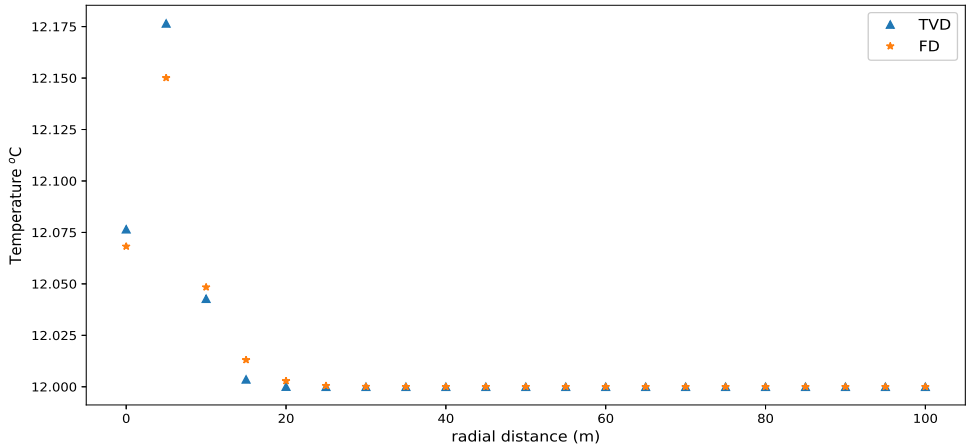


Figure 3.5: Temperature distribution after one-time-step simulation at the first clay layer above the aquifer in SEAWAT environment (isotropic dispersivity).

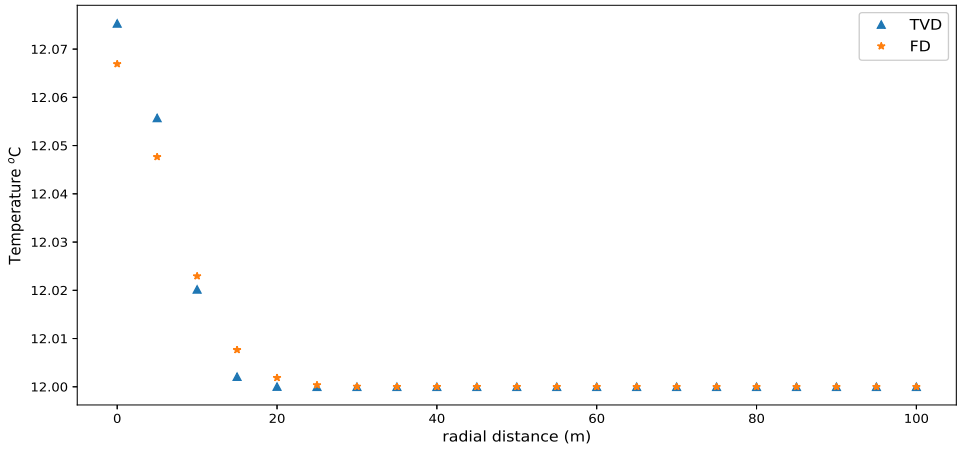


Figure 3.6: Temperature distribution after one-time-step simulation at the first clay layer above the aquifer in SEAWAT environment (anisotropic dispersivity).

Aquifer

Figure 3.7 depicts that the heat transport distance of numerical solutions is larger than the exact solution in the aquifer part, especially the FD scheme which can be explained by the largest numerical dispersion caused by the coarse spatial resolution, presented in the Sand model. Additionally, the temperature distribution for three different numerical solutions presents a rightward migration based on the analytical solution. As introduced in chapter 2, the analytical solution for the clay-sand-clay model is in the Laplace form which is solved numerically as well, leading to some errors during iterations.

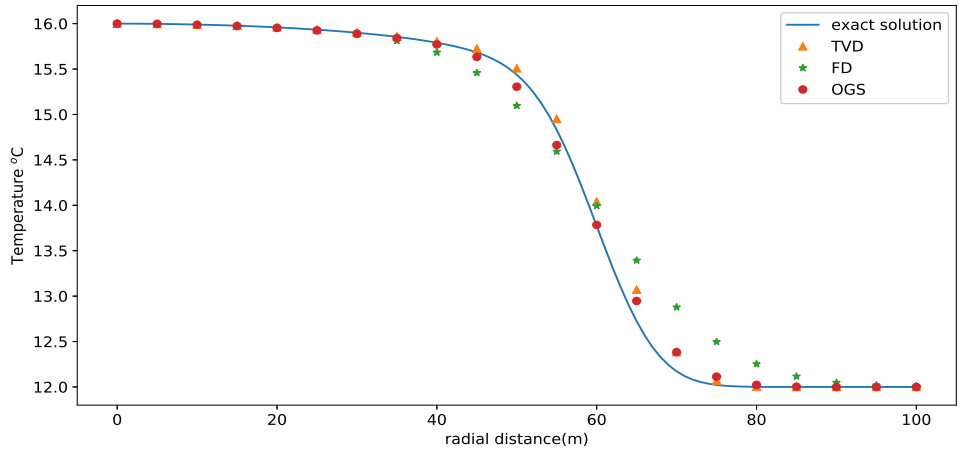
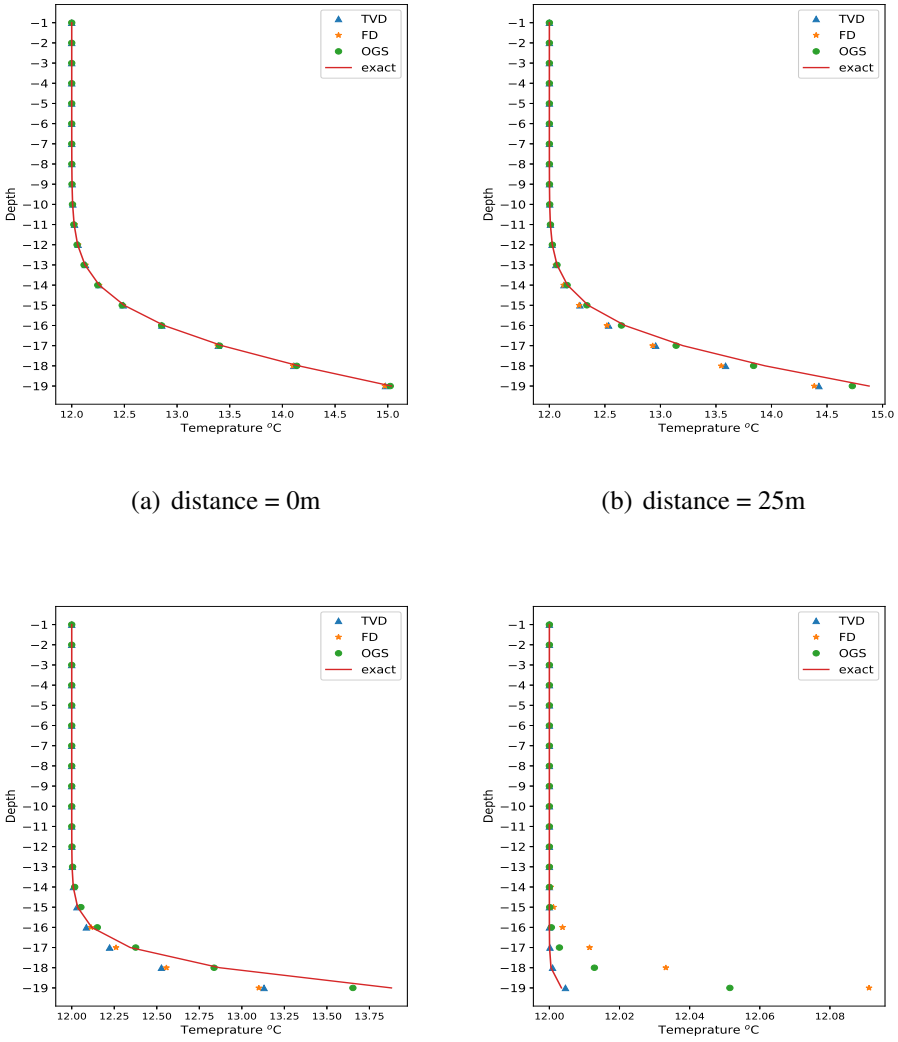


Figure 3.7: Temperature distribution along the radial direction in the aquifer of the analytical solution and numerical solutions at t = 180 days.

Aquitard

In order to check how numerical models perform in the aquitards, the temperature distribution along vertical direction is plotted and several distances to the wellbore are selected as shown in figure 3.8. In subfigure (a), all three numerical models present a good fit with the exact solution. As the distance to the wellbore increases, SEAWAT shows an significant underestimation of the temperature transport within the clay layer while OGS presents a smaller underestimation. In the last subfigure, there is an overestimation of the energy transported into the aquifer for all models, especially FD scheme and OGS. This is because the longer

heat transport distance resulted from the numerical dispersion leads to a larger influence range in the clay layer.



(a) distance = 0m

(b) distance = 25m

(c) distance = 50m

(d) distance = 75m

Figure 3.8: Vertical temperature distribution in the aquitard with different distances to the wellbore after 180 days.

Energy Balance

The energy stored in the aquifer part and the aquitard part is computed for OGS, TVD scheme and FD scheme in the similar way with the sand model, respectively. While for the exact solution, the quad method is applied since the temperature distribution is continuum. According to table 3.2, there is more energy stored in the aquitard and less energy stored in the aquifer for the exact solution. Compared to SEAWAT (both TVD and FD schemes), OGS shows a similar pattern with the exact solution. When looking at the energy balance error, the slight underestimation exists for both OGS and SEAWAT while the exact solution presents an overestimation. Overall the energy balance error is small.

Table 3.2: Energy balance error (%) for different models in Clay-sand-clay model.

	energy (J)			energy balance (%)
	Aquifer	Aquitard	Total	
OGS	1.682×10^{12}	0.158×10^{12}	1.8402×10^{12}	-0.017
TVD	1.706×10^{12}	0.135×10^{12}	1.8405×10^{12}	-0.0011
FD	1.705×10^{12}	0.135×10^{12}	1.8405×10^{12}	-0.0011
Exact	1.648×10^{12}	0.193×10^{12}	1.841×10^{12}	0.022

In conclusion, a finer cell size is suggested for SEAWAT and a shorter time step is recommended for OpenGeoSys in order to eliminate the numerical dispersion in this study. Specifically, a cell size of 2m$\times$2m in horizontal direction is well enough for TVD while a much smaller cell size is recommended for FD. As for OGS, a time step of 1 day is recommended. However, in later models, a cell size of 5m$\times$5m and a time step of 5 days are applied initially in order to save computational time. In the vertical direction, a thickness of 1m for each layer is suggested and applied in later models.

3.2 Compare the performance of OGS and SEAWAT in the single-cycle ATES system

In this section, a simple ATES system is built and simulated in both OGS and SEAWAT environments. The parameters applied in this model is same with the Clay-sand-clay model and the storage volume is 250,000 m^3/year. In order to prevent the energy loss due to the interaction between two wells with opposite temperature deviation according to the ambient temperature (12 ^{o}C), a distance around two times the thermal radius is adopted in this model. According to the definition of the thermal radius:

$$R_{th} = \sqrt{\frac{V_s C_w}{C_{aq} \pi L}} \tag{3.1}$$

where V_s is the seasonal storage volume [m^3], C_w is the heat capacity of water [J/m^3/K], C_{aq} is the volumetric heat capacity of the aquifer [J/m^3/K] and L is the well screen length [m], the distance is determined to be 140 meters which is two times of the thermal radius as shown in figure 3.9. The black dashed line is the observation line for the temperature distribution along x-axis. The model mesh domain is 400m$\times$400m with a cell size of 5m$\times$5m and the time step is 5 days. The warm well is located at a distance of 70 meters towards left the center of the domain and the cold well is 70 meters towards right the domain center. In this section, the ATES system is operated for only one cycle to check how different numerical models perform. In this model, the setting of OGS is different from that of former models due to the changing of the flow direction. The changing of the flow direction means that the flow is not constant during the operation time. Moreover, the temperature boundary condition is not feasible neither during the extraction period. To deal with this scenario, OGS provides the RFD object. The changing of flow can be defined in this object by specifying a certain flow rate with a corresponding timing. In order to apply this timing curve, the TIM_TYPE within the ST object requires to be called by specifying the serial number of the curve as there are commonly more than one curve in a model. To control the boundary condition for the temperature, the RFD

object and a function called TIME_CONTROLLED_ACTIVE in the BC object are combined. Given a two-dimension list file of time and a value (0 or 1), the TIME_CONTROLLED_ACTIVE would activate the boundary condition during the time when the value is 1 and deactivate the boundary condition when it is 0. Thus, in the RFD file, a list file is defined including timing and corresponding values. The value is set to be 1 during the injection period and 0 for the extraction period. Similarly, the curve can be called by specifying the curve serial number in the TIME_CONTROLLED_ACTIVE. The examples are given in the Appendix.

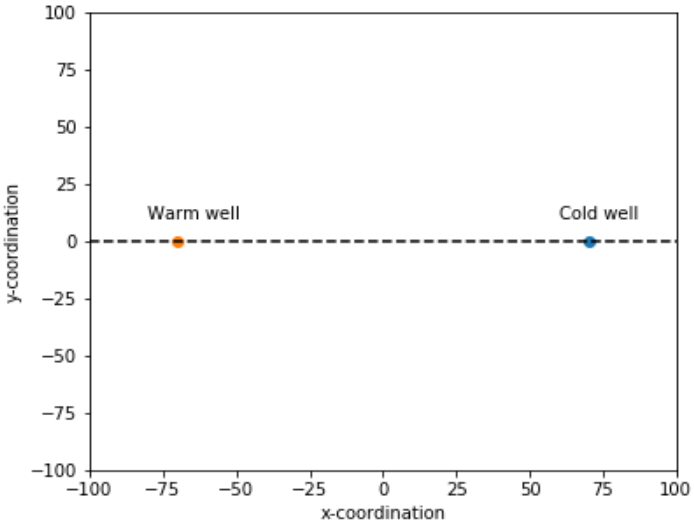


Figure 3.9: Wells location in the doublet model.

Aquifer

In order to compare the heat transport result for OGS and SEAWAT, the temperatures of different layers for the aquifer are averaged. The averaged temperature distribution within the aquifer along the observation line is presented in figure 3.10. Different numerical models present a similar pattern with different numerical dispersion extents, especially OGS and TVD scheme. The temperature at the warm well for OGS is slightly higher than that for both TVD scheme and FD scheme. In addition, the computational time for OGS is much longer than that of

SEAWAT (over ten times) in this model.

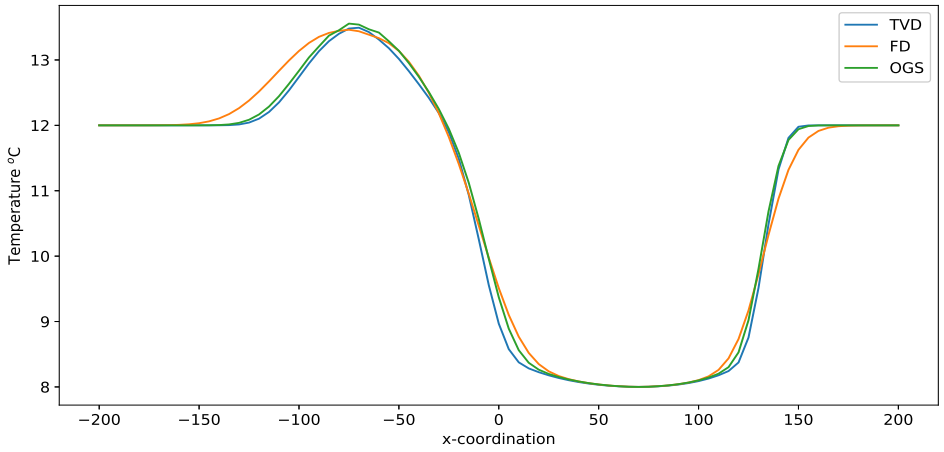


Figure 3.10: Temperature distribution along observation line in the aquifer after 360 days.

Aquitard

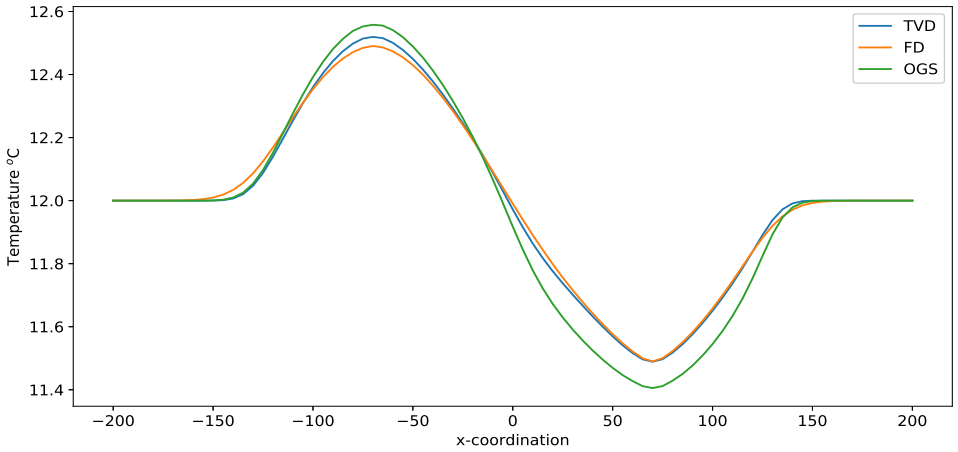


Figure 3.11: Temperature distribution along observation line in the aquitard after 360 days.

It is obvious that the difference is larger in aquitard, see figure 3.11 which can be

explained by different heat transport solution method for OGS and SEAWAT in a vertical direction. There is a higher temperature at the warm well and a lower temperature at the cold well which indicates that more energy is transported into the aquitard in the OGS numerical model.

Warm Well

In an ATES system, the extraction temperature is of great importance as it is highly associated with the extracted energy. Thus, the temperature of the warm well against the operation time is plotted and depicted in figure 3.12. The cold well is neglected here since there is only one operation cycle.

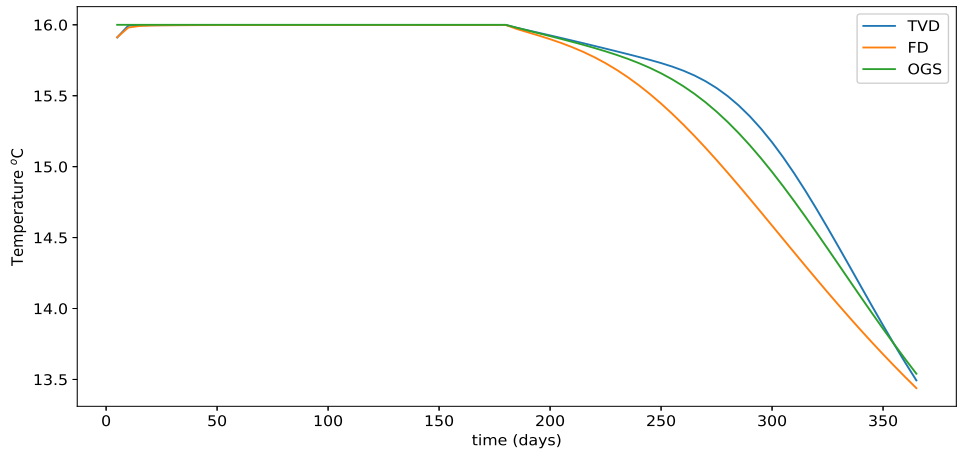
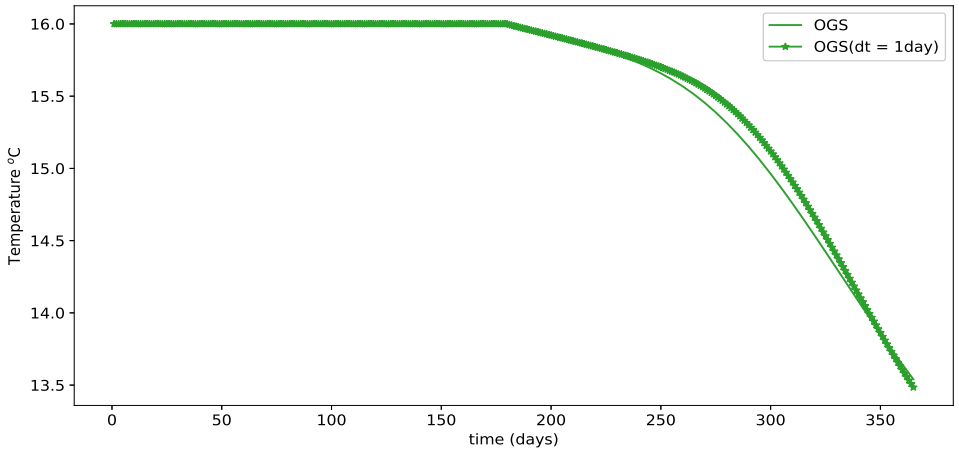


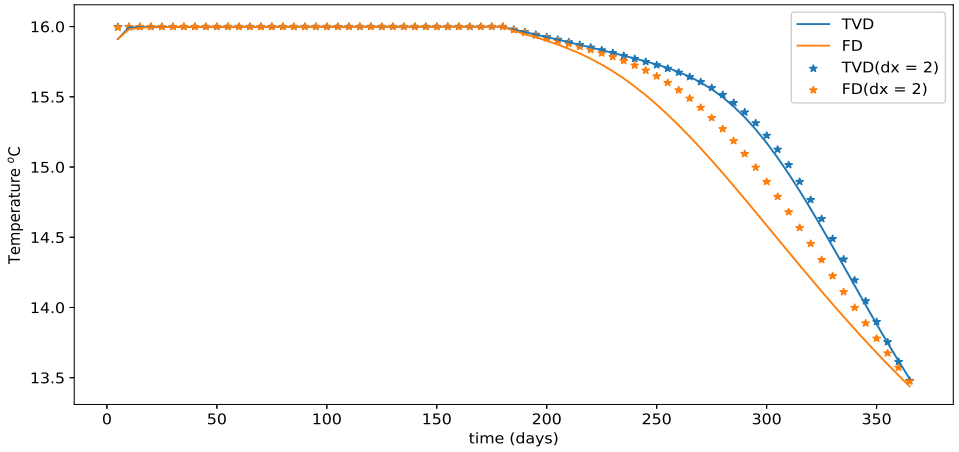
Figure 3.12: Temperature of the warm well at different operation time.

As shown in the figure above, the little drop at the beginning of SEAWAT is caused by the combination of lack of enough injection water and the diffusion process. However, this is not the case for OGS because the temperature of the warm well is set to be the boundary value to represent the injection temperature. Apart from this, the well temperature of TVD scheme is higher than that of OGS and FD scheme. As mentioned in section 3.1.2, there is more energy stored in the aquifer for SEAWAT which can be one of the reason that the extraction temperature is lower for OGS. In addition, the numerical dispersion of OGS and FD scheme is larger, resulting in that longer time is required to extract the same amount of en-

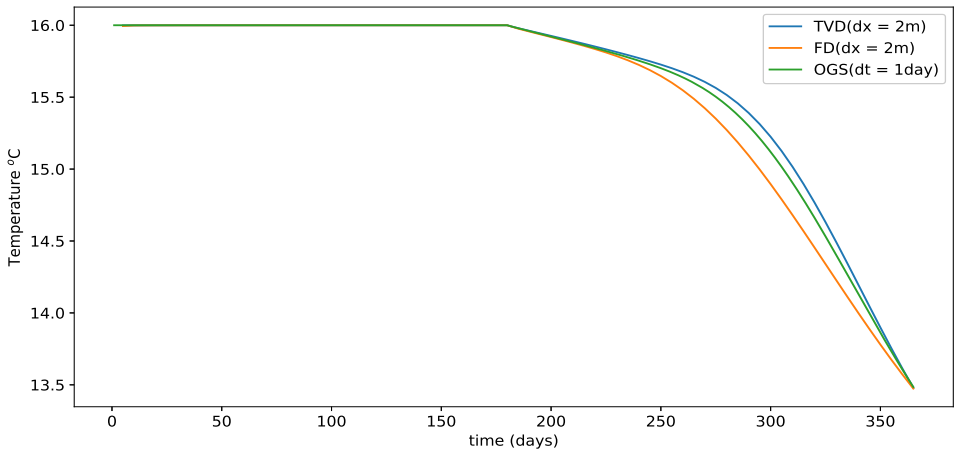
ergy as TVD. The larger the numerical dispersion, the longer the extraction time needed and the lower the temperature during the extraction period. To see how the model performance at the warm well temperature is affected by eliminating the numerical dispersion, an adjusting of cell size and time step is applied and the result is presented in figure 3.13.



(a) Temporal resolution adjustment for OGS



(b) Spatial resolution adjustment for SEAWAT



(c) The result after adjustment

Figure 3.13: Temperature of the warm well at different operation time when applying the temporal and spatial resolution adjustment in both SEAWAT and OGS.

As shown in configuration a, the temperature at the warm well increases during the extraction period while decreases at the tail part when applying a shorter time step. As for SEAWAT, there is also an increasing of the well temperature for both FD and TVD scheme but in different extent when applying a finer spatial resolution. Overall the difference between three numerical models has been diminished and the pattern for OGS and FD behaves closer to that of TVD which indicates that TVD scheme has a better performance because of small numerical dispersion. And this also proves that numerical dispersion is the most significant factor that affect the performance for numerical models and can be diminished by applying smaller time step and cell size at the cost of computation time. Based on this, it can be speculated that the well extraction temperature for different numerical models would be same if a small enough cell size for SEAWAT and a short enough time step for OGS can be utilized.

Energy Balance

The extracted energy and the energy remained in the system are calculated where the extracted energy is computed by Simpson's rule, see tabel 3.3.

Table 3.3: Energy balance error (%) for different models in doublet model .

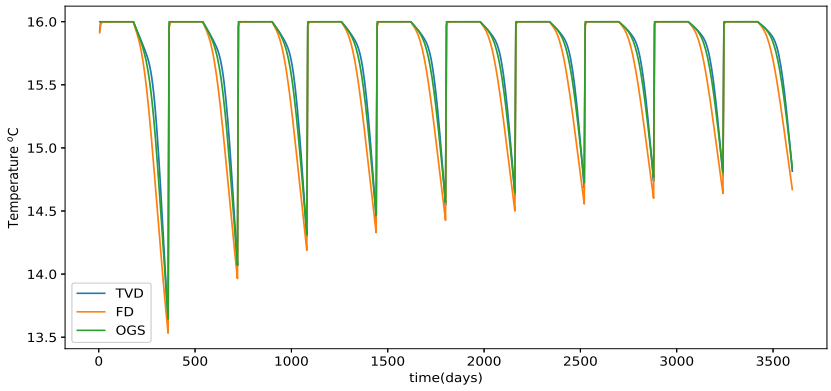
	energy (J)			energy balance (%)
	Extracted	Reamined	Total	
OGS	1.673×10^{12}	2.503×10^{12}	4.176×10^{12}	-0.163
TVD	1.720×10^{12}	2.460×10^{12}	4.179×10^{12}	-0.103
FD	1.563×10^{12}	2.548×10^{12}	4.112×10^{12}	-1.705

The energy conservation performance in this model is worse than that of one-well-injection model but still acceptable. As can be seen from the table, the energy balance error for FD scheme is much larger than that of the other numerical models which can be caused by the interaction between two wells with opposite temperature deviations based on the ambient temperature. Theoretically, this interaction would not exist because the distance between two wells is longer than two times the thermal radius. However, due to the numerical dispersion, the heat transport distance becomes longer, especially FD scheme. Additionally, more energy is extracted in the TVD scheme numerical model which indicates a higher recovery efficiency.

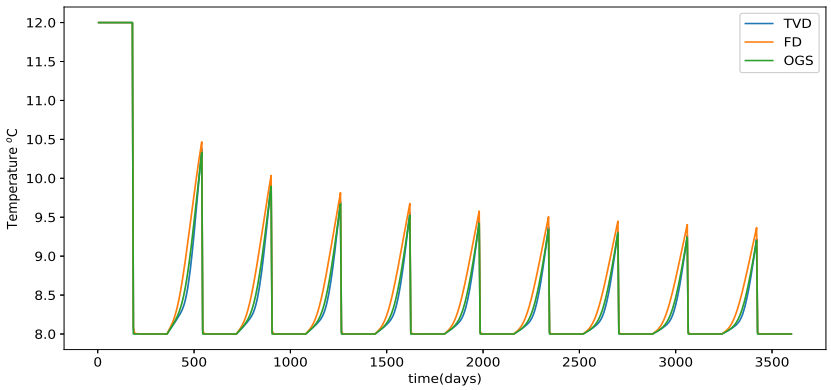
3.3 Compare the performance of OGS with SEAWAT when applied in the multi-cycle ATES system

In order to save computational time and get rid of the wells interaction, a larger domain with 500m×500m and a distance of 200 meters between two wells are applied while the cell size and the time step stay the same. In this section, more operation cycles are taken into account so that it can be more realistic. In this section, 10 cycles (years) are applied. Within each cycle, the water is injected into the warm well and at the same time extracted from the cold well for the first 180 days and then the flow direction is reversed for another 180 days.

Figure 3.14 depicts how different numerical models perform in the wells temperature at different operation time. At the warm well, the lowest temperature increases by years due to that not all the energy would be extracted during the extraction period, resulting in an energy accumulating over time in the soil. Similarly, the highest temperature at the cold well decreases by years. Generally, three numerical methods present a similar pattern and the difference can be diminished by adjusting the temporal and spatial resolution as mentioned in section 3.2.



(a) Warm Well



(b) Cold Well

Figure 3.14: Temperature of wells at different operation time in multi-cycle model in SEAWAT and OGS environment.

Figure 3.15 depicts the variation of the recovery efficiency of the ATES system during operation time. It is seen from plots that TVD scheme presents a higher recovery efficiency than that of the other two methods during the whole operation time. Additionally, all three methods present a phenomenon that the increasing rate of the recovery efficiency becomes smaller by longer operational time. And it is expected that the recovery efficiency will become constant after enough operational time. This is because the warm and cool zone keep growing until reaching the thermal zone that the distance at the zone boundary to the wellbore is equal to the thermal radius.

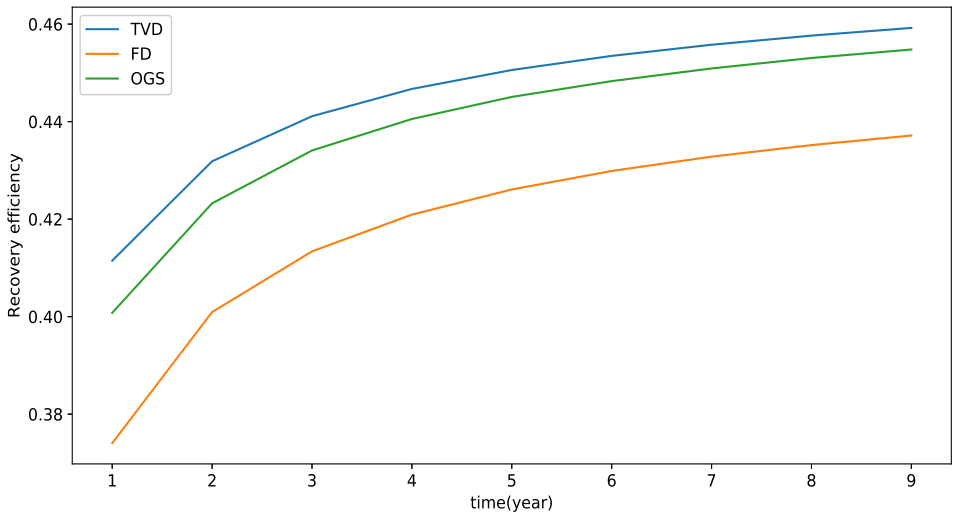


Figure 3.15: Recovery efficiency variation against the operation cycles.

Energy balance

The energy balance error has been computed for each cycle by equation 2.46, see figure 3.16. Initially, SEAWAT still presents a better energy conservation than OpenGeoSys. However, the energy balance error increases significantly for FD scheme by time. This is caused by the doublet interaction and the larger numerical dispersion reinforced this process in FD scheme. As for OpenGeoSys, the energy balance error value presents a decreasing tendency which is caused by the energy loss due to the interaction as well. Overall, TVD scheme shows a better energy conservation in this model due to the limited numerical dispersion.

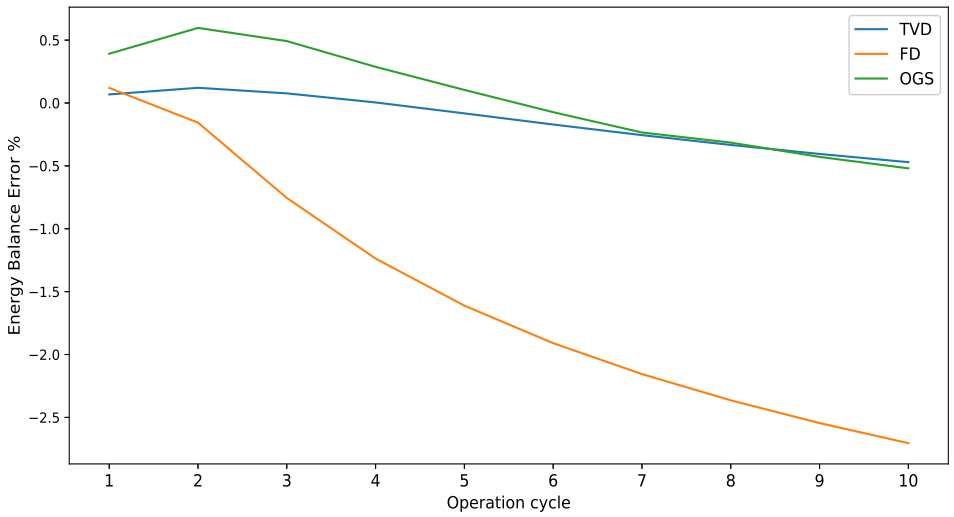


Figure 3.16: Energy balance error variation against the operation cycles.

3.4 Compare the performance of OGS with SEAWAT in field case

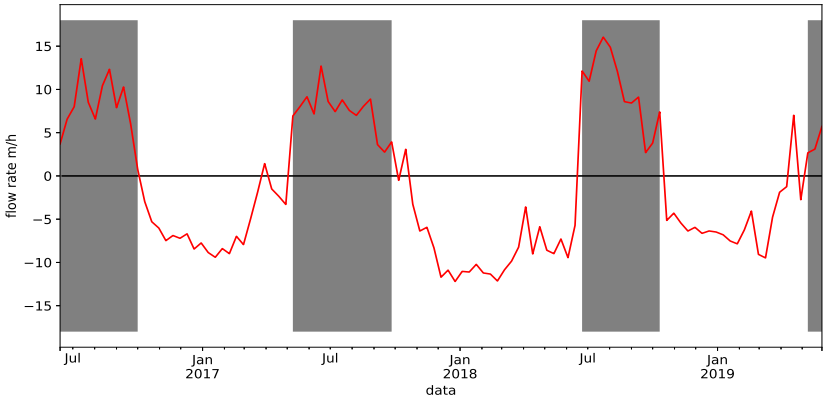
In this section, a real field data applied in three numerical models are compared. 3M building in TU Delft campus is selected as the field data to be studied and the data includes dates, pressures at the doublet and the flow rate as well as the temperatures for the doublet.

3.4.1 Data Aggregation

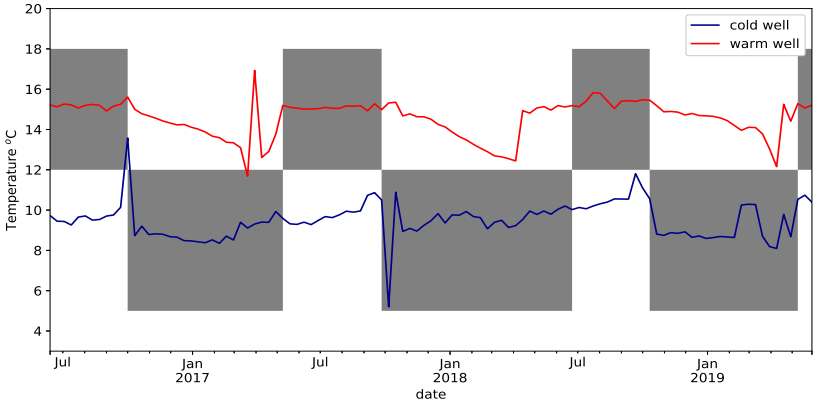
The interpolation is applied to fill in the vacant data by averaging the data at the same time in other years. In addition, the temporal resolution of the data is 8 minutes which is too fine to be applied in the numerical models. Thus, the weighted temperature to the flow rates is utilized with an aggregation period of 10 days.

The interpolation and aggregation results are depicted in figure 3.17. The shadow part indicates an injection period for the warm well. In addition, there are some

dates that the water is injected into the warm well outside the shadow part which is due to that the ATES system in reality would adjust the flow direction according to the energy requirements rather than applying a steady cycle. Generally, the warm water is injected into the warm well during summer time and extracted during winter time. Since the temperature is weighted to the flow rates during aggregation, there can be some spikes once the aggregated flow rate is relatively low as can be seen from sub-figure b.



(a) flow rate

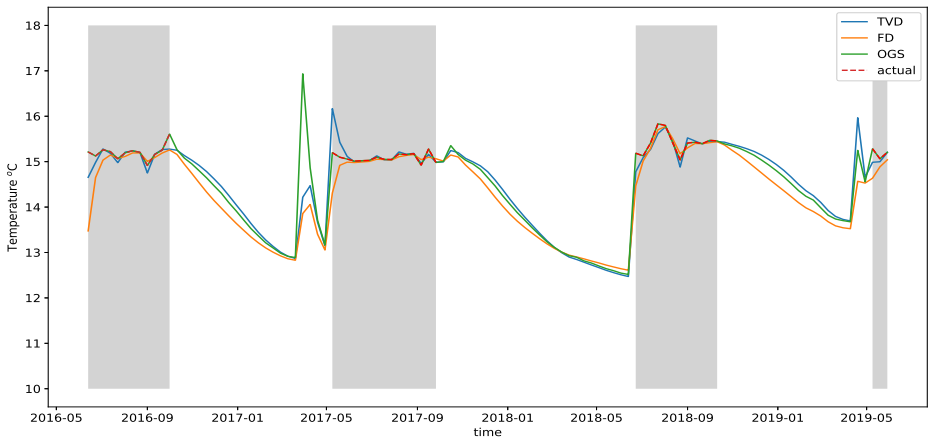


(b) temperature

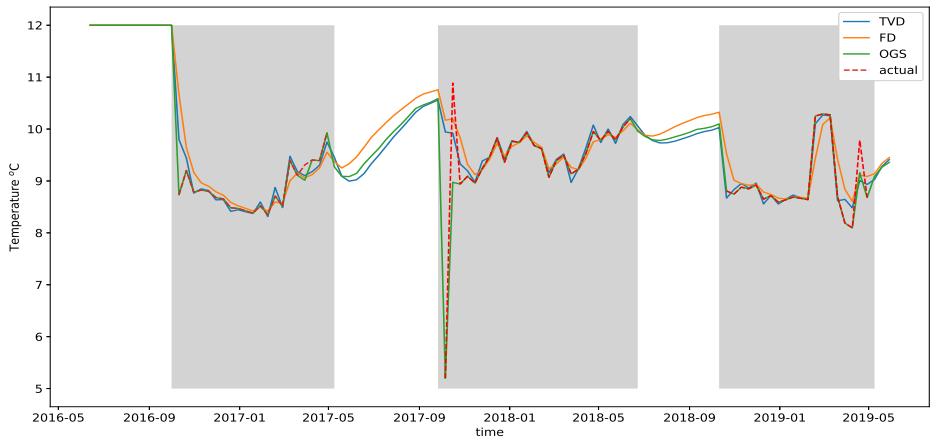
Figure 3.17: a)flow rate at the warm well (positive represents the injection while negative represents the extraction); b)measuring temperature at warm/cold well

3.4.2 Results of the temperature at the doublet

The field study is proceeding by assigning the flow rate and the corresponding temperature in the numerical models setting. The time step is set to be 10 days initially according to the data temporal resolution. Figure 3.18 presents that three numerical models behave similarly. More specifically, during the shadow part which is the injection period, the temperature at the wells is consistent with the measuring field data OpenGeoSys due to the boundary condition setting. As for SEAWAT, sometimes the temperature at the warm well is lower than the measuring data and the cold well temperature is higher compared to the field data because of the limited amount of injected water during those periods and the dispersion effect. In addition, FD scheme shows a larger temperature difference within injection period as it has the largest numerical dispersion which intensifies the effect of the dispersion most. The spikes occurring in April of 2017 and 2019 at the warm well are due to that there are water injected into the warm well during that period. Compared with models with the constant inflow rate and constant injection temperature, the difference between three models is more obvious.



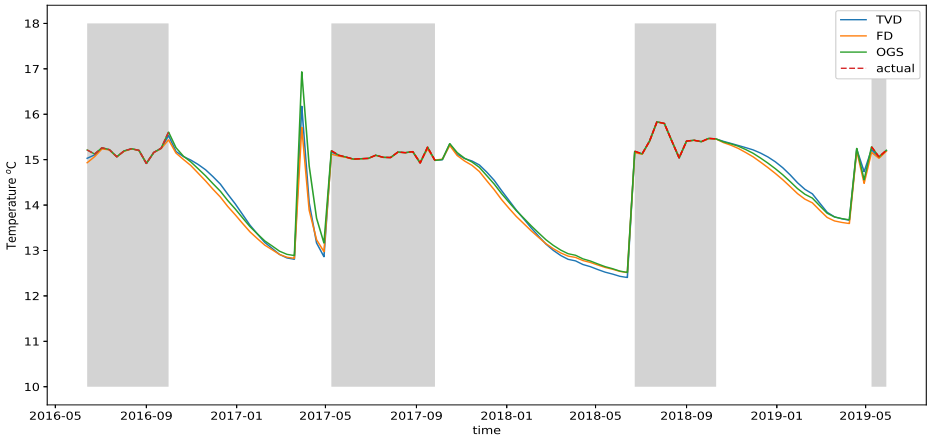
(a) Warm well



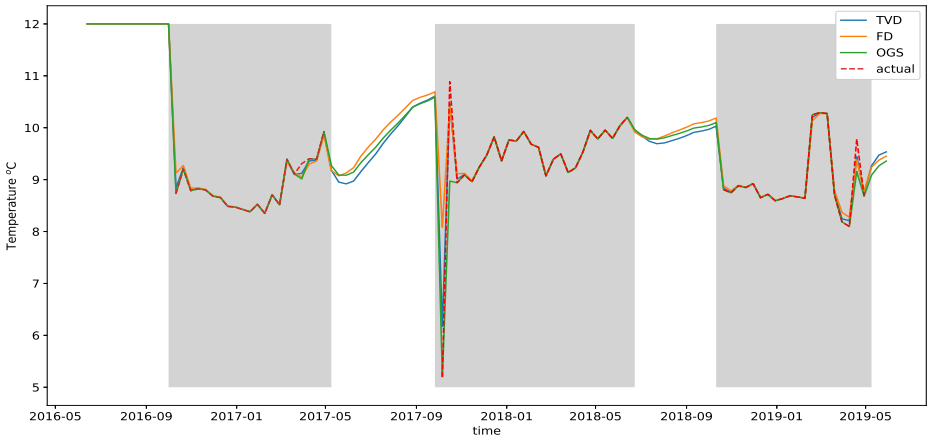
(b) Cold well

Figure 3.18: The numerical solutions of the doublet temperature variation in operation at 3M building.

In subfigure a, there are some oscillations occurring in TVD scheme in 2017-05 and 2019-04 that the temperature at the warm well is higher than the injected water temperature which is not possible. Similarly, this happens at the cold well as well. In order to get rid of the oscillation and ensure that the temperature at the well is equal to the injection temperature, a smaller cell size (2m$\times$2m) is applied in SEAWAT, see figure 3.19. The oscillations for TVD scheme is eliminated and the temperature at the doublet during shadow period is same with the measuring temperature for all numerical models despite some dates that the flow rate is extremely small. In addition, the difference between three models during the extraction period has been narrowed.



(a) Warm well



(b) Cold well

Figure 3.19: The numerical solutions of the doublet temperature variation in operation at 3M building(finer cell size for SEAWAT).

4 | Discussion

This chapter focuses on the different aspects which are considered in this study and need to be considered before applying the results in practice such as the model set-up and assumptions.

Both OpenGeoSys and SEAWAT perform well in energy conservation without considering the interaction between the doublet; the error is much smaller than 1%. Among two environments, SEAWAT performs better in energy conservation when applying the same spatial and temporal resolutions. For the sake of the same spatial resolution, FD scheme shows a larger numerical dispersion compared to TVD scheme and OpenGeoSys.

Model set-up - meshgrid
The OGS model meshgrid is set to be in a regular shape in this study so that the sum-up method can be applied easily. In order to compare different numerical models, similar meshgrid is applied in SEAWAT as well. However, the mesh domain could be much larger and the operation time would be much longer in a more complicated ATES well configuration system. In this case, it would be time-consuming if applying this regular meshgrid and an irregular grid could be much more efficient. In the irregular-meshgrid, the cells around wells needs to be refined while the cells away from wells can be coarse to reduce the number of cells so that less computational time is needed. In addition, the Paraview is a suitable alternative to compute the energy in a irregular-meshgrid model where the sum-up method is not feasible for OGS. It is expected that this irregular-meshgrid model would not significantly influence the result if the refined domain is large enough. Future studies can take the comparison of regular meshgrid and irregular meshgrid into account.

Model set-up - time step
In MT3DMS of SEAWAT, the heat transport time step is automatically adjusted according to the cell size to meet the Courant condition. There is no method to get to know the adjusted time step applied in SEAWAT. For OGS, the simulation

time step for heat transport applies the value specified in the TIM object for the corresponding process. In this case, only the time step for groundwater flow is the same for both numerical models. Thus, the numerical dispersion results from the coarse spatial resolution for SEAWAT while both coarse temporal and spatial resolutions could lead to numerical dispersion for OGS. In order to reduce the numerical dispersion, the spatial resolution is adjusted for SEAWAT and both spatial and temporal resolutions are adjusted for OGS. However, the spatial resolution does not affect the model performance if the temporal resolution is not adjusted as well in OGS environment which is explained by that smaller cell size requires shorter time step to meet the Courant condition. Thus, only adjusting spatial resolution might not be able to improve the model performance.

Model set-up - temperature boundary

The temperature of the injected water is set to be a boundary condition in OGS while it is regarded as a source term in SEAWAT during the injection periods. In a case where there is a relative small flow rate while the cell size is large, the well temperature might not be able to reach the temperature for the injection water in SEAWAT. As for OGS, the well is represented as a polyline so that there is no limitation for the injection flow rate. The temperature of the well is exactly the injection temperature for OGS. This leads to a different pattern when looking at the wells temperature during injection period, especially when there is a sharp decline or increase of the temperature for the injection water. This difference pattern of temperature during injection period has no influence on the temperature at the well during extraction period except the scenario that the peak temperature occurs right before the extraction period. Different temperatures of the well at the beginning of the extraction period might lead to different extracted energy computed by Simpson's rule at the very early time for SEAWAT and OGS. In principle, this little amount of flow with an extreme temperature has no significant influence on the surrounding temperature and the temperature at the well rapidly falls or increases back to the surrounding temperature once starting extracting water. Compared with the total extracted energy during the extraction period, this influence is limited and can be neglected.

Approximations

In this research, several approximations have been made which may not be realistic in practice. The temperature variation is not large in this study, so that the variation of density and viscosity is neglected for both OGS and SEAWAT. In practice, the temperature variation can be larger which could significantly affect the density and viscosity. It is feasible to simulate the varying density and viscosity models in both SEAWAT and OGS environments. For future studies, the influence of the varying density and viscosity can be included.

In order to simulate the analytical solution, the hydraulic conductivity of the aquifer is approximated as homogeneous. However, the horizontal hydraulic conductivity and the vertical hydraulic conductivity are usually different in practice. Wells penetrate the whole aquifer in all models in this study so that the water flowing in the aquifer vertically does not influence the model performance. The thermal dispersivity of the aquitards is approximated as 0 as well so that the dispersion process is neglected in the aquitards. This works for models with low permeability clay layers where there is almost no flow in aquitards, so that the dispersion process can be neglected. In practice, the hydraulic conductivity for the aquifer is heterogeneous. But this has no effect on the head distribution since wells penetrate the whole aquifer. Although the dispersivity for the aquitards is not 0 in practice, the dispersion resulted from the small flow velocity in the aquitards can be neglected.

The ambient groundwater flow is not considered in this study while it can affect the model performance in practice. The presence of the ambient groundwater flow influences the thermal and hydraulic radius. As a result, the heating/cooling energy transporting farther along the ambient groundwater flow direction which leads to a decline of the thermal efficiency. In addition, in the case that the ambient groundwater flows between the doublet, the energy balance error increases due to the increasing interaction of the two wells. This interaction becomes more significant when the numerical dispersion is larger. For this condition, the temporal resolution should be refined for OGS to prevent the numerical dispersion.

The head distribution near the well is different between OpenGeoSys and SEAWAT. In this study, only the heating/cooling energy is taken into account to assess the performance. However, the pumping energy is of great importance as well in an ATES system. The difference in heads at the wells leads to a different pumping energy requirement, leading to different ATES system performance. Larger drawdown of wells in OGS has a negative effect on ATES system performance as more energy is needed for well pumps to extract and infiltrate the groundwater. However, the difference is small and can be neglected compared to the large amount of thermal energy.

The comparison between the measured data in the field case and the numerical simulation results is not included in this study due to the limited subsurface and historical data to get a reliable calibration. In order to assess how numerical modelling performs compared to the measured data, more details are needed such as the initial ambient groundwater temperature, and the occurrence of other ATES systems around the building.

Comparison of FE and FD methods

Due to the different meshgrid setting, the number of degree of freedom (DOF) is different for FEM and FDM. Within FEM, each element is formed by several nodes and the unknown variables are computed and saved at each node. While in FDM, the unknown variables are only computed and saved at the center of each cell. For instance, in a domain of $10\times10\times10$ with the spatial resolution of $5\times5\times5$, the number of nodes within FEM is $3\times3\times3$ (216 DOF) while the number of nodes within FDM is $2\times2\times2$ (64 DOF). As a result, the number of DOF in the FEM is larger when applying the same domain and spatial resolution. But this is negligible when considering the whole model.

5 | Conclusions

5.1 Conclusions

The objective of this research is to assess the usability of OpenGeoSys tools in the simulation of ATES systems. This study solves the problem of interests of how OGS performs in ATES systems simulations compared with analytical solutions and other numerical methods. The performance is assessed by comparing the temperature distribution and the energy balance error. This study proved that the ATES systems can be built and simulated in OGS environment. Both OGS and SEAWAT perform well in energy conservation that the errors are much smaller than 1% .

This research demonstrated that both SEAWAT and OpenGeoSys can reproduce the heat transport process of analytical solutions for one-well models. There are some deviations for the numerical solutions from the analytical solutions when looking at the horizontal temperature distribution which are primarily caused by the numerical dispersion. The TVD scheme presents the least numerical dispersion while the FD scheme presents the largest one when using approximately the same number of cells/nodes. When taking the heat transport between the aquifer and aquitards into account, OGS shows the least overestimation of the thermal energy remained in the aquifer compared with the analytical solution.

Both the numerical dispersion and the overestimation of the thermal energy have the influence on the ATES system thermal recovery efficiency. Larger numerical dispersion indicates a longer heat transport distance, resulting in more time needed for the thermal energy to transport back to the wells during the extraction period. Besides, the interaction between two wells could be enhanced due to the longer heat transport distance and, thus, results in a larger energy balance error. Similarly, the overestimation of the thermal energy allows more heat to transport in the aquifer and prolongs the heat transport distance as well but simultaneously provides more thermal energy during the extraction period. Longer heat transport

distance has a negative effect on the thermal recovery efficiency for the ATES systems while the overestimation has both negative and positive effects.
The OGS is more reliable in the aspect of distribution of the thermal energy in different soil types compared with the SEAWAT and thus a more suitable software for ATES systems simulations. The numerical dispersion can be reduced by refining the spatial and temporal resolution while the overestimation cannot. On the one hand, the simulation results based on SEAWAT could lead to an overestimation of the ATES system performance if the numerical dispersion is negligible. On the other hand, the overestimation of the thermal energy in the aquifer increases the possibility of the interaction of the doublet, leading to an energy loss and a decrease of the performance. Overall, the least overestimation and the moderate numerical dispersion of the numerical solutions based on OGS makes it a better alternative for ATES systems simulations.

However, there are some drawbacks of simulating ATES systems in the OGS environment. It is more complicated to implement an ATES system in the OGS environment than that in the SEAWAT environment. Specific platforms are required to define the meshgrid. For varying flow rate and injection temperature, OGS requires files that define time-variables curves as shown in Appendix A. The running time for a model which uses the same number of cells/nodes is longer for OGS than that for SEAWAT by a factor of around 10.

5.2 Recommendations

The recommendations for future study and ATES planning are given based on this research.

Future study
In this study, only analytical solutions for the one-well-injection model are investigated as no other solutions are available as far as the author knows. Future research can work on comparison of numerical solutions with analytical solutions in the doublet scenario since the extraction temperature is the most significant

factor in an ATES system. In addition, the exact solution for the clay-sand-clay model is solved numerically that results in a relatively large energy balance error. Future research can investigate which solver minimizes the error so that the analytical solution can be met more closely.
In practice, the ATES systems are more complicated as the installation of wells is not as simple as the models applied in this study and the soil profile can be more complex as well as the soil properties. For instance, the wells can penetrate multiple confined aquifers and the hydraulic conductivity can be heterogeneous everywhere as well as thermal properties in practice. In addition, multiple ATES systems can clustered together. Thus, future research can also take more complex/realistic models into account.
This study only uses the regular cell size in the meshgrid. In practice, the simulation area is much more large so that using the regular grid with a fine size is not suitable as it can be much more time-consuming. The irregular grid is a good alternative for this condition. The fine cell size is only required for some specific small areas that are significantly influenced by heat transport which saves the computational time efficiently.

ATES planning

For ATES planners, OpenGeoSys is recommended to be utilized in the ATES systems simulations although it is time-consuming and complicated to operate compared with SEAWAT. The FD scheme presents the largest numerical dispersion which requires the longest distance between two wells in a system. Besides, the numerical dispersion causes a lower extraction temperature as more energy is stored in the ground. In order to reduce the numerical dispersion, an extreme fine grid is required which leads to a much longer computational time. TVD scheme has the smallest numerical dispersion and energy balance error when applying the same spatial resolution, however, it presents a larger overestimation of the energy stored in the aquifer after injection periods compared with OGS. This results in an overestimation of the recovery efficiency. In addition, the large varying range of the temperature for the injected water would lead to an overshoot/undershoot in TVD scheme. Overall, OGS has the moderate numerical dispersion and can be eliminated by applying smaller time step.

Bibliography

Aravena, D., Muñoz, M., Morata, D., Lahsen, A., Parada, M. Á., & Dobson, P. (2016). Assessment of high enthalpy geothermal resources and promising areas of chile. *Geothermics*, *59*, 1–13.

Bakker, M., Post, V., Langevin, C. D., Hughes, J. D., White, J., Starn, J., & Fienen, M. N. (2016). Scripting modflow model development using python and flopy. *Groundwater*, *54*(5), 733–739.

Bennett, G. D., et al. (1995). *Applied contaminant transport modeling: theory and practice.*

Beyer, C., Hintze, M., & Bauer, S. (2016). Impacts of convection on high-temperature aquifer thermal energy storage. *EGUGA*, EPSC2016–13417.

Bloemendal, M., & Hartog, N. (2018). Analysis of the impact of storage conditions on the thermal recovery efficiency of low-temperature ates systems. *Geothermics*, *71*, 306–319.

Bloemendal, M., Olsthoorn, T., & van de Ven, F. (2015). Combining climatic and geo-hydrological preconditions as a method to determine world potential for aquifer thermal energy storage. *Science of the Total Environment*, *538*, 621–633.

Bottcher N, K. O., Watanabe N. (2015). Opengeosys tutorial. basics of heat transport processes in geothermal systems.

Bozkaya, B., Li, R., Labeodan, T., Kramer, R., & Zeiler, W. (2017). Development and evaluation of a building integrated aquifer thermal storage model. *Applied Thermal Engineering*, *126*, 620–629.

Cherubini, Y., Cacace, M., Blöcher, G., & Scheck-Wenderoth, M. (2013). Impact of single inclined faults on the fluid flow and heat transport: results from 3-d finite element simulations. *Environmental earth sciences*, *70*(8), 3603–3618.

Christopher J. Neville, S. P. . A. (n.d.). *A technical discussion of the mt3d solution methods.* [EB/OL]. (https://www.waterloohydrogeologic.com/help/vmod-flex/index.html?vm_mt3dms-solutionmethods-techrev.htm

Accessed February 8th, 2021)

COSMOL. (n.d.). *Detailed explanation of finite element method.* [EB/OL]. (https://www.comsol.jp/multiphysics/finite-element-method Accessed January 1st, 2021)

Dharma, S. (2009). Modeling of aquifer thermal energy storage (ates) using heat and solute transport in 3d (hst3d). *Civil Engineering Dimension*, *11*(2), 119–126.

Gauss, C. F. (1815). *Methodus nova integralium valores per approximationem inveniendi*. apvd Henricvm Dieterich.

Giordano, N., Comina, C., & Mandrone, G. (2015). The first italian experience of ground thermal energy storage: an integrated approach for design and monitoring, from laboratory to field scale. In *World geothermal congress, melbourne (australia)* (pp. 1–12).

Guimerà, J., Ortuño, F., Ruiz, E., Delos, A., & Pérez-Paricio, A. (2007). Influence of ground-source heat pumps on groundwater. In *Conference proceedings: European geothermal congress.*

Harbaugh, A. W. (2005). *Modflow-2005, the us geological survey modular ground-water model: the ground-water flow process*. US Department of the Interior, US Geological Survey Reston, VA.

Harbaugh, A. W., Banta, E. R., Hill, M. C., & McDonald, M. G. (2000). Modflow-2000, the u. s. geological survey modular ground-water model-user guide to modularization concepts and the ground-water flow process. *Open-file report. U. S. Geological Survey*(92), 134.

Hughes, J. D., Langevin, C. D., & Banta, E. R. (2017). *Documentation for the modflow 6 framework* (Tech. Rep.). US Geological Survey.

Kolditz, O., Bauer, S., Bilke, L., Böttcher, N., Delfs, J.-O., Fischer, T., ... others (2012). Opengeosys: an open-source initiative for numerical simulation of thermo-hydro-mechanical/chemical (thm/c) processes in porous media. *Environmental Earth Sciences*, *67*(2), 589–599.

Kolditz, O., Jakobs, L. A., Huenges, E., & Kohl, T. (2013). *Geothermal energy: a glimpse at the state of the field and an introduction to the journal.* SpringerOpen.

Kong, Y., Pang, Z., Shao, H., & Kolditz, O. (2017). Optimization of well-doublet

placement in geothermal reservoirs using numerical simulation and economic analysis. *Environmental Earth Sciences*, *76*(3), 118.

Kuhlman, K. L. (2013). Review of inverse laplace transform algorithms for laplace-space numerical approaches. *Numerical Algorithms*, *63*(2), 339–355.

Langevin, C. D., Thorne Jr, D. T., Dausman, A. M., Sukop, M. C., & Guo, W. (2008). *Seawat version 4: a computer program for simulation of multi-species solute and heat transport* (Tech. Rep.). Geological Survey (US).

Leonard, B., & Niknafs, H. (1990). *A cost-effective strategy for nonoscillatory convection without clipping* (Tech. Rep.). NASA Technical Memorandum.

Leonard-Barton, D., & Deschamps, I. (1988). Managerial influence in the implementation of new technology. *Management science*, *34*(10), 1252–1265.

Lewis, R. W., Nithiarasu, P., & Seetharamu, K. N. (2004). *Fundamentals of the finite element method for heat and fluid flow*. John Wiley & Sons.

Lin, Y.-C., Hu, T.-F., & Yeh, H.-D. (2019). Analytical model for heat transfer accounting for both conduction and dispersion in aquifers with a robin-type boundary condition at the injection well. *Water Resources Research*, *55*(8), 7379–7399.

Meng, B., Vienken, T., Kolditz, O., & Shao, H. (2018). Modeling the groundwater temperature response to extensive operation of ground source heat pump systems: A case study in germany. *Energy Procedia*, *152*, 971–977.

Müller, S., Zech, A., & Heße, F. (2020). ogs5py: A python-api for the opengeosys 5 scientific modeling package. *Groundwater*.

Nordbeck, J., Beyer, C., & Bauer, S. (2017). Experimental and numerical investigation of a scalable modular geothermal heat storage system. *Energy Procedia*, *125*, 604–611.

Paraview. (2020). http://www.paraview.org.

Scipy. (2020). https://docs.scipy.org/doc/scipy/reference/tutorial/integrate.html.

Seibt, P., & Kellner, T. (2003). Practical experience in the reinjection of cooled thermal waters back into sandstone reservoirs. *Geothermics*, *32*(4-6), 733–741.

Simpson(Formula). (2001). Encyclopedia of mathematics. *EMS Press*.

Trefry, M. G., & Muffels, C. (2007). Feflow: A finite-element ground water flow

and transport modeling tool. *Groundwater*, *45*(5), 525–528.

Vélez, M. I., Blessent, D., Córdoba, S., López-Sánchez, J., Raymond, J., & Parra-Palacio, E. (2018). Geothermal potential assessment of the nevado del ruiz volcano based on rock thermal conductivity measurements and numerical modeling of heat transfer. *Journal of South American Earth Sciences*, *81*, 153–164.

Volpi, G., Magri, F., Colucci, F., Fisher, T., De Caro, M., & Crosta, G. B. (2018). Modeling highly buoyant flows in the castel giorgio: Torre alfina deep geothermal reservoir. *Geofluids*, *2018*.

Wikipedia contributors. (2020). *Gaussian quadrature — Wikipedia, the free encyclopedia.* Retrieved from https://en.wikipedia.org/w/index.php?title=Gaussian_quadrature&oldid=996491671 ([Online; accessed 28-December-2020])

Zeghici, R. M., Essink, G. H. O., Hartog, N., & Sommer, W. (2015). Integrated assessment of variable density–viscosity groundwater flow for a high temperature mono-well aquifer thermal energy storage (ht-ates) system in a geothermal reservoir. *Geothermics*, *55*, 58–68.

Zheng, C., Wang, P. P., et al. (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.

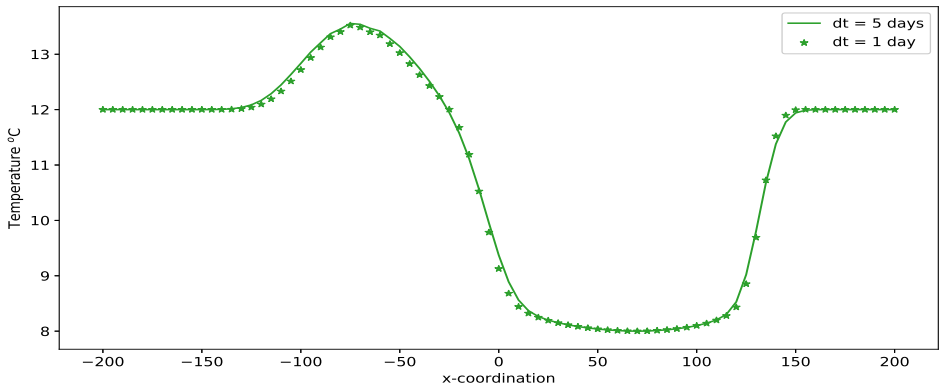
A | Implementing a doublet in OpenGeoSys

The committed steps of implementation of a doublet in OGS is given below. The RFD object allows user to define the time-variables curve. In the BC object, the boundary condition would be activated if the corresponding variable value at a certain time is 1 and inactivated if it's 0. By defining the curve as shown in the script, the boundary condition would be activated between 0 to 180 days for the warm well and inactivated for the rest of the time while it's reversed for the cold well. In addition, the TIM_TYPE in the ST object allows user to control the flow direction. The flow rate applied in the simulation is the value that computed by the constant value in DIS_TYPE multiplying the corresponding value in the time-variables curve at the certain time.

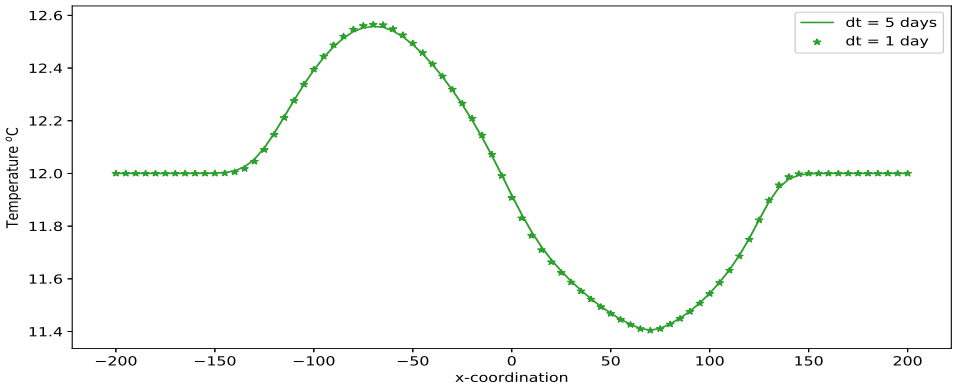
```
model.gli.add_polyline("warmwell", \
        [[-70, 0, -20], [-70, 0, -35]])
model.gli.add_polyline("coldwell", \
          [[70, 0, -20], [70, 0, -35]])
time = [0, 180*24*3600, 180*24*3600+0.1, \
          360*24*3600]
T_w = [1, 1, 0, 0]
T_c = [0, 0, 1, 1]
F_w = [1, 1, -1, -1]
F_c = [-1, -1, 1, 1]
Boundary_warm = zip(time, T_w)
Boundary_cold = zip(time, T_c)
Stress_warm = zip(time, F_w)
Stress_cold = zip(time, F_c)
model.rfd.add_block(CURVES = Boundary_warm)
model.rfd.add_block(CURVES = Boundary_cold)
model.rfd.add_block(CURVES = Stress_warm)
model.rfd.add_block(CURVES = Stress_cold)
model.bc.add_block(
     main_key = "BOUNDARY_CONDITION",
```

```
21        PCS_TYPE  =  "HEAT_TRANSPORT",
22        PRIMARY_VARIABLE="TEMPERATURE1",
23        GEO_TYPE=["POLYLINE",  'warmwell'],
24        DIS_TYPE  =  ["CONSTANT",  16.0],
25        TIME_CONTROLLED_ACTIVE  =  1,
26    )
27    model.bc.add_block(
28        main_key  =  "BOUNDARY_CONDITION",
29        PCS_TYPE  =  "HEAT_TRANSPORT",
30        PRIMARY_VARIABLE="TEMPERATURE1",
31        GEO_TYPE=["POLYLINE",  'coldwell'],
32        DIS_TYPE  =  ["CONSTANT",  8.0],
33        TIME_CONTROLLED_ACTIVE  =  2,
34    )
35    model.st.add_block(   # source term
36        main_key='SOURCE_TERM',
37        PCS_TYPE="GROUNDWATER_FLOW",
38        PRIMARY_VARIABLE="HEAD",
39        GEO_TYPE=["POLYLINE",  "warmwell"],
40        DIS_TYPE=["CONSTANT_NEUMANN",  flow/30],
41        TIM_TYPE  =  ["CURVE",  3]
42    )
43
44    model.st.add_block(   # source term
45        main_key='SOURCE_TERM',
46        PCS_TYPE="GROUNDWATER_FLOW",
47        PRIMARY_VARIABLE="HEAD",
48        GEO_TYPE=["POLYLINE",  "coldwell"],
49        DIS_TYPE=["CONSTANT_NEUMANN",  flow/30],
50        TIM_TYPE  =  ["CURVE",  4]
51    )
```

B | Spatial and temporal resolution adjustment in clay-sand-clay model

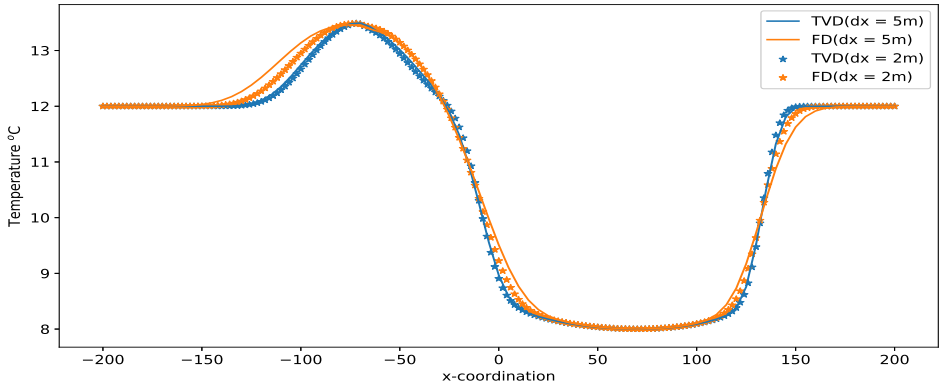


(a) aquifer

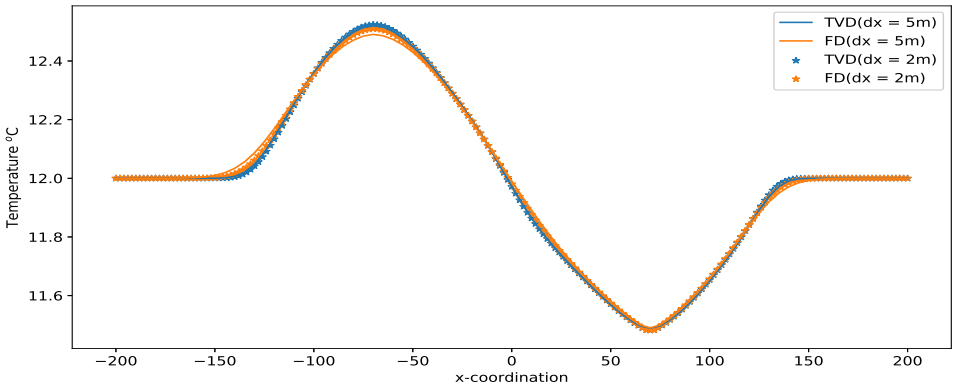


(b) aquitard

Figure B.1: Temperature distribution along the observation line after temporal resolution adjustment in OGS.



(a) aquifer



(b) aquitard

Figure B.2: Temperature distribution along the observation line after spatial resolution adjustment in SEAWAT.

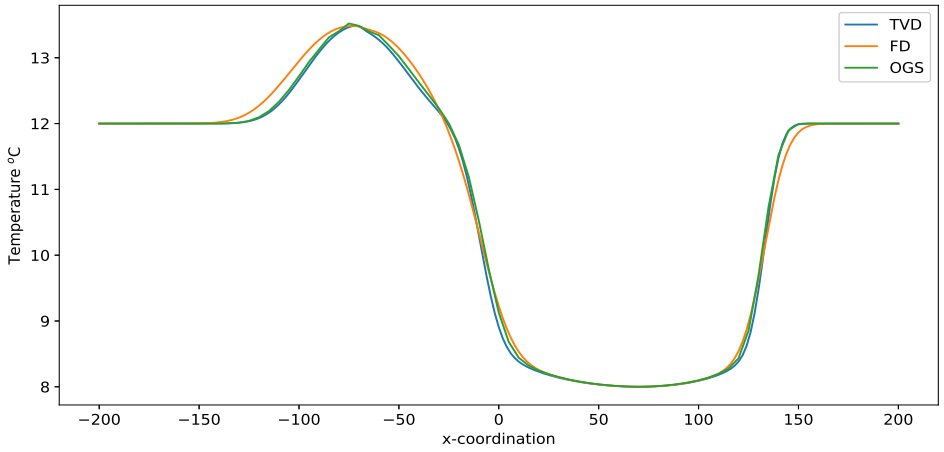


Figure B.3: Temperature distribution for numerical models after spatial/temporal resolution adjustment.