

UACM

Universidad Autónoma
de la Ciudad de México

NADA HUMANO ME ES AJENO

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN MODELACIÓN MATEMÁTICA

Estudio de un modelo de transporte de especies de nitrógeno en acuíferos

TESIS

QUE PARA OPTAR POR EL TÍTULO DE

LICENCIADA EN MODELACIÓN MATEMÁTICA

PRESENTA

GUADALUPE MARTÍNEZ ORTEGA

DIRECTORA: **DRA. ROSA MARGARITA ÁLVAREZ GONZÁLEZ**

CODIRECTORA: **DRA. GRACIELA DEL SOCORRO HERRERA ZAMARRÓN**

Ciudad de México, junio de 2025.

SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN



UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO COORDINACIÓN ACADÉMICA

RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

DERECHOS RESERVADOS[©]

La presente obra y cada uno de sus elementos está protegido por la Ley Federal del Derecho de Autor; por la Ley de la Universidad Autónoma de la Ciudad de México, así como lo dispuesto por el Estatuto General Orgánico de la Universidad Autónoma de la Ciudad de México; del mismo modo por lo establecido en el Acuerdo por el cual se aprueba la Norma mediante la que se Modifican, Adicionan y Derogan Diversas Disposiciones del Estatuto Orgánico de la Universidad de la Ciudad de México, aprobado por el Consejo de Gobierno el 29 de enero de 2002, con el objeto de definir las atribuciones de las diferentes unidades que forman la estructura de la Universidad Autónoma de la Ciudad de México como organismo público autónomo y lo establecido en el Reglamento de Titulación de la Universidad Autónoma de la Ciudad de México.

Por lo que el uso de su contenido, así como cada una de las partes que lo integran y que están bajo la tutela de la Ley Federal de Derecho de Autor, obliga a quien haga uso de la presente obra a considerar que solo lo realizará si es para fines educativos, académicos, de investigación o informativos y se compromete a citar esta fuente, así como a su autor ó autores. Por lo tanto, queda prohibida su reproducción total o parcial y cualquier uso diferente a los ya mencionados, los cuales serán reclamados por el titular de los derechos y sancionados conforme a la legislación aplicable.

Dedicatoria

A Dios
A mis padres
A mis hermanos

Agradecimientos

Mis más sinceros agradecimientos a:

- ◇ Mi alma máter, la Universidad Autónoma de la Ciudad de México (UACM). A lo largo de estos años, la UACM ha sido un espacio de crecimiento no sólo académico, sino también personal y humano.
- ◇ El Instituto de Geofísica de la Universidad Nacional Autónoma de México (UNAM) por haberme abierto las puertas y permitirme desarrollar parte de este trabajo en el proyecto **Ciencia de Frontera**. En especial quiero agradecer a los Drs. Guillermo Hernández, Iván Contreras, Mario Hernández y Norberto Vera por su apoyo, paciencia, orientación y disposición para compartir su conocimiento. Sus aportaciones fueron fundamentales para la consolidación de este proyecto.
- ◇ El proyecto financiado por el Fondo Ciencia de Frontera (SECIHTI), con número de convenio CF-2023-G-904, en cuyo marco fue posible desarrollar el trabajo de investigación que dio origen a esta tesis.
- ◇ Mi directora, la Dra. Rosa Margarita Álvarez, le expreso mi más profundo agradecimiento y admiración. Su guía académica, su constante apoyo y su confianza en mí fueron fundamentales en cada etapa de esta tesis. Gracias por alentarme a salir de mi zona de confort y por enseñarme a creer en mi trabajo. Gracias por estar presente también en lo personal, por escucharme, orientarme y acompañarme con calidez en los momentos difíciles.
- ◇ Mi codirectora, la Dra. Graciela Herrera por hacerme parte de su equipo de trabajo, por su guía académica, sus valiosas observaciones y por contribuir con su experiencia al desarrollo de este trabajo.
- ◇ Mis lectores, los Drs. Roberto Murillo y Marcos Ángel González por darse a la tarea de leer y acompañar esta tesis. Sus comentarios y observaciones ayudaron a enriquecer este trabajo. Gracias por brindarme su tiempo. En especial, quiero agradecer a la Dra. Rita Vázquez, quien ha estado presente desde el inicio de mi formación. Su compañía constante y sus palabras de aliento fueron fundamentales para que no abandonara este camino. Gracias por enseñarme, en muchos momentos sin saberlo, a creer más en mí.
- ◇ Mis profesores y profesoras, por su entrega, su pasión por la docencia y por ser parte esencial de este recorrido formativo. Los llevó en el corazón.
- ◇ Mis padres, Pablo y Vicky, por darme la oportunidad de continuar con mis estudios, por confiar en mí incluso cuando yo misma dudaba. Gracias por su apoyo incondicional, por estar presentes todos los días, a veces en gestos pequeños, otras veces en palabras necesarias, y por sostenerme en

los momentos en que sentía que no podía más. Su amor constante, su paciencia y su fe en mí fueron una base firme sobre la cual pude construir este camino. Esta tesis es también el resultado de su esfuerzo, de los sacrificios que hicieron, y del cariño con el que siempre me acompañaron. Gracias por enseñarme con el ejemplo el valor del compromiso y la perseverancia.

- ◇ A Manuel y Aarón, por amarme a su manera y por hacerme reír incluso en los días más difíciles. Su presencia fue y sigue siendo un refugio, una fuente de alegría y de fuerza. Gracias por los momentos compartidos, por acompañarme sin pedir explicaciones, y por recordarme, sin decirlo, que nunca estoy sola. Son, sin duda, los mejores hermanos que la vida me dio.
- ◇ Mis amigos, quienes con su compañía, palabras, risas, apoyo incondicional y tardes en “el roof” hicieron más ligero este camino. Gracias por estar en los días buenos y, sobre todo, en los complicados. Su presencia ha sido un verdadero sostén emocional y una fuente constante de motivación. Aunque no mencione sus nombres, cada uno de ustedes tiene un lugar especial en este logro y en mi vida.
- ◇ A Cris, Marisol, Audy y Erick, gracias por ser esa mezcla perfecta entre terapia, fiesta y familia. Gracias por sacarme de la rutina con salidas que empezaban como “una pijamada” y terminaban siendo el mejor antídoto contra el estrés universitario. Pero sobre todo, gracias por sus consejos, por sus palabras de aliento cuando más las necesitaba, y por estar ahí con el corazón abierto, listos para escuchar o para hacerme reír hasta que se me olvidara el agobio. Tenerlos cerca fue un recordatorio constante de que no todo en la vida es tesis. . . y eso también salvó este proceso.
- ◇ A Ingrid y Yaz, dos personas que han sido un pilar fundamental en mi vida. Ingrid, desde que te conocí, has llenado mi vida de alegría y cariño. Gracias por regalarme la experiencia maravillosa de ser tía de Dorian, nuestro príncipe azul, un pequeño que llegó para llenarnos de amor. Y a mi querida Yazmín, mi mejor amiga desde hace 19 años, gracias por estar siempre a mi lado en cada etapa, por las risas, las lágrimas y la complicidad que solo nosotras entendemos. Nuestra amistad es un tesoro que valoro con todo mi corazón. Gracias a ambas por ser la familia que elegí y por acompañarme en este camino. Las quiero muchísimo.
- ◇ Eva González, por brindarme la oportunidad y facilidad para adquirir una computadora que fue fundamental durante mi estancia en la universidad. Ese apoyo hizo una gran diferencia en mi camino académico y siempre estaré muy agradecida por ello.
- ◇ Erika Bermudez y mi hermanita Fer. Gracias por hacerme parte de su familia y por quererme tanto.
- ◇ A mi hada madrina, Sonia Benitez, gracias por apoyarme emocionalmente en los momentos difíciles y por ser una verdadera inspiración con tu de-

dicación y compromiso en todo lo que haces. Tu ejemplo me motiva cada día a seguir adelante.

Gracias a todas y todos por ser parte de mi vida y acompañarme en este viaje.

Índice general

	Página
Introducción	9
Objetivos	12
1. Fundamentos del flujo en medios porosos.	13
1.1. Conceptos básicos de hidrogeología.	13
1.2. Ley de Darcy	17
2. Fundamentos del transporte de solutos en medios porosos.	20
2.1. Advección	21
2.2. Difusión	22
2.3. Dispersión	23
2.4. Transporte reactivo	25
3. Modelos de flujo y de transporte en medios porosos saturados.	27
3.1. Ecuación de flujo.	27
3.2. Ecuación de transporte advección-dispersión.	30
3.3. Ecuación advección-dispersión-reacción.	31
3.4. Condiciones iniciales y de frontera.	32
4. Métodos de solución numérica de los modelos de flujo y transporte.	34
4.1. Ideas básicas del método de diferencias finitas.	34
4.2. Herramientas de cómputo utilizadas para la resolución numérica de algunos ejemplos.	38
4.2.1. MODFLOW 6	39
4.2.2. MT3D-USGS	40
4.2.3. Flopy	40
5. Presentación y resolución de ejemplos de flujo y de transporte en medios porosos saturados.	42
5.1. Ejemplo de flujo estacionario en una dimensión espacial.	43
5.1.1. Modelo conceptual.	43
5.1.2. Modelo matemático.	44
5.1.3. Simulación del flujo con FloPy-MODFLOW 6	46

5.2. Ejemplo de flujo y transporte en una dimensión espacial.	57
5.2.1. Modelo conceptual	57
5.2.2. Modelo matemático	58
5.2.3. Implementación en MODFLOW 6-FloPy	58
5.3. Ejemplo de transporte advección-dispersión con decaimiento de primer orden.	66
5.3.1. Elementos básicos del ciclo del nitrógeno.	66
5.3.2. Modelo conceptual	67
5.3.3. Modelo matemático	68
5.3.4. Implementación en MODFLOW 6 y MT3D-USGS	69
Conclusiones	80
Anexos	82

Índice de figuras	95
--------------------------	-----------

Introducción

A lo largo de nuestra historia el ser humano ha creado un vínculo muy importante con uno de los cuatro elementos de la naturaleza, el agua. Además de ser el líquido vital para los seres vivos, en muchas culturas es considerada como un elemento sagrado y ha estado presente en el crecimiento y desarrollo de las civilizaciones en el mundo. En la actualidad, el agua sigue siendo uno de los recursos más importantes en el desarrollo económico, ambiental y social.

Según estudios realizados por el Instituto Mexicano para la Competitividad (IMCO), “los recursos hídricos en México enfrentan distintos retos dependiendo de su tipo (superficial o subterráneos)”[19]. Es importante mencionar que ambos tipos juegan un papel clave para la sociedad, sin embargo en el presente trabajo, nos enfocaremos al estudio del agua subterránea (acuíferos).

Los acuíferos son formaciones geológicas en el subsuelo que almacenan agua y permiten que cantidades significativas de tal líquido circulen y se acumulen en ellos a través de sus poros o grietas. En México existen alrededor de 653 acuíferos y se estima que el 70 % del agua que se utiliza en la Ciudad de México proviene de ellos.[9]

La investigación presentada por el IMCO [19] indica que en el año 2018 el 18 % de los acuíferos mexicanos estaban sobreexplotados, afectando tanto el abastecimiento en hogares como las actividades agropecuarias e industriales. Aunado a estas problemáticas sociales y económicas se presenta también la crisis ambiental en torno al agua.

La contaminación de los acuíferos es un problema ambiental significativo que afecta la calidad del agua y la salud humana. Entre los contaminantes más preocupantes se encuentra el nitrógeno, el cual proviene de diversas fuentes como: la industria, los vertidos urbanos, la materia fecal, la agricultura y la ganadería, siendo estas últimas las que más impactan en la actualidad.

El consumo de agua contaminada por nitrógeno afecta la salud humana, principalmente a los bebés menores de seis meses provocando metahemoglobinemia, también conocida como “síndrome del bebé azul”[27].

Adicionalmente, cuando la calidad del agua subterránea es afectada por nitrógeno el daño se expande en gran parte del territorio en donde se encuentra. Debido a ciertas características que los acuíferos poseen las sustancias contaminantes quedan retenidas en el subsuelo y la posible recuperación de la calidad del acuífero necesita periodos muy largos de tiempo y el costo es elevado, por lo que de manera frecuente se opta por el abandono de la región, causando que el desarrollo ecológico y económico se vean perjudicados.

El comportamiento del nitrógeno en los acuíferos está determinado por procesos físicos, químicos y biológicos complejos, incluyendo el flujo del agua, la dispersión y difusión de los solutos, y diversas reacciones químicas. En este contexto, el uso de modelos matemáticos y herramientas computacionales resulta fundamental para analizar, predecir y gestionar la dinámica del nitrógeno en medios porosos saturados.[31]

Estos modelos permiten representar el flujo del agua subterránea y el transporte de contaminantes de manera cuantitativa, facilitando la toma de decisiones informadas para la protección de los recursos hídricos. Además, los modelos numéricos son una herramienta poderosa para evaluar escenarios hipotéticos, simular procesos en condiciones no observables directamente, y estimar tiempos de recuperación de los acuíferos afectados.[3]

En este proyecto se estudiarán modelos matemáticos de transporte de especies de solutos, revisando los fundamentos teóricos y mostrando su comportamiento con el diseño y puesta a prueba de experimentos numéricos específicos.

Para comenzar, en el primer capítulo de este trabajo se describen los conceptos fundamentales que permiten sentar las bases para el estudio del comportamiento del flujo de fluidos en medios porosos.

En el segundo capítulo se estudian los elementos básicos en los procesos de transporte de solutos a través de medios porosos. Durante la revisión teórica nos referiremos al transporte de solutos en general y no sólo al transporte de contaminantes, ya que en la práctica hay presencia de sustancias que no afectan al medio por lo que no se consideran contaminantes.

El capítulo tres está dedicado al análisis de los modelos matemáticos de flujo y de transporte en medios porosos saturados, los cuales se componen de ecuaciones diferenciales parciales.

En el capítulo cuatro se expone el método de diferencias finitas utilizado para resolver ecuaciones diferenciales parciales de primer y segundo orden. También se describen las herramientas computacionales utilizadas para la solución numérica de problemas de flujo y transporte en aguas subterráneas.

El capítulo cinco contiene tres ejemplos de flujo y de transporte en medios

porosos saturados, incluyendo el modelo conceptual, el modelo matemático y los resultados obtenidos con Modflow 6 y MT3D-USGS.

Finalmente se presentan las conclusiones obtenidas como resultado del desarrollo de esta tesis; los códigos utilizados para realizar las simulaciones de los ejemplos del capítulo cinco y el índice de las figuras .

Objetivos

Objetivo general.

Estudiar los fundamentos teóricos de los modelos matemáticos de flujo en medios porosos y del transporte de especies de nitrógeno en estos sistemas, con el objetivo de facilitar su comprensión e implementación en la resolución numérica.

Objetivos Específicos

- Analizar y dar al lector la comprensión de la importancia del ciclo de nitrógeno y de las ecuaciones que describen su transporte en medios porosos.
- Seleccionar el modelo en el que se trabajará, que describe el flujo de agua y el transporte del contaminante. Revisar el fundamento de dichas ecuaciones.
- Resolver algunos ejemplos de modelos donde se pueda obtener la solución analítica (carga hidráulica y concentración del contaminante)
- Resolver de manera numérica con el método de diferencias finitas, implementado en los software MODFLOW 6 y MT3D-USGS, algunos de los modelos teóricos, usando lenguajes de programación apropiados como Python o Matlab.

Capítulo 1

Fundamentos del flujo en medios porosos.

1.1. Conceptos básicos de hidrogeología.

Para entender el comportamiento de un fluido es necesario tomar en cuenta el medio en el que se traslada. Entre los ambientes más comunes se encuentran los canales abiertos como los ríos, o cerrados como tuberías y conductos, como los sistemas de plomería. Por otro lado, están los medios porosos, como suelos o materiales que contienen espacios vacíos que permiten el flujo a través de ellos.

Dado que el objetivo principal en este capítulo es estudiar, analizar y comprender los procesos de flujo en medios porosos, a continuación se describen conceptos fundamentales de hidrogeología.

Acuífero: es una formación geológica que almacena agua y permite que cantidades significativas de tal líquido circulen con facilidad a través de sus poros o grietas. Existen acuíferos en medios porosos o en medios fracturados. En este trabajo nos enfocamos en los que se encuentran en medios porosos.

Superficie freática: es la superficie formada (en un acuífero) por los puntos con una presión igual a la presión atmosférica. Bajo ella, todos los poros del acuífero están llenos de agua, sobre ella los poros pueden estar secos, húmedos o parcialmente llenos de agua. El agua que se encuentra debajo de la superficie freática se considera *agua subterránea*.

Existen diferentes tipos de acuíferos y se clasifican según su nivel de presión.

- **Acuífero confinado:** es un acuífero en el que el agua subterránea está contenida bajo presión, limitada por formaciones geológicas de baja per-

meabilidad tanto por encima como por debajo, lo que impide su contacto directo con la atmósfera.

- **Acuífero no confinado (o freático):** es un acuífero que no está confinado por formaciones de baja permeabilidad en su parte superior; su límite superior lo define la superficie freática, y el agua subterránea se encuentra en contacto con el aire del suelo no saturado, por lo que está a presión atmosférica.
- **Acuífero semiconfinado:** es un acuífero en el que el agua subterránea se encuentra bajo presión, similar al caso de los acuíferos confinados, pero la diferencia principal es que está limitado por capas de baja permeabilidad que permiten cierto flujo vertical, es decir, no son completamente impermeables.

A continuación se presentan las características más relevantes de un medio poroso, indicando su definición y en el caso de algunas de ellas sus unidades de medida, estas unidades se expresan de manera general con la siguiente notación: longitud (L), tiempo (T) y masa (M).

Porosidad (θ): es el total de espacios vacíos del suelo que permiten el flujo de agua y se calcula de la siguiente manera[17]:

$$\theta = \frac{V_v}{V_T}$$

donde

θ : porosidad

V_v : volumen de los espacios vacíos (L^3)

V_T : volumen total del suelo (L^3)

Tortuosidad: es la relación entre la longitud real que recorre el fluido en un medio poroso y el espesor del medio:[13]

$$\tau = \frac{\text{longitud real recorrida}}{\text{longitud en línea recta}}$$

El concepto de tortuosidad es inconsistente en la bibliografía. En algunos casos se considera el cociente como lo hemos referido y en otras ocasiones el cuadrado de dicho cociente. En todos los casos es $\tau > 1$. [28]

Permeabilidad intrínseca (k): es la capacidad del medio poroso para transmitir fluidos. Esta propiedad depende sólo del medio, es decir, es independiente del fluido.

Conductividad hidráulica (K): Expresa la facilidad con la cual un fluido

se transporta a través de un medio poroso, por lo que este coeficiente es una propiedad que depende del medio y del fluido:

$$K = \frac{k\rho g}{\mu}$$

donde

K : conductividad hidráulica (L/T)

k : permeabilidad intrínseca (L^2)

μ : viscosidad del fluido (M/LT)

g : fuerza de gravedad (L/T^2)

ρ : densidad del fluido (M/L^3)

La densidad del fluido es una propiedad física que permite medir la cantidad de sustancia en un volumen específico, es decir, *densidad = masa/volumen*:

$$\rho = M/V$$

y será útil en capítulos posteriores para el entendimiento de la ecuación de flujo.

Homogeneidad y heterogeneidad: se dice que un medio poroso es homogéneo si sus propiedades físicas y químicas son las mismas en toda la formación del acuífero, es decir, son independientes de la posición. En otro caso, cuando dichas propiedades son dependientes de la posición dentro del acuífero, el medio se dice heterogéneo.

Isotropía y anisotropía: un medio poroso es isotrópico cuando sus propiedades físicas (por ejemplo, porosidad o permeabilidad) no cambian en ninguna dirección. Si las propiedades físicas varían en diferentes direcciones el medio es anisotrópico.

Zona saturada: es aquella en que los poros están completamente llenos de agua. En esta zona, la presión a la que se encuentra el agua es superior a la de la atmósfera.

Recarga de agua: es el proceso mediante el cual el agua superficial, proveniente principalmente de la infiltración de precipitaciones, ríos, lagos u otras fuentes, penetra y alimenta un acuífero.

Carga hidráulica (h): al realizarse una perforación en un acuífero confinado, como el que nos ocupa en este trabajo, se podrá observar cómo el nivel de agua comienza a subir en la columna generada por dicha perforación. La altura (medida desde un punto de referencia arbitrario) que alcanza el agua en la columna es denominada **carga hidráulica, nivel piezométrico**.

Coefficiente de almacenamiento (S): Indica la relación entre el cambio en la cantidad de agua almacenada en un acuífero y el cambio en las elevaciones de la superficie piezométrica:

$$S = \frac{\text{Volumen de agua liberado}}{\text{Volumen total que ha bajado la superficie piezométrica}}$$

Este coeficiente representa la capacidad del acuífero para almacenar o liberar agua por unidad de cambio en la carga hidráulica.

Almacenamiento específico (S_s): es el volumen de agua liberado (L^3) por unidad de descenso en la carga hidráulica (L).

Descarga específica (q): es el volumen de agua que fluye por unidad de tiempo a través de una unidad de área de sección transversal normal a la dirección del flujo.[7]

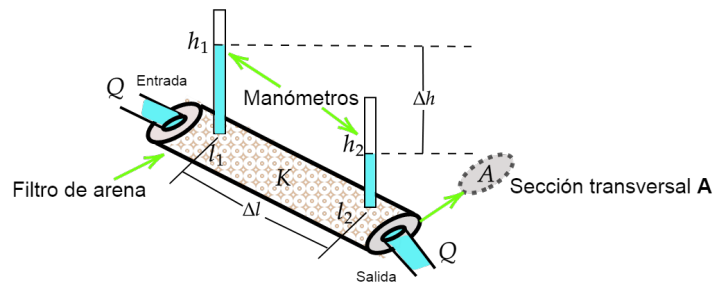
Ahora que se han descrito los conceptos base que serán empleados para el análisis del flujo en aguas subterráneas, continuaremos con la justificación matemática que modela los procesos físicos dentro de un medio poroso.

1.2. Ley de Darcy

La ley de Darcy es una de las principales herramientas utilizadas para modelar y predecir el comportamiento de un fluido en aguas subterráneas. En esta sección se presentará una breve descripción sobre los experimentos realizados por Darcy, los cuales lo llevaron a la formulación matemática de esta ley.

En 1856, el ingeniero Henry Darcy (1803-1858) desarrolló una investigación sobre el abastecimiento de agua en la ciudad de Dijon, Francia. Durante esta investigación, Darcy se interesó en el movimiento del agua a través de medios porosos, por lo que realizó diversas pruebas que lo llevaron a concluir su famosa ley.

La siguiente figura se creó con la intención de explicar de manera sencilla la ley formulada por Darcy.



$$Q = \frac{KA(h_1 - h_2)}{(l_2 - l_1)} = KA \frac{\Delta h}{\Delta l}$$

Figura 1.1: Diagrama de la ley de Darcy y su expresión matemática

Como podemos observar en la figura 1.1, los experimentos hechos por Darcy, consistían en utilizar un cilindro de sección transversal de área A , lleno de algún tipo de material poroso (arcilla, grava, arena, etc.), tapado en cada extremo, con tubos de entrada y salida de flujo y un par de manómetros.¹

El agua se introduce en el cilindro y esta comienza a fluir hasta que todos los poros del material dentro del cilindro se cubran en su totalidad y la tasa de entrada de flujo Q sea igual a la tasa de salida. Se considera h_1 y h_2 como los niveles de elevación del fluido y $(l_2 - l_1) = \Delta l$ como la distancia entre cada manómetro.

El resultado de los experimentos de Darcy fue el siguiente:

¹Estos instrumentos permiten medir la presión de un fluido dentro de un recipiente cerrado.

1. $Q \propto A$: El volumen de agua por unidad de tiempo Q es directamente proporcional al área de la sección transversal A .
2. $Q \propto \Delta h$: El volumen de agua por unidad de tiempo Q es directamente proporcional a la diferencia en las elevaciones del nivel del agua, $(h_1 - h_2) = \Delta h$.
3. $Q \propto \frac{1}{\Delta l}$: El volumen de agua por unidad de tiempo Q es inversamente proporcional a la distancia entre cada manómetro, $(l_2 - l_1) = \Delta l$.

Las tres conclusiones anteriores se pueden expresar en una sola ecuación y para poder llegar a ella es necesario utilizar una constante de proporcionalidad que nos permita incluir una igualdad y eliminar el símbolo de proporción. En este caso esa constante es la conductividad hidráulica K que, como se mencionó en la sección anterior, es una propiedad del medio.

Por otro lado, se debe considerar que el nivel de h disminuye en sentido del flujo por lo que el gradiente hidráulico $\left(\frac{\Delta h}{\Delta l}\right)$ es negativo, en consecuencia, será necesario agregar un signo $-$ para que el caudal Q sea positivo.

De esta forma, la ley de Darcy puede escribirse de forma matemática de la siguiente manera:

$$Q = -KA \left(\frac{\Delta h}{\Delta l} \right) \quad (1.1)$$

Utilizando la definición de **descarga específica**:

$$q = \frac{Q}{A}$$

y aplicando el límite al gradiente hidráulico y utilizando la definición de derivada:

$$\lim_{\Delta l \rightarrow 0} \frac{\Delta h}{\Delta l} = \frac{dh}{dl}$$

podemos reescribir la ecuación (1.1) como:

$$q = -K \frac{dh}{dl} \quad (1.2)$$

siendo esta última ecuación la manera matemáticamente correcta de expresar la **ley de Darcy**.

Cabe destacar que esta ley es válida en un medio saturado, homogéneo e isótropo. Es importante tomar en cuenta que también existen algunas limitaciones para la ley de Darcy, por lo que puede ser menos precisa en condiciones turbulentas, descuida los efectos térmicos relevantes en aplicaciones de transferencia de calor y es menos ideal para analizar escenarios de flujo multifásico.

Como se describió en esta sección, la ley de Darcy es un principio fundamental en la hidrogeología que describe el flujo de fluidos a través de medios porosos. La ley de Darcy junto con la ecuación de conservación de masa permite construir la ecuación de flujo de aguas subterráneas, la cual se describirá en el capítulo 3.

La ecuación de flujo es necesaria para poder resolver la ecuación de transporte por lo que a continuación se describen los procesos más importantes que intervienen en el transporte de solutos en medios porosos.

Capítulo 2

Fundamentos del transporte de solutos en medios porosos.

Para evaluar el impacto de un soluto en el agua subterránea es fundamental identificar su ubicación, el tiempo que tarda en desplazarse a través del medio poroso, y su concentración en función de la distancia x y del tiempo t , a partir del lugar y momento en que ingresa al sistema.

Por lo que, en el desarrollo de este capítulo se expondrán los fundamentos que intervienen en el proceso del transporte de un soluto en un medio poroso en el que no existen reacciones de ningún tipo con otros solutos o con el mismo medio.

Antes de describir los procesos que intervienen en el transporte de solutos, se debe considerar el término velocidad lineal media. Para explicarlo retomaremos lo visto en el capítulo 1 sobre la ley de Darcy.

Velocidad lineal media: Recordemos que a partir de la ley de Darcy, se cumple que $q = \frac{Q}{A}$ sin embargo, la velocidad q obtenida a partir de esta ecuación no es precisamente la velocidad real que un fluido tiene dentro del filtro de arena utilizado para los experimentos hechos por Darcy y esto se debe a que el fluido circula sólo por una pequeña parte del filtro y no por toda el área de la sección transversal. A esta velocidad q le llamaremos *velocidad de Darcy*.

Por otro lado, un término significativo dentro de la *velocidad lineal media* es la porosidad eficaz θ_e , que indica la parte de la sección total por la que puede circular un fluido.

Ahora que tenemos presentes estos dos conceptos, la *velocidad lineal media* se

puede expresar como:

$$\text{Velocidad lineal media} = \frac{\text{Velocidad de Darcy}}{\theta_e} \quad (2.1)$$

En la práctica, se utiliza la expresión (2.1) para calcular el tiempo de recorrido del agua subterránea de un punto a otro.

De manera conjunta con la velocidad lineal media, existen otros procesos físicos y químicos que intervienen en la descripción del transporte de solutos en medios porosos, los cuales se describen a continuación.

2.1. Advección

La advección es el proceso mediante el cual un soluto es transportado por el flujo del agua subterránea. Si consideráramos que durante el transporte sólo interviene el proceso de advección entonces el contaminante viajaría a la misma velocidad que el agua y su extensión sería constante.

En un medio poroso, el flujo de masa por advección se expresa como:

$$J_{adv} = \theta_e v C \quad (2.2)$$

donde

J_{adv} : flujo de masa por advección (M/L^2T)
 θ_e : porosidad eficaz
 C : concentración (M/L^3)
 v : velocidad lineal media (L/T)

Como se mencionó en el capítulo 1, el flujo a través de un medio poroso se describe por la ley de Darcy y utilizando las ecuaciones (1.2) y (2.1):

$$q = \theta_e v = -K \nabla h \quad (2.3)$$

donde

q : velocidad de Darcy
 θ_e : porosidad eficaz
 v : velocidad lineal media
 K : conductividad hidráulica
 ∇h : gradiente de la carga hidráulica.

Sustituyendo la ecuación (2.3) en (2.2), tenemos:

$$J_{adv} = qC \quad (2.4)$$

Esta ecuación representa el transporte causado por la advección.

2.2. Difusión

La **difusión molecular**, o simplemente **difusión**, es el proceso mediante el cual un soluto se desplaza desde una zona de mayor concentración hacia una de menor concentración, debido al movimiento aleatorio de las moléculas. En un medio completamente líquido, el flujo de masa asociado a este proceso se describe mediante la primera ley de Fick, la cual establece que el flujo es proporcional al gradiente de concentración:

$$J_{dif} = -D_m \frac{dC}{dx} \quad (2.5)$$

donde

J_{dif} : flujo de masa por difusión (M/L^2T)

D_m : coeficiente de difusión (L^2/T)

C : concentración del soluto (M/L^3)

x : distancia (L)

Cuando la difusión ocurre en un medio poroso, como un acuífero, es necesario considerar propiedades adicionales del medio, en particular la porosidad eficaz y la tortuosidad.

Estas características modifican el camino que sigue el soluto, reduciendo la eficiencia de la difusión con respecto a un medio puramente líquido. Para tomar esto en cuenta, se sustituye el coeficiente de difusión molecular por un coeficiente de difusión efectiva D^* , el cual incorpora dichos efectos. La ecuación (2.5) se transforma entonces en: (2.5) se reformula de la siguiente manera:

$$J_{dif} = -D^* \frac{dC}{dx} \quad (2.6)$$

donde

J_{dif} : flujo de masa por difusión (M/L^2T)

D^* : coeficiente de difusión efectiva o coeficiente de difusión en un medio poroso (L^2/T)

C : concentración del contaminante (M/L^3)

x : distancia (L)

La ecuación (2.6) representa la **primera ley de Fick para un medio poroso** y el coeficiente de difusión efectiva se puede calcular de diversas formas, una de ellas es la propuesta por Charles W. Fetter:

$$D^* = (D_m)(\omega)$$

donde

D^* : Coeficiente de difusión efectiva

D_m : Coeficiente de difusión

ω : Coeficiente que depende del medio poroso y tiene valores entre 0.01 y 0.5

La primera ley de Fick considera la difusión en un estado estacionario, por tanto si se desea analizar el proceso de difusión con variación en el tiempo se da lugar a la **segunda ley de Fick** que se expresa de la siguiente forma:

$$\frac{\partial C}{\partial t} = -\frac{\partial J_{dif}}{\partial x} \quad (2.7)$$

Sustituyendo la ecuación (2.6) en (2.7):

$$\frac{\partial C}{\partial t} = D_m \frac{\partial^2 C}{\partial x^2} \quad (2.8)$$

donde

D_m : coeficiente de difusión (L^2/T)

C : concentración del soluto (M/L^3)

x : distancia (L)

t : tiempo (T)

Una forma de escribir la solución para esta ecuación en un medio poroso fue propuesta por Crank:

$$C(x, t) = C_0 \left(\operatorname{erfc} \frac{x}{2\sqrt{D^*t}} \right) \quad (2.9)$$

donde

$C(x, t)$: Concentración a una distancia x transcurrido un tiempo t

C_0 : concentración inicial (M/L^3)

D^* : coeficiente de difusión efectiva (L^2/T)

erfc : función de error complementaria

x : distancia (L)

t : tiempo (T)

Esta solución considera un medio saturado y libre de solutos, es decir, la concentración previa de la sustancia a analizar en el medio es nula o despreciable.

2.3. Dispersión

En un medio poroso un fluido viaja a distintas velocidades, esto es debido a que se encuentra en una zona “irregular”, lo que provoca que el fluido se disperse en diferentes direcciones, cambiando su trayectoria y su distribución en el espacio.

Dispersión mecánica: El resultado de las variaciones en las velocidades de flujo es conocido como *dispersión mecánica*.

La dispersión mecánica se produce en dos direcciones, paralela y perpendicular al flujo. La dispersión que ocurre en sentido del flujo se denomina **dispersión longitudinal** y los cambios de las velocidades del flujo en esta trayectoria se deben a la tortuosidad del medio, lo que provoca que las moléculas del fluido se retrasen. O bien, a la separación de los poros, lo que provoca que las moléculas que viajan por caminos más anchos avancen más rápido.

Por otro lado, la dispersión que ocurre en un sentido perpendicular al flujo es llamada **dispersión transversal** y es causada por la bifurcación de las rutas que sigue el fluido.

La dispersión mecánica se calcula mediante el producto del coeficiente de *dispersividad dinámica* o *dispersividad*, el cual indica la capacidad del medio poroso para dispersar un fluido, por la velocidad lineal media:

$$\text{Dispersión mecánica} = (\alpha)(v) \quad (2.10)$$

donde

α : Dispersividad dinámica. Este coeficiente se puede expresar en dirección transversal α_T y longitudinal α_L
 v : velocidad lineal media.

Dispersión hidrodinámica: este proceso actúa diluyendo los contaminantes y disminuyendo su concentración gracias a la acción conjunta de la difusión y la dispersión mecánica. Para expresar estos dos fenómenos de manera conjunta, es necesario introducir el coeficiente de dispersión hidrodinámica D , que se calcula de la siguiente forma:

$$D = \text{dispersión mecánica} + \text{coeficiente de difusión efectiva}$$

Si D se analiza en direcciones separadas, es decir, paralela y perpendicular al flujo entonces:

$$D_L = \alpha_L v + D^* \quad D_T = \alpha_T v + D^*$$

donde

D_L : coeficiente de dispersión hidrodinámica longitudinal
 D_T : coeficiente de dispersión hidrodinámica transversal
 α_L : coeficiente de dispersividad dinámica longitudinal
 α_T : coeficiente de dispersividad dinámica transversal
 v : velocidad lineal media
 D^* : coeficiente de difusión efectiva.

Tomando en cuenta la dispersión hidrodinámica, la solución de la ecuación (2.8) se calcula mediante la siguiente expresión propuesta por Ogata-Banks en 1961:

$$C(x, t) = \frac{C_0}{2} \left[\operatorname{erfc} \left(\frac{x - v \cdot t}{2\sqrt{D_L \cdot t}} \right) + \exp \left(\frac{v \cdot x}{D_L} \right) \operatorname{erfc} \left(\frac{x + v \cdot t}{2\sqrt{D_L \cdot t}} \right) \right] \quad (2.11)$$

donde

$C(x, t)$: Concentración a una distancia x transcurrido un tiempo t

C_0 : concentración inicial (M/L^3)

D_L : coeficiente de dispersión hidrodinámica longitudinal

erfc : función de error complementaria

v : velocidad lineal media x : distancia (L)

t : tiempo (T)

Esta solución es válida para un medio poroso, homogéneo e isotrópico. El flujo de agua subterránea se asume unidimensional en la dirección x y la concentración inicial del soluto en todo el medio es cero, es decir $C_0 = C(x, 0) = 0$. El soluto no se degrada ni reacciona químicamente en el medio.

2.4. Transporte reactivo

En el movimiento de solutos a través de un medio poroso se produce interacción entre solutos y/o entre el soluto y la matriz porosa, es decir, el medio, generando diferentes reacciones químicas durante el proceso de transporte.

Las reacciones químicas ocasionan que las sustancias se muevan más lento a través del medio, se transformen en otras sustancias o queden retenidas en alguna parte del terreno. Los tipos de reacciones en un medio poroso son:

Reacciones ácido-base. Son procesos químicos en los que un ácido dona protones y una base los acepta. Estas reacciones son fundamentales en la química ambiental, especialmente en la regulación del pH en aguas subterráneas y sistemas naturales.[26]

Complejación. Es el proceso en el cual un ion metálico se asocia con una o más moléculas o iones llamados ligandos. Este proceso afecta la movilidad y biodisponibilidad de metales en sistemas acuáticos y suelos. En aguas subterráneas, la complejación con ligandos orgánicos e inorgánicos puede influir en el transporte de contaminantes.[23]

Oxidación-reducción (redox). Las reacciones redox involucran la transferencia de electrones entre especies químicas. La oxidación implica la pérdida de electrones y la reducción la ganancia de estos. En aguas subterráneas, las condiciones redox controlan la especiación y movilidad de muchos elementos, como hierro, manganeso, arsénico y nitrógeno.[4]

Absorción/Adsorción. El proceso de absorción se refiere a la penetración de una sustancia al interior de un sólido, mientras que la adsorción es la adherencia de una sustancia a la superficie del sólido.[28]

Sorción. La sorción implica los dos procesos anteriores juntos.

Biodegradación de compuestos orgánicos. La biodegradación es el proceso mediante el cual los microorganismos descomponen compuestos orgánicos en productos más simples, como dióxido de carbono, metano o biomasa. En aguas subterráneas, este proceso es crucial para la atenuación natural de contaminantes orgánicos, como hidrocarburos y solventes clorados

Estas reacciones se deben tanto a las propiedades de los solutos (solubilidad, coeficiente de partición, y reactividad química) como a las condiciones del medio (pH, temperatura).

Capítulo 3

Modelos de flujo y de transporte en medios porosos saturados.

Después de comprender de manera conceptual los procesos de flujo y transporte, en este capítulo se presenta la formulación de los modelos matemáticos que se usan comúnmente para describir la evolución espacial y temporal de un fluido y/o un soluto en medios porosos, retomando los conceptos mencionados en capítulos anteriores.

3.1. Ecuación de flujo.

Como se mencionó en el capítulo 1, la ley básica en el movimiento de un fluido es la ley de Darcy. Cuando esta ley es expresada dentro de la ecuación de continuidad se obtiene una ecuación en derivadas parciales que es conocida como ecuación de flujo.

En este capítulo se describirá la forma en que se obtiene la ecuación de flujo a partir de la ley de Darcy y la ecuación de continuidad, considerando un medio poroso saturado y un estado transitorio.

Para comenzar, es conveniente pensar en una unidad de volumen del medio poroso (como la que se muestra en la siguiente figura). A esta unidad se le denomina frecuentemente volumen de control elemental (VCE).²

²El VCE es una abstracción matemática empleada en el proceso de creación de modelos matemáticos de procesos físicos.

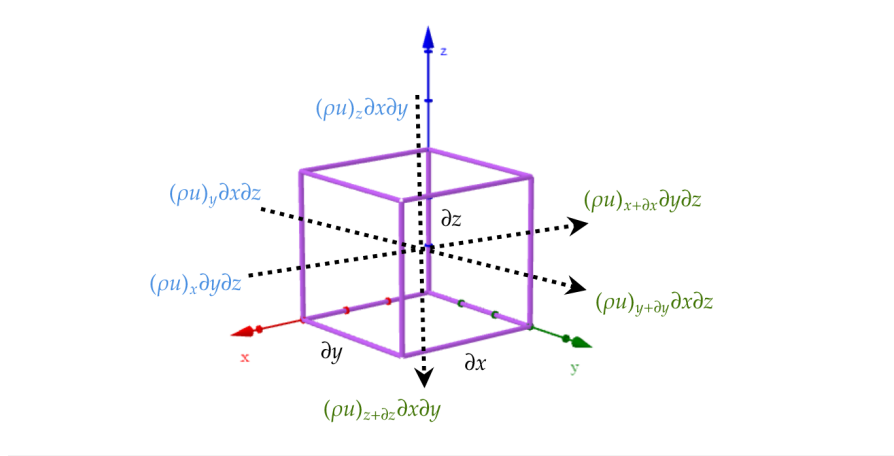


Figura 3.1: Volumen de control elemental

De acuerdo al *Teorema de transporte de Reynolds*, la conservación del flujo másico en un VCE es igual a la suma de los flujos másicos que entran menos la suma de los flujos másicos que salen:

$$\frac{\partial m_{vce}}{\partial t} = \sum_{n=1}^{N_e} \dot{m}_e - \sum_{n=1}^{N_s} \dot{m}_s \quad (3.1)$$

Con base a la figura 3.1, podemos hacer un balance de entrada y salida de flujos para verificar la conservación del flujo másico:

$$\begin{aligned} \frac{\partial m_{vce}}{\partial t} = & (\rho u)_x \partial y \partial z + (\rho u)_y \partial x \partial z + (\rho u)_z \partial x \partial y \\ & - (\rho u)_{x+\partial x} \partial y \partial z - (\rho u)_{y+\partial y} \partial x \partial z - (\rho u)_{z+\partial z} \partial x \partial y \end{aligned} \quad (3.2)$$

aquí,

ρ : es la densidad del fluido

u : es la velocidad del fluido

ρu : es el flujo másico (masa por unidad de área por unidad de tiempo).

Agrupando términos semejantes:

$$\begin{aligned} \frac{\partial m_{vce}}{\partial t} = & (\rho u)_x \partial y \partial z - (\rho u)_{x+\partial x} \partial y \partial z \\ & + (\rho u)_y \partial x \partial z - (\rho u)_{y+\partial y} \partial x \partial z \\ & + (\rho u)_z \partial x \partial y - (\rho u)_{z+\partial z} \partial x \partial y \end{aligned} \quad (3.3)$$

factorizando $\partial y \partial z$, $\partial x \partial z$ y $\partial x \partial y$:

$$\begin{aligned} \frac{\partial m_{vce}}{\partial t} &= (\rho u)_x - (\rho u)_{x+\partial x} \partial y \partial z \\ &+ (\rho u)_y - (\rho u)_{y+\partial y} \partial x \partial z \\ &+ (\rho u)_z - (\rho u)_{z+\partial z} \partial x \partial y. \end{aligned} \quad (3.4)$$

En el capítulo 1 se mencionó que la densidad $\rho = M/V$, por lo que la masa para el VCE puede calcularse como $m_{vce} = (\rho)(V)$ y tomando en cuenta que estamos analizando un medio poroso es importante introducir la porosidad del medio (θ) en el análisis, de modo que podemos escribir el incremento de la masa de agua por unidad de volumen del medio poroso por unidad de tiempo como:

$$\frac{\partial \theta \rho V}{\partial t}. \quad (3.5)$$

Considerando que el volumen de nuestro VCE es $\partial x \partial y \partial z$, dividiendo la ecuación (3.4) entre dicho volumen y sustituyendo el lado izquierdo de la ecuación (3.4) por (3.5) tenemos:

$$\begin{aligned} \frac{1}{\partial x \partial y \partial z} \frac{\partial \theta \rho \partial x \partial y \partial z}{\partial t} &= \frac{(\rho u)_x - (\rho u)_{x+\partial x}}{\partial x} \\ &+ \frac{(\rho u)_y - (\rho u)_{y+\partial y}}{\partial y} \\ &+ \frac{(\rho u)_z - (\rho u)_{z+\partial z}}{\partial z}. \end{aligned} \quad (3.6)$$

Como el volumen de control no cambia respecto al tiempo la ecuación (3.6) se puede reescribir de la siguiente manera:

$$\frac{\partial \theta \rho}{\partial t} = \frac{(\rho u)_x - (\rho u)_{x+\partial x}}{\partial x} + \frac{(\rho u)_y - (\rho u)_{y+\partial y}}{\partial y} + \frac{(\rho u)_z - (\rho u)_{z+\partial z}}{\partial z}. \quad (3.7)$$

Factorizando la ecuación (3.7) tenemos:

$$\frac{\partial \theta \rho}{\partial t} = - \frac{[(\rho u)_{x+\partial x} - (\rho u)_x]}{\partial x} - \frac{[(\rho u)_{y+\partial y} - (\rho u)_y]}{\partial y} - \frac{[(\rho u)_{z+\partial z} - (\rho u)_z]}{\partial z}. \quad (3.8)$$

Usando la definición de derivada llegamos a la última ecuación llamada **ecuación de continuidad**:

$$\frac{\partial(\theta \rho)}{\partial t} = - \frac{\partial(\rho u)}{\partial x} - \frac{\partial(\rho u)}{\partial y} - \frac{\partial(\rho u)}{\partial z}. \quad (3.9)$$

La ecuación de continuidad (3.9) describe la ley de conservación de masa de un fluido durante su circulación a través de un medio poroso.

Si la densidad del fluido es constante, podemos escribir la ecuación (3.9) como:

$$\frac{1}{\rho} \frac{\partial(\theta \rho)}{\partial t} = - \frac{\partial(u)}{\partial x} - \frac{\partial(u)}{\partial y} - \frac{\partial(u)}{\partial z} \quad (3.10)$$

y al introducir los módulos de elasticidad del agua y del medio poroso y expresarlos en términos de la variación de la carga hidráulica (h)[2], el primer término de la ecuación (3.10) se puede escribir como:

$$\frac{1}{\rho} \frac{\partial(\theta\rho)}{\partial t} = S_s \frac{\partial h}{\partial t}. \quad (3.11)$$

Por otro lado, retomando la ecuación (2.3), sustituyendo en la ecuación (3.10) y agregando los términos que expresan fuentes o sumideros presentes en el dominio del flujo obtenemos finalmente:

$$\frac{\partial}{\partial x} \left(K_x \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left(K_y \frac{\partial h}{\partial y} \right) + \frac{\partial}{\partial z} \left(K_z \frac{\partial h}{\partial z} \right) + W = S_s \frac{\partial h}{\partial t} \quad (3.12)$$

donde

h : carga hidráulica,
 K_x, K_y, K_z : conductividad hidráulica del medio,
 W : fuentes y sumideros
 S_s : almacenamiento específico.
 t : tiempo,

La ecuación de flujo en medios porosos (3.12) es válida para medios saturados, no homogéneos e anisotrópicos en estado transitorio.

3.2. Ecuación de transporte advección-dispersión.

La ecuación advección-dispersión describe el movimiento de solutos en medios porosos considerando el transporte debido al flujo del agua (por advección) y la mezcla de solutos (por difusión).

El transporte de solutos se rige, al igual que el flujo, por el principio de conservación de masa dentro de un VCE. En este contexto, el cambio de la masa del soluto respecto al tiempo es igual al flujo total de masa que entra menos el flujo total de masa que sale:

$$\frac{\partial C}{\partial t} + \nabla \cdot \mathbf{J}_T = 0 \quad (3.13)$$

donde

C : concentración del soluto,
 $\nabla \cdot$: divergencia que mide la diferencia entre el flujo que entra y el flujo que sale del volumen de control elemental,
 \mathbf{J}_T : flujo total de masa (por advección y dispersión).

Si consideramos que el flujo total de masa se compone por el flujo advectivo (ecuación (2.4)) y el flujo difusivo (ecuación (2.6)) entonces:

$$\mathbf{J}_T = J_{adv} + J_{dif} = qC - D^* \nabla C \quad (3.14)$$

y sustituyendo (3.14) en (3.13):

$$\frac{\partial C}{\partial t} + \nabla \cdot (qC - D^* \nabla C) = 0. \quad (3.15)$$

Utilizando la propiedad distributiva se tiene que:

$$\frac{\partial C}{\partial t} + \nabla \cdot (qC) - \nabla \cdot (D^* \nabla C) = 0. \quad (3.16)$$

Finalmente, desarrollando la divergencia $\nabla \cdot$ y el gradiente ∇ obtenemos la ecuación de transporte advección-dispersión:

$$\frac{\partial C}{\partial t} + \frac{\partial(q_x C)}{\partial x} + \frac{\partial(q_y C)}{\partial y} + \frac{\partial(q_z C)}{\partial z} = D_m \left(\frac{\partial^2 C}{\partial x^2} + \frac{\partial^2 C}{\partial y^2} + \frac{\partial^2 C}{\partial z^2} \right) \quad (3.17)$$

donde

C : concentración del soluto en estado líquido
 q_x, q_y, q_z : velocidad de Darcy
 D_m : coeficiente de dispersión-difusión
 t : tiempo.

Esta ecuación describe cómo el agua transporta solutos (advección) y cómo los gradientes de concentración generan dispersión o difusión. Nuevamente el medio poroso es considerado no homogéneo, anisotrópico y saturado.

3.3. Ecuación advección-dispersión-reacción.

De manera más completa, la ecuación (3.16) puede modificarse indicando también las reacciones que ocurren dentro del volumen de control elemental. En este sentido el cambio en la concentración de un soluto, respecto al tiempo es igual al flujo total de entrada/salida más el término de las reacciones químicas.

Entonces el modelo matemático que describe la fase de transporte de solutos en un medio poroso, donde se experimentan los procesos de advección, dispersión y reacción, se puede escribir como:

$$\frac{\partial (\theta C^k)}{\partial t} = \frac{\partial}{\partial x_i} \left(\theta D_{ij} \frac{\partial C^k}{\partial x_j} \right) - \frac{\partial}{\partial x_i} (\theta v_i C^k) + q_s C_s^k + \Sigma R_n \quad (3.18)$$

donde

θ : porosidad
 C^k : es la concentración disuelta de la especie k
 t : tiempo
 x_i, x_j : distancias a lo largo del eje x y el eje y
 D_{ij} : Coeficiente de dispersión

v_i : velocidad lineal del flujo
 q_s : caudal (fuentes o sumideros)
 C_s^k : es la concentración en fuentes o sumideros de la especie k
 ΣR_n : término de reacción química.

En la ecuación (3.18), el primer término del lado derecho representa la dispersión hidrodinámica, en la cual está involucrada la dispersión mecánica y la difusión molecular.

El segundo término expresa la advección. El tercero indica las concentraciones (con su flujo) que entran y/o salen del medio. Y el cuarto representa las reacciones químicas.

3.4. Condiciones iniciales y de frontera.

Las ecuaciones diferenciales parciales presentadas en las secciones anteriores modelan fenómenos físicos, por lo que es necesario especificar un conjunto de condiciones iniciales y de frontera. Al definir estas condiciones aseguramos que el resultado obtenido al resolver una EDP exista y sea único.

Las condiciones iniciales, en el caso del estudio de la propagación de un soluto en un acuífero, se utilizan para indicar el estado inicial del medio. Una condición inicial, en general, se escribe de la siguiente forma:

$$\varphi(x, y, z, t) = \varphi_0(x, y, z); \quad t = 0$$

donde φ puede representar concentración, velocidad, temperatura o presión.

Por su parte, las condiciones de frontera se utilizan para determinar la interacción del medio con su entorno. Existen tres tipos de condiciones de frontera que son consideradas al solucionar las ecuaciones diferenciales, las cuales indican la velocidad del flujo, la concentración, temperatura o presión, estableciendo un valor fijo en la frontera del dominio. Estas tres condiciones se presentan a continuación.

Condiciones tipo Dirichlet. En un modelo de flujo, las condiciones tipo Dirichlet indican la carga hidráulica sobre la frontera del dominio, mientras que en un modelo de transporte indica la concentración del soluto. Estas condiciones se expresan, para el flujo como:

$$h(x, y, z) = h_D(x, y, z),$$

y para el transporte:

$$C(x, y, z) = C_D(x, y, z),$$

donde h_D y C_D son funciones conocidas.[6]

Condiciones tipo Neumann. En el caso del flujo en medios porosos, estas condiciones corresponden a una tasa de flujo normal a la frontera. Matemáticamente se expresan como:

$$\frac{\partial h}{\partial n} = q_N(x, y, z, t)$$

Para el transporte de solutos, representa un flujo dispersivo o difusivo:

$$\frac{\partial C}{\partial n} = J_N(x, y, z, t)$$

Donde q_N y J_N representan las tasas de flujo impuestas en la frontera.

Condiciones tipo Cauchy. Estas combinan las condiciones de Dirichlet y Neumann, imponiendo un flujo dependiente de la diferencia de la carga hidráulica o la concentración. Su expresión general para el flujo es:

$$q(x, y, z, t) = \alpha(h(x, y, z) - h_{ext}(x, y, z, t))$$

y para el transporte:

$$J(x, y, z, t) = \beta(C(x, y, z) - C_{ext}(x, y, z, t))$$

donde α , β son llamados coeficientes de intercambio y h_{ext} , C_{ext} representan las condiciones externas.[3]

Es importante elegir de manera adecuada las condiciones iniciales y de frontera para garantizar la estabilidad del modelo. La representación, aplicación e interpretación de los resultados en contextos reales también dependen de la buena definición de estas condiciones.

Capítulo 4

Métodos de solución numérica de los modelos de flujo y transporte.

En la actualidad existen diversos métodos numéricos que se usan para aproximar las soluciones de las ecuaciones diferenciales parciales. Uno de los métodos más comunes es el llamado **diferencias finitas**, desarrollado por Leonhard Euler en 1768 para una dimensión espacial e incrementado a dos dimensiones por C. Runge en 1908. Este método es de fácil implementación y eficiente en términos computacionales.

4.1. Ideas básicas del método de diferencias finitas.

El objetivo del método de diferencias finitas es transformar un problema de cálculo en un problema algebraico [18]. Para resolver una ecuación diferencial parcial, el método discretiza el dominio en una malla y aproxima la derivada de una función mediante la razón de dos diferencias.

Sea $u(x)$ una función derivable dentro del dominio de interés y sea N_x el número de subintervalos en el dominio a lo largo del eje x . Se considerará la siguiente notación:

$$u_i = u(x_i)$$

donde $x_i = i\Delta x$ para $i = 1, 2, 3, \dots, n$ y $\Delta x = \frac{1}{N_x}$.

Las aproximaciones a su primera derivada pueden calcularse mediante:

Diferencias hacia adelante:

$$\left. \frac{du}{dx} \right|_{x_i} \approx \frac{u_{i+1} - u_i}{\Delta x}$$

Diferencias hacia atrás:

$$\left. \frac{du}{dx} \right|_{x_i} \approx \frac{u_i - u_{i-1}}{\Delta x}$$

Diferencias centradas:

$$\left. \frac{du}{dx} \right|_{x_i} \approx \frac{u_{i+1} - u_{i-1}}{2\Delta x}$$

En el caso de la segunda derivada de la función $u(x)$, se calcula como:

$$\left. \frac{d^2u}{dx^2} \right|_{x_i} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}$$

De manera que, si consideramos que $u(x, y)$ o $u(x, y, z)$ y tomando en cuenta la notación explicada anteriormente tenemos para y y z :

$$y_j = j\Delta y \text{ para } j = 1, 2, 3, \dots, n \text{ y } \Delta y = \frac{1}{N_y}$$

y

$$z_k = k\Delta z \text{ para } k = 1, 2, 3, \dots, n \text{ y } \Delta z = \frac{1}{N_z}$$

donde N_y y N_z son los subintervalos correspondientes a las coordenadas y y z , respectivamente.

Diferencias finitas en dos dimensiones

$$\left. \frac{\partial^2 u}{\partial x^2} \right|_{(x_i, y_j)} + \left. \frac{\partial^2 u}{\partial y^2} \right|_{(x_i, y_j)} \approx \frac{u_{i+1, j} - 2u_{i, j} + u_{i-1, j}}{\Delta x^2} + \frac{u_{i, j+1} - 2u_{i, j} + u_{i, j-1}}{\Delta y^2}$$

Diferencias finitas en tres dimensiones

$$\begin{aligned} \left. \frac{\partial^2 u}{\partial x^2} \right|_{(x_i, y_j, z_k)} + \left. \frac{\partial^2 u}{\partial y^2} \right|_{(x_i, y_j, z_k)} + \left. \frac{\partial^2 u}{\partial z^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{u_{i+1, j, k} - 2u_{i, j, k} + u_{i-1, j, k}}{\Delta x^2} \\ &+ \frac{u_{i, j+1, k} - 2u_{i, j, k} + u_{i, j-1, k}}{\Delta y^2} \\ &+ \frac{u_{i, j, k+1} - 2u_{i, j, k} + u_{i, j, k-1}}{\Delta z^2} \end{aligned} \quad (4.1)$$

Cuando aplicamos el método de diferencias finitas a la ecuación de flujo y transporte en medios porosos, a cada nodo del dominio se asocia una ecuación que resulta de la discretización de la ecuación diferencial parcial en el tiempo y el espacio.

Para la ecuación de flujo (3.12), es posible escribir la discretización espacial utilizando diferencias centradas:

$$\begin{aligned}\left. \frac{\partial^2 h}{\partial x^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{h_{i+1, j, k} - 2h_{i, j, k} + h_{i-1, j, k}}{\Delta x^2} \\ \left. \frac{\partial^2 h}{\partial y^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{h_{i, j+1, k} - 2h_{i, j, k} + h_{i, j-1, k}}{\Delta y^2} \\ \left. \frac{\partial^2 h}{\partial z^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{h_{i, j, k+1} - 2h_{i, j, k} + h_{i, j, k-1}}{\Delta z^2}\end{aligned}$$

El término temporal, del lado izquierdo de la ecuación (3.12), se escribe utilizando diferencias hacia adelante y su cambio se representa en el supra índice de cada término de carga hidráulica, recordando que $h(x, y, z, t)$:

$$\left. \frac{\partial h}{\partial t} \right|_{(x_i, y_j, z_k, t_n)} \approx \frac{h_{i, j, k}^{n+1} - h_{i, j, k}^n}{\Delta t}$$

Sustituyendo las aproximaciones en la ecuación de flujo:

$$\begin{aligned}S_s \frac{h_{i, j, k}^{n+1} - h_{i, j, k}^n}{\Delta t} &= K_x \frac{h_{i+1, j, k}^{n+1} - 2h_{i, j, k}^{n+1} + h_{i-1, j, k}^{n+1}}{\Delta x^2} \\ &+ K_y \frac{h_{i, j+1, k}^{n+1} - 2h_{i, j, k}^{n+1} + h_{i, j-1, k}^{n+1}}{\Delta y^2} \\ &+ K_z \frac{h_{i, j, k+1}^{n+1} - 2h_{i, j, k}^{n+1} + h_{i, j, k-1}^{n+1}}{\Delta z^2} + W\end{aligned}\quad (4.2)$$

Agrupando los términos semejantes se tiene que:

$$\begin{aligned}\frac{S_s}{\Delta t} h_{i, j, k}^{n+1} + W &= -\frac{K_x}{\Delta x^2} h_{i-1, j, k}^{n+1} - \frac{K_y}{\Delta y^2} h_{i, j-1, k}^{n+1} - \frac{K_z}{\Delta z^2} h_{i, j, k-1}^{n+1} \\ &+ \left(\frac{S_s}{\Delta t} + \frac{2K_x}{\Delta x^2} + \frac{2K_y}{\Delta y^2} + \frac{2K_z}{\Delta z^2} \right) \\ &- \frac{K_x}{\Delta x^2} h_{i+1, j, k}^{n+1} - \frac{K_y}{\Delta y^2} h_{i, j+1, k}^{n+1} - \frac{K_z}{\Delta z^2} h_{i, j, k+1}^{n+1}\end{aligned}\quad (4.3)$$

En el caso de flujo en una dimensión, como el primer problema que se trabaja en la siguiente sección, para cada nodo i , esta ecuación forma parte de un sistema de ecuaciones lineales que se resuelve para encontrar h^{n+1} en todos los nodos.

Si tenemos un sistema con N nodos, la ecuación discreta se escribe en la forma matricial

$$\begin{bmatrix} A_1 & B_1 & 0 & 0 & \cdots \\ B_2 & A_2 & B_2 & 0 & \cdots \\ 0 & B_3 & A_3 & B_3 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} = \begin{bmatrix} h_1^{n+1} \\ h_2^{n+1} \\ h_3^{n+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \end{bmatrix} \quad (4.4)$$

donde

$A_i = \frac{S_s}{\Delta t} + \frac{2K_x}{\Delta x^2}$ es el término diagonal principal.

$B_i = -\frac{K_x}{\Delta x^2}$ representa los coeficientes fuera de la diagonal.

$b_i = \frac{S_s}{\Delta t} h_i^n + Q_i$ es el lado derecho del sistema.

Por otro lado, la ecuación de transporte se aproxima con diferencias finitas de la siguiente forma:

Para el término de advección, las derivadas se aproximan como

$$\begin{aligned} \left. \frac{\partial C}{\partial x} \right|_{(x_i, y_j, z_k, t_n)} &\approx \frac{C_{i,j,k}^n - C_{i-1,j,k}^n}{\Delta x} \\ \left. \frac{\partial C}{\partial y} \right|_{(x_i, y_j, z_k, t_n)} &\approx \frac{C_{i,j,k}^n - C_{i,j-1,k}^n}{\Delta y} \\ \left. \frac{\partial C}{\partial z} \right|_{(x_i, y_j, z_k, t_n)} &\approx \frac{C_{i,j,k}^n - C_{i,j,k-1}^n}{\Delta z} \end{aligned}$$

Los términos difusivos se expresan con diferencias centradas

$$\begin{aligned} \left. \frac{\partial^2 C}{\partial x^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{C_{i+1,j,k} - 2C_{i,j,k} + C_{i-1,j,k}}{\Delta x^2} \\ \left. \frac{\partial^2 C}{\partial y^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{C_{i,j+1,k} - 2C_{i,j,k} + C_{i,j-1,k}}{\Delta y^2} \\ \left. \frac{\partial^2 C}{\partial z^2} \right|_{(x_i, y_j, z_k)} &\approx \frac{C_{i,j,k+1} - 2C_{i,j,k} + C_{i,j,k-1}}{\Delta z^2} \end{aligned}$$

De modo que, sustituyendo en la ecuación (3.17) e incluso cuando se incluye el

término de reacción:

$$\begin{aligned}
\frac{C_{i,j,k}^n}{\Delta t} - R \approx & -\frac{D_x}{\Delta x^2} C_{i-1,j,k} - \frac{D_y}{\Delta y^2} C_{i,j-1,k} - \frac{D_z}{\Delta z^2} C_{i,j,k-1} \\
& + \left(\frac{1}{\Delta t} + \frac{2D_x}{\Delta x^2} + \frac{2D_y}{\Delta y^2} + \frac{2D_z}{\Delta z^2} \right) C_{i,j,k}^{n+1} \\
& - \frac{D_x}{\Delta x^2} C_{i+1,j,k} - \frac{D_y}{\Delta y^2} C_{i,j+1,k} - \frac{D_z}{\Delta z^2} C_{i,j,k+1}
\end{aligned} \tag{4.5}$$

Al utilizar cualquiera de estas aproximaciones es importante especificar condiciones de frontera adecuadas para asegurar una solución única, también se debe tomar en cuenta un Δx adecuado para evitar errores grandes en la aproximación.

El método de diferencias finitas se utiliza en software como MODFLOW y MT3D-USGS para resolver los modelos de flujo y transporte de solutos en medios porosos. Estos software se describen a continuación.

4.2. Herramientas de cómputo utilizadas para la resolución numérica de algunos ejemplos.

El modelado numérico de flujos subterráneos y la simulación del transporte de solutos son etapas esenciales en el estudio de un acuífero, por lo que durante el desarrollo de este trabajo utilizamos diversos software (MODFLOW 6, MT3D-USGS y FloPy), que permitieron realizar simulaciones numéricas de los procesos de flujo y transporte de solutos en medios porosos.

MODFLOW 6 es la versión más reciente del software desarrollado por el Servicio Geológico de los Estados Unidos por sus siglas en inglés “USGS” (United States Geological Survey), para la simulación numérica del flujo en aguas subterráneas [21].

Por otro lado, la simulación del transporte de solutos es una necesidad frecuente en estudios de contaminación, y para ello, MODFLOW 6 se puede acoplar con modelos de transporte como los que ofrece MT3D-USGS para simular la propagación de contaminantes en aguas subterráneas. [30]

La interacción entre MODFLOW y MT3D-USGS, se realiza gracias a Flopy, una biblioteca de Python, que permite la creación, ejecución y postprocesamiento de las simulaciones. A través de Flopy, es posible desarrollar simulaciones de modelos que se adapten a las necesidades específicas de cada estudio.

A continuación se describen las características y el funcionamiento de los software mencionados anteriormente.

4.2.1. MODFLOW 6

MODFLOW 6 está constituido por un programa principal y una serie de subrutinas independientes llamadas módulos y estos, a su vez, están agrupados en paquetes[11]. Un paquete es la parte del programa que trata un solo aspecto de la simulación.

Debido a la gran variedad de opciones que presenta, gracias a los paquetes que incluye, es uno de los programas más populares para modelar el flujo de agua subterránea.

Los principales módulos de *MODFLOW 6* son:

- Simulation (SIM): define los parámetros generales de la simulación
- Groundwater Flow Model (GWF): representa el flujo de agua subterránea en un medio poroso.
- Groundwater Transport Model (GWT): modela el transporte de solutos.

Dentro de las simulaciones realizadas con GWF o GWT es posible considerar acuíferos libres o confinados de una o más capas, donde el flujo se encuentra en estado estacionario o transitorio. Todas estas características se establecen en los paquetes DIS y STO.

Para definir el dominio del modelo se consideran los siguientes tipos de celdas:

- Celdas activas, que participan en la simulación del flujo subterráneo y pueden recibir o perder agua.
- Celdas inactivas, estas no intervienen en la simulación, generalmente utilizadas para definir los límites del dominio del modelo.

Paralelamente se pueden incluir factores como pozos, drenajes, ríos, recargas o descargas que se encuentren en el dominio del sistema. Esto es gracias a los paquetes WEL y CHD.

Por otra parte, el paquete IMS define los parámetros numéricos y de convergencia del modelo. MODFLOW 6 emplea el método de diferencias finitas para resolver las ecuaciones de flujo de aguas subterráneas. Los modelos matemáticos se consideran en una, dos o tres dimensiones y las condiciones iniciales del sistema se definen en el paquete IC.

Otra de las ventajas de MODFLOW 6, es la manipulación de los resultados del modelo, ya que gracias al paquete OC es posible seleccionar los datos a mostrar, la frecuencia, y en algunos casos, el formato de salida de los resultados.

La flexibilidad de MODFLOW 6 lo convierte en una herramienta clave para la investigación en aguas subterráneas y la gestión de recursos hídricos.

4.2.2. MT3D-USGS

MT3D-USGS es un simulador de transporte de solutos en aguas subterráneas y se considera una versión mejorada del modelo de transporte de solutos MT3DMS desarrollado por el USGS.

Entre las mejoras más destacadas se encuentran: la compatibilidad con MODFLOW 6, lo que permite acoplar los términos de flujo calculados por los paquetes MODFLOW que anteriormente no eran compatibles con MT3DMS y nuevas formulaciones para reacciones químicas, lo que permite incluir la capacidad de simular reacciones entre especies y reacciones en cadena. Los paquetes de MT3D-USGS utilizados para el desarrollo de este trabajo son:

- ADV: Modela el transporte por advección.
- BTN: Contiene los parámetros generales del modelo de transporte, como la discretización temporal, las condiciones iniciales de concentración y las propiedades generales del soluto.
- DSP: Modela la dispersión hidrodinámica, que incluye la dispersión mecánica y la difusión molecular.
- GCG: Es el solucionador numérico principal para resolver la ecuación del transporte de solutos.
- RCT: Modela reacciones químicas del soluto, como degradación, adsorción y reacciones de primer y orden mayor.
- SSM: Define las fuentes y sumideros de soluto en el modelo, como pozos de extracción/inyección, recarga contaminada, drenajes o intercambio con ríos.

Estos paquetes trabajan juntos para simular el transporte reactivo de solutos en medios porosos, permitiendo analizar fenómenos como la propagación de solutos en acuíferos.

4.2.3. Flopy

FloPy es una biblioteca de *Python* que permite crear, modificar, ejecutar y visualizar modelos de MODFLOW 6 y MT3D-USGS realizando simulaciones numéricas de flujo en aguas subterráneas y de transporte de solutos.

FloPy opera organizando los modelos de MODFLOW 6 en una estructura basada en objetos de Python. Su funcionamiento se divide en varias etapas:

1. Creación del modelo.
2. Definición de la Malla y Parámetros del Flujo.
3. Definición de Condiciones de Frontera y Fuentes/Sumideros.

4. Configuración del Solucionador y Ejecución.

5. Visualización y Análisis de Resultados.

Esto permite que dentro del ambiente de FloPy se generen modelos paramétricos fácilmente para posteriormente realizar análisis avanzados y procesamiento de datos con NumPy, Pandas, SciPy y finalmente visualizar resultados con Matplotlib.

Para conocer el funcionamiento y la estructura de dichos software participé en un taller en el Instituto de Geofísica de la UNAM, con el objetivo de construir, ejecutar y analizar resultados de ejemplos de flujo y transporte reactivo de primer orden.

Durante el taller verificamos la capacidad de FloPy para manejar simulaciones avanzadas de flujo y transporte reactivo, facilitando una integración eficiente entre MODFLOW 6 y MT3D-USGS.

Las actividades realizadas en dicho taller fueron fundamentales para el desarrollo de esta tesis. Entre ellas destacan: la revisión teórica de los modelos de flujo y transporte en aguas subterráneas descritos en la documentación de los software mencionados; el análisis de la arquitectura y diseño de MT3D-USGS, así como su relación con la programación orientada a objetos en FloPy; y, por último, el desarrollo de ejemplos prácticos que permitieron una mejor comprensión e integración de MODFLOW 6 y MT3D-USGS en la interfaz de FloPy.

Los ejemplos prácticos trabajados durante el curso se presentan en el siguiente capítulo.

La dinámica del taller contribuyó a que obtuviera conocimientos de diversas áreas para lograr una mejor comprensión del estudio de flujo y transporte en acuíferos y proponer soluciones de acuerdo a las necesidades presentadas en cada problema.

Capítulo 5

Presentación y resolución de ejemplos de flujo y de transporte en medios porosos saturados.

En este capítulo se presentan dos ejemplos: el primero modela únicamente el flujo, mientras que el segundo incorpora el transporte de un soluto. La solución de la ecuación de flujo está estrechamente relacionada con la ecuación de transporte. En el primer ejemplo, se obtiene la solución de la ecuación (3.12), correspondiente a la carga hidráulica. A partir de estos resultados, se determina la velocidad del flujo en el medio, la cual se emplea en el segundo ejemplo para analizar el transporte de un soluto dentro del acuífero.

Por último se presenta la simulación del transporte de nitratos en un medio poroso utilizando el modelo de advección-dispersión con decaimiento de primer orden.

Los ejemplos trabajados en este capítulo, fueron tomados de la documentación oficial de MODFLOW 6 y MT3D-USGS y para cada uno de ellos se infirió y construyó un modelo conceptual y/o figurativo basados en los archivos de entrada y los resultados obtenidos.

Por modelo conceptual nos referimos a la descripción simplificada del sistema que ayuda a comprender sus principales componentes, interacciones y procesos. Se utiliza como un primer paso en la modelación para estructurar el problema y definir los elementos clave antes de desarrollar un modelo matemático o computacional.

5.1. Ejemplo de flujo estacionario en una dimensión espacial.

En esta sección se describe la simulación de un flujo estacionario en un medio poroso utilizando MODFLOW 6. El problema modela la entrada de un fluido al subsuelo con una velocidad constante, se analiza la distribución del flujo en una geometría simple y bajo condiciones bien definidas.

5.1.1. Modelo conceptual.

Se considera un flujo unidimensional en un pilar horizontal con dimensiones de 12 cm de largo, 1 cm de ancho y 1 cm de alto. Este pilar representa un medio poroso homogéneo e isotrópico con una conductividad hidráulica $K = 0.01$ cm/s. El agua que ingresa al dominio tiene una velocidad constante de $q = 0.1$ cm/s.

Dado que el problema es estacionario, se asume que las condiciones de flujo no varían con el tiempo.

El experimento se realiza durante un tiempo de 120 segundos contando con un solo periodo de estrés.

Condiciones iniciales y de frontera.

La carga hidráulica mantiene un nivel constante igual a 1 cm en cualquier instante de tiempo en la celda 120, es decir en $x = 12$ cm, por lo que la condición de frontera se puede expresar de la siguiente manera:

$$h(12, t) = 1\text{cm}$$

Por otro lado, debemos tomar en cuenta el flujo de agua constante que entra en la frontera en el dominio del modelo, de manera que la segunda condición inicial para nuestro modelo se expresa como:

$$q = 0.1\text{cm/s}$$

Con la información proporcionada anteriormente se diseñó el siguiente esquema para facilitar la comprensión del modelo.

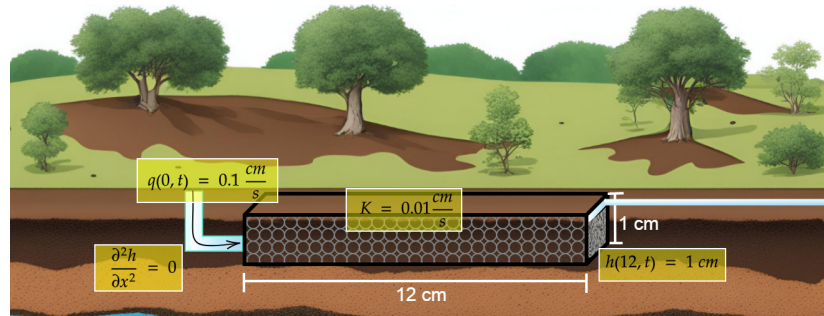


Figura 5.1: Modelo figurativo.

5.1.2. Modelo matemático.

El modelo matemático que describe el flujo en un medio poroso, en una dimensión espacial se puede escribir como:

$$\frac{\partial}{\partial x} \left(K \frac{\partial h}{\partial x} \right) + W = S_s \frac{\partial h}{\partial t} \quad . \quad (5.1)$$

Como se considera K constante y $W = 0$ pues a lo largo de la columna examinada no existe alguna fuente o sumidero, la ecuación (5.1) se reescribe como:

$$K \frac{\partial^2 h}{\partial x^2} = S_s \frac{\partial h}{\partial t} \quad (5.2)$$

despejando el lado izquierdo de la ecuación (5.2) tenemos

$$\frac{\partial^2 h}{\partial x^2} = \frac{S_s}{K} \frac{\partial h}{\partial t} \quad . \quad (5.3)$$

Considerando que en este ejemplo trabajamos en un estado estacionario, es decir, que la carga hidráulica h no cambia con respecto al tiempo, entonces

$$\frac{\partial h}{\partial t} = 0 \quad ,$$

por lo que la ecuación a resolver es

$$\frac{\partial^2 h}{\partial x^2} = 0 \quad . \quad (5.4)$$

Al integrar la ecuación anterior dos veces con respecto a x obtenemos la siguiente solución analítica

$$h(x) = Cx + D \quad C, D, \in \mathbb{R}.$$

Utilizando las condiciones de frontera que se indican en la figura 5.1

$$\begin{aligned} h(12, t) &= 1 \text{ cm} \quad , \\ q(0, t) &= 0.1 \frac{\text{cm}}{\text{s}} \end{aligned}$$

podemos encontrar los valores de C y D de la solución analítica.

A partir de lo anterior, cuando se integra una vez la ecuación (5.4), tenemos

$$\frac{dh}{dx} = C$$

por Darcy ecuación (1.1),

$$q = -K \frac{dh}{dx}$$

entonces

$$C = -\frac{q}{K}$$

sustituyendo los valores de q y K

$$C = -\frac{0.1}{0.01} = -10$$

El valor de D se obtiene sustituyendo C en la primera condición de frontera

$$h(12) = -10(12) + D = 1,$$

despejando

$$D = 121.$$

De esta forma obtenemos la solución analítica

$$h(x) = -10x + 121. \tag{5.5}$$

En esta solución $h(x)$ representa la carga hidráulica en función de la distancia x y como se muestra en la gráfica, el término $-10x$ indica un gradiente hidráulico negativo, lo que sugiere que el flujo disminuye a lo largo del sistema.

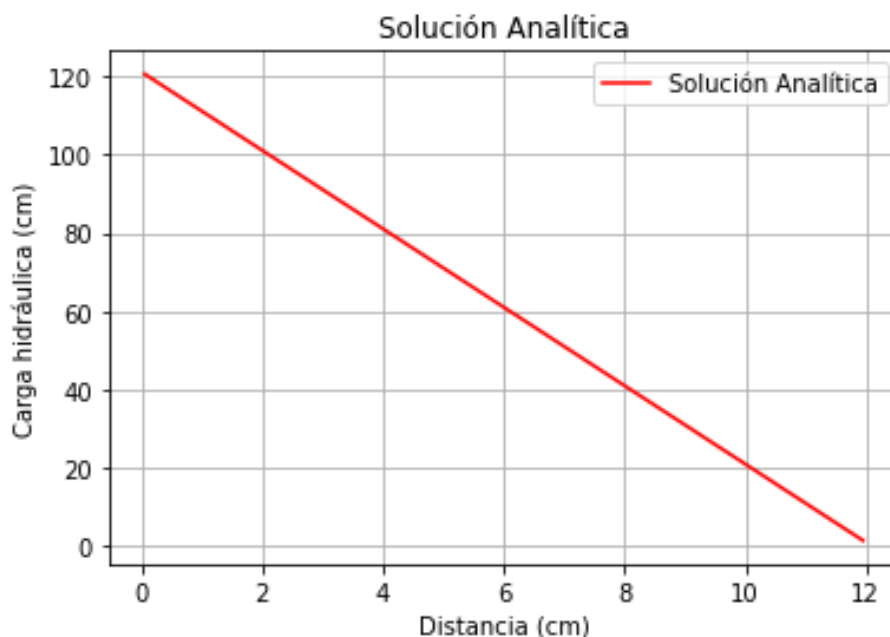


Figura 5.2: Gráfica de la solución analítica

5.1.3. Simulación del flujo con FloPy-MODFLOW 6

Para validar la solución analítica desarrollada anteriormente, se implementó el problema en *MODFLOW 6* utilizando *FloPy*.

En esta sección se describe el código desarrollado en un notebook de Jupyter para la simulación del flujo en medios porosos utilizando *MODFLOW 6* a través de la biblioteca *FloPy*. Se presentan los pasos clave para la construcción del modelo, incluyendo la definición del dominio, las condiciones de frontera, los parámetros hidráulicos y la configuración de los paquetes necesarios para la simulación.

Simulación de flujo.

Antes de comenzar con la simulación del flujo es importante la instalación de *FloPy* dentro Jupyter Notebook. Esta se realiza de manera sencilla utilizando el siguiente comando:

```
[1]: # En caso de no tener instalado Flopy es necesario descomentar
      # la siguiente línea de código
      #!pip install flopy
```

Posteriormente es necesario ejecutar el script `ruta_temporal.py`, que contiene definiciones y configuraciones relacionadas con el proyecto.³

```
[2]: # Temporal, para agregar las rutas correctas
%run C:\\Users\\ADMON\\FyT\\src\\ft1D\\ruta_temporal.py
if not(src_path in sys.path[0]):
    sys.path.insert(0, os.path.abspath(a_path + src_path))
```

Paso 0. Importación de bibliotecas.

En Python, las bibliotecas permiten ampliar las funciones del lenguaje mediante módulos que contienen funciones y clases predefinidas, por lo que un paso importante es la importación de bibliotecas.

```
[3]: import os, sys      # Interfaces con el sistema operativo.
import numpy as np # Manejo de arreglos numéricos
                        multidimensionales
import matplotlib.pyplot as plt # Graficación

# Biblioteca y módulos de flopy
import flopy
from flopy.plot.styles import styles

# Extras para mf6 y flopy (módulo creado en este proyecto)
import xmf6
import sys
#sys.path
```

Paso 1. Definición de parámetros del problema.

Comenzando con las variables requeridas para la simulación, se deben configurar los parámetros que establecen las condiciones físicas y matemáticas del modelo como la discretización espacial, discretización temporal, parámetros físicos y unidades de medida del tiempo y espacio.

La malla utilizada en este modelo para la simulación numérica tiene las siguientes características:

- ◊ Número de columnas: 120 de tamaño $\Delta x = 0.1$ cm
- ◊ Número de filas: 1 de tamaño $\Delta y = 0.1$ cm
- ◊ Número de capas: 1

y dentro de FloPy se define como:

³Este script se encuentra en el anexo número (1).

```
[4]: # Diccionario para ingresar los datos de la malla del sistema
mesh = xmf6.MeshDis(
    nrow = 1,      # Número de filas
    ncol = 120,   # Número de columnas
    nlay = 1,     # Número de capas
    row_length = 12.0, # Longitud del sistema (cm)
    column_length = 0.1, # Length of system (cm)
    top = 1.0,    # Top of the model (cm)
    bottom = 0,   # Layer bottom elevation (cm)
)

xmf6.nice_print(mesh.get_dict(), 'Discretización espacial')
```

Para la discretización temporal se considera un periodo de estrés con solo un paso de tiempo y la duración de la simulación es de 12 segundos

```
[5]: # En este diccionario se ingresan los datos del tiempo para el
      # flujo
tm_par = dict(
    nper = 1,      # Número de periodos
    total_time = 120.0, # Tiempo de simulación ($s$)
    nstp = 1.0,   # Número de pasos de tiempo
    tsmult = 1.0 # Multiplier for the length of
    successive time steps.
)

xmf6.nice_print(tm_par, 'Discretización temporal para el
flujo')
```

Las unidades de medida para esta simulación son segundos y centímetros para el tiempo y el espacio, respectivamente

```
[6]: ## Unidades del modelo
ml_units = {}
ml_units["time"] = "seconds"
ml_units["length"] = "centimeters"
xmf6.nice_print(ml_units, 'Unidades')

## Parámetros físicos
ph_par = {}
ph_par["specific_discharge"] = 0.1 # descarga específica (cm/
↪s)
ph_par["hydraulic_conductivity"] = 0.01 # conductividad
hidráulica (cm/s)
ph_par["source_concentration"] = 1.0 # Source concentration
(unitless)
```

```
xmf6.nice_print(ph_par, 'Parámetros físicos')
```

Paso 2. Ambiente de MODFLOW6 y salida de datos.

En este paso, se define la configuración del entorno de simulación con MODFLOW 6 y de los nombres de los archivos de salida. Aquí se adelanta la configuración para la simulación de transporte que se llevará a cabo en la siguiente sección.

```
[7]: os_par = dict(
    ws = os.getcwd() + '/Salidas', # Ruta de donde estamos
    actualmente
    mf6_exe = 'C:\\Users\\ADMON\\Desktop\\mf6.exe', # ruta al
    archivo ejecutable de MODFLOW 6
    flow_name = 'flow', # Nombre de la simulación para el flujo
    tran_name = 'transport' # Nombre de la simulación para
    transporte
)
xmf6.nice_print(os_par, 'Ambiente MODFLOW 6')

oc_par = dict(
    head_file = f"{os_par['flow_name']}.hds", # archivo donde se
    almacenarán los resultados de las alturas piezométricas (carga
    hidráulica).
    fbudget_file = f"{os_par['flow_name']}.bud", # este archivo
    contiene información sobre las entradas y salidas de agua en el
    sistema.
    concentration_file=f"{os_par['tran_name']}.ucn", # archivo
    donde se almacenarán las concentraciones de un contaminante o
    sustancia
    tbudget_file = f"{os_par['tran_name']}.bud",
)
xmf6.nice_print(oc_par, 'Archivos de salida')
```

Paso 3. Inicio de la simulación.

En este bloque se utiliza el módulo `MFSimulation` el cual carga, crea y/o guarda una simulación de MODFLOW 6. Para nuestro caso estamos cargando un modelo existente, `mf6`.

```
[8]: # flopy.mf6.MFSimulation es una clase en FloPy que se utiliza
    para crear un objeto de simulación en MODFLOW 6
    sim = flopy.mf6.MFSimulation(
        sim_name = os_par['flow_name'], # nombre de la simulación
```

```

sim_ws = os_par['ws'], # directorio de trabajo donde se
guardarán los archivos de entrada y salida de la simulación.
exe_name = os_par['mf6_exe'] # Especifica la ruta al archivo
ejecutable de MODFLOW 6.
)
#print(sim)

```

Paso 4. Definición de la discretización temporal.

El módulo ModflowTdis nos ayuda a definir el paquete de la discretización temporal de la simulación TDIS, utilizando las variables definidas anteriormente en la configuración de la discretización temporal.

```

[9]: flopy.mf6.ModflowTdis(
    sim,
    nper = tm_par["nper"], # Número de periodos de simulación
    perioddata = ((tm_par["total_time"], # Longitud del
periodo.
                    tm_par["nstp"], # Número de pasos de tiempo
en un periodo de estrés
                    tm_par["tsmult"]),), # Multiplicador para los
pasos de tiempo
    time_units = ml_units["time"] # Unidad de tiempo
)

```

Paso 5. Definición de la solución numérica.

El módulo ModflowIms define el paquete IMS (Iterative Model Solution) el cual es el solucionador no lineal en MODFLOW 6.

```

[10]: ims = flopy.mf6.ModflowIms(sim)
#print(ims)

```

Paso 6. Modelo de Flujo

El módulo ModflowGwf utiliza el paquete GWF en el que se define el modelo utilizado. Este paquete define aspectos como la discretización espacial, propiedades hidráulicas y condiciones iniciales.

```

[11]: gwf = flopy.mf6.ModflowGwf(
    sim,
    modelname = os_par['flow_name'], # Nombre del modelo
    save_flows = True # Indica que todos los términos de flujo
del paquete se escribirán en un archivo específico
)
#print(gwf)

```

Paso 7. Paquete: discretización espacial.

El paquete ModflowGwfdis define la malla del modelo.

```
[12]: dis = flopy.mf6.ModflowGwfdis(  
    gwf, # Modelo utilizado en la simulación  
    length_units = ml_units["length"], # Unidades de longitud  
    nlay = mesh.nlay, # Número de capas  
    nrow = mesh.nrow, # Número de filas  
    ncol = mesh.ncol, # Número de columnas  
    delr = mesh.delr, # es el espacio entre columnas en la  
    dirección de las filas  
    delc = mesh.delc, # es el espacio entre filas en la  
    dirección de la columna  
    top = mesh.top, # es la elevación superior de cada celda  
    botm = mesh.bottom, # es la elevación inferior de cada  
    celda  
    )  
    #print(dis)
```

Paso 8. Paquete: condiciones iniciales.

Este paquete define las alturas piezométricas iniciales.

```
[13]: ic = flopy.mf6.ModflowGwfic(  
    gwf,  
    strt=1.0 # Altura inicial  
    )  
    #print(ic)
```

Paso 9. Paquete: propiedades hidráulicas.

Define la conductividad hidráulica y otras propiedades de flujo.

```
[14]: npf = flopy.mf6.ModflowGwfnpf(  
    gwf,  
    save_specific_discharge = True, # indica que las componentes  
    x, y, z de la descarga específica se calcularán en los centros  
    de las celdas  
    save_saturation = True, # indica que la saturación de la  
    celda se escribirá en un archivo específico llamado BUDGET SAVE  
    FILE  
    icelltype = 0, # flujo confinado  
    k = ph_par['hydraulic_conductivity'], # conductividad  
    hidráulica  
    )
```

```
#print(npf)
```

Paso 10. Paquete: Constant Head, CHD.

Se utiliza para definir celdas con cargas hidráulicas constantes.

```
[15]: chd = flopy.mf6.ModflowGwfchd(  
      gwf,  
      stress_period_data=[[0, 0, mesh.ncol - 1), 1.0]] # Capa 1,  
      fila 1, columna 119: altura hidráulica constante = 1.0  
      )  
      # print(chd)
```

Paso 11. Paquete: Pozos (ModflowGwfwel).

Este paquete se utiliza para modelar pozos que extraen o inyectan agua en el sistema de flujo subterráneo.

```
[16]: q = ph_par["specific_discharge"] * mesh.delc * mesh.delr **  
      mesh.top Aquí es Q  
      aux = ph_par["source_concentration"]  
  
      wel = flopy.mf6.ModflowGwfwel(  
      gwf,  
      stress_period_data = [[(0, 0, 0), q, aux,]], # Capa 1, fila  
      1, columna 1; tasa de inyección  
      pname = "WEL-1", # Nombre del paquete  
      auxiliary = ["CONCENTRATION"],  
      )  
  
      print(wel)
```

Paso 12. Paquete: salida (ModflowGwfoc)

Este paquete controla la salida de datos del modelo, permitiendo guardar valores de carga hidráulica en archivos de salida específicos.

```
[17]: oc = flopy.mf6.ModflowGwfoc(  
      gwf,  
      head_filerecord = oc_par['head_file'], # Archivo de  
      cabezas hidráulicas  
      budget_filerecord = oc_par['fbudget_file'],  
      saverecord = [("HEAD", "ALL"), ("BUDGET", "ALL")], # Guarda  
      cabezas y presupuesto  
      )  
      #print(oc)
```

Paso 13. Escritura de los archivos de entrada para MODFLOW 6

En este paso se generan los archivos de entrada requeridos para ejecutar una simulación en MODFLOW 6. Su función principal es convertir la configuración definida de los paquetes utilizados en Python a archivos compatibles con MODFLOW 6, permitiendo ejecutar el modelo.

Al ejecutar esta instrucción, FloPy crea los archivos correspondientes en la carpeta de trabajo definida en la simulación, garantizando que la estructura del modelo esté correctamente escrita antes de su ejecución.

```
[18]: sim.write_simulation() # Escribir archivos de simulación
```

Paso 14. Ejecución de la simulación.

Esta función llama al ejecutable de MODFLOW 6 y procesa el modelo numérico de flujo subterráneo.

```
[19]: sim.run_simulation() # Correr la simulación
```

Paso 15. Postprocessing.

Cuando la simulación se ejecuta correctamente, MODFLOW 6 generará los archivos de salida correspondientes, que pueden incluir datos de carga hidráulica, balance de agua y otros resultados de interés.

```
[20]: # Obtenemos los resultados de la carga hidráulica
head = flopy.utils.HeadFile(
    os.path.join(os_par['ws'],
                 oc_par['head_file'])) .get_data()

# Obtenemos los resultados del BUDGET
bud = flopy.utils.CellBudgetFile(
    os.path.join(os_par['ws'],
                 oc_par['fbudget_file']),
    precision='double'
)

# Obtenemos las velocidades
spdis = bud.get_data(text='DATA-SPDIS')[0]
qx, qy, qz = flopy.utils.postprocessing.
    ↪get_specific_discharge(spdis, gwf)
```

```
[21]: # Verificamos el tipo y dimensiones de los arreglos donde
# están almacenados la carga hidráulica, el BUDGET, y la
# velocidad.
print('Head : ', type(head), head.shape)
```

```
print('Budget : ', type(bud), bud.shape)
print('spdis : ', type(spdis), spdis.shape)
print('qx : ', type(qx), qx.shape)
print('qy : ', type(qy), qy.shape)
print('qz : ', type(qz), qz.shape)
```

El siguiente comando imprime las dimensiones y el contenido de la variable `head`

```
[22]: print(head.shape, '\n', head)
```

```
(1, 1, 120)
[[[12.9 12.8 12.7 12.6 12.5 12.4 12.3 12.2 12.1 12.  11.9 11.8
11.7 11.6
   11.5 11.4 11.3 11.2 11.1 11.  10.9 10.8 10.7 10.6 10.5 10.4
10.3 10.2
   10.1 10.  9.9  9.8  9.7  9.6  9.5  9.4  9.3  9.2  9.1  9.
8.9  8.8
   8.7  8.6  8.5  8.4  8.3  8.2  8.1  8.  7.9  7.8  7.7  7.6
7.5  7.4
   7.3  7.2  7.1  7.  6.9  6.8  6.7  6.6  6.5  6.4  6.3  6.2
6.1  6.
   5.9  5.8  5.7  5.6  5.5  5.4  5.3  5.2  5.1  5.  4.9  4.8
4.7  4.6
   4.5  4.4  4.3  4.2  4.1  4.  3.9  3.8  3.7  3.6  3.5  3.4
3.3  3.2
   3.1  3.  2.9  2.8  2.7  2.6  2.5  2.4  2.3  2.2  2.1  2.
1.9  1.8
   1.7  1.6  1.5  1.4  1.3  1.2  1.1  1. ]]]
```

El primer resultado indica que la variable `head` es un arreglo de tamaño (nlay, nrow, ncol). Lo que es congruente con la configuración que se indicó en la discretización espacial.

El segundo resultado devuelve los valores de la carga hidráulica en cada celda del modelo. Estos valores representan la altura piezométrica en el medio poroso simulado.

El comando posterior imprime información sobre el flujo en la dirección x almacenada en la variable `qx`

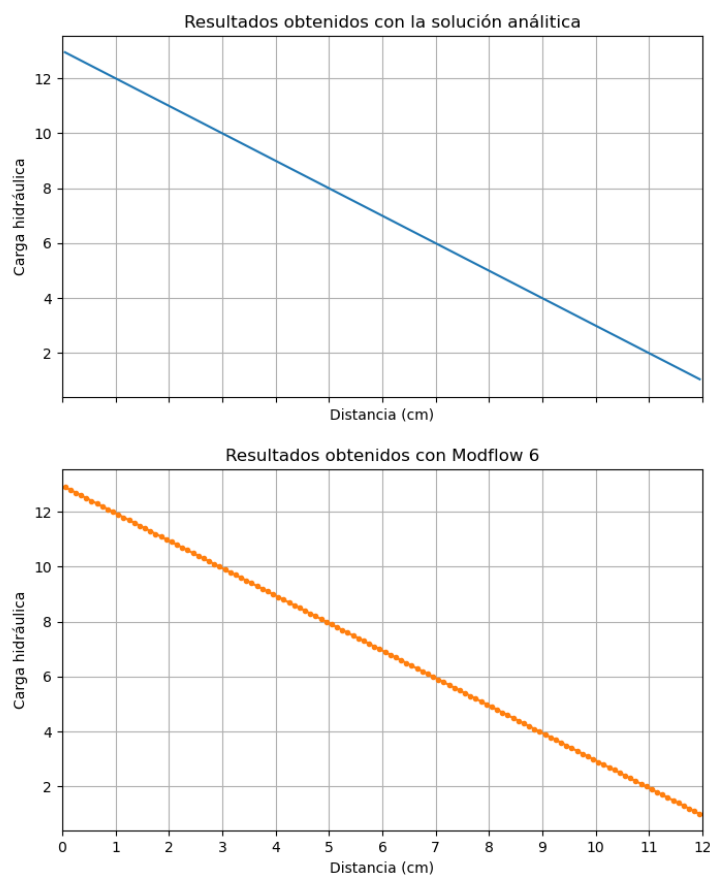
```
[23]: print(qx.shape, '\n', qx)
```

```
(1, 1, 120)
[[[0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01
0.01 0.01
```


Esta gráfica permiten visualizar la distribución de la carga hidráulica, en el dominio considerado, obtenida con MODFLOW 6 - FloPy comparándola con la solución analítica calculada en la sección anterior.

Es importante resaltar que durante el desarrollo de este experimento se encontró una discrepancia en los resultados obtenidos con *MODFLOW 6* y la solución analítica. Esto se debe a que *MODFLOW 6* considera en el paquete *WEL* la entrada de un flujo volumétrico, por lo que el valor de $q = 0.1$ cambia a $q = 0.001$.

Al corregir esta entrada en FloPy, los resultados numéricos coincidieron con la solución analítica, validando así la implementación del modelo.



5.2. Ejemplo de flujo y transporte en una dimensión espacial.

En esta sección se presenta la simulación del transporte de un soluto en un medio poroso saturado utilizando MODFLOW 6 y el paquete GWT (Groundwater Transport). El ejemplo seleccionado se basa en el caso clásico de MOC3D, tomado de la documentación oficial de MODFLOW 6 y FloPy. Este modelo permite analizar el transporte de solutos en un medio poroso, considerando los procesos de advección-dispersión de un soluto en un acuífero homogéneo bajo flujo estacionario.

5.2.1. Modelo conceptual

En esta sección se amplía el análisis del flujo subterráneo para incluir el transporte de un soluto con un único componente químico disuelto. La simulación se lleva a cabo en una columna horizontal de 12 cm de longitud, manteniendo las mismas condiciones hidráulicas previamente establecidas.

El tiempo total de la simulación es de 120 segundos y la concentración inicial del soluto es $C(x, 0) = 0$ en todo el dominio. A lo largo de la simulación, se evaluará la evolución de la concentración en función del tiempo y la distancia, considerando la influencia del flujo subterráneo en el transporte del soluto.

Condiciones iniciales y de frontera.

Condición inicial $C(x, 0) = 0$

Condición de entrada: Una fuente de concentración constante en el extremo izquierdo de la columna.

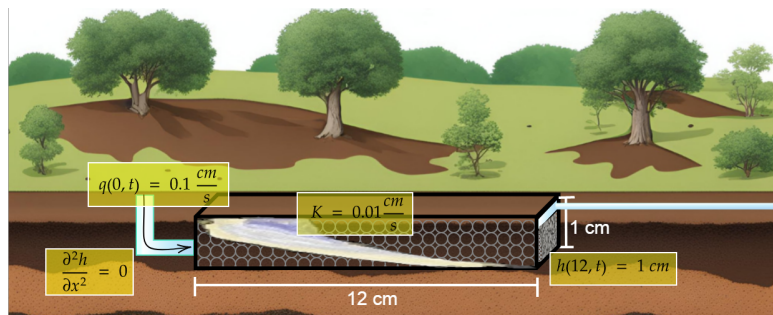


Figura 5.4: Modelo figurativo.

5.2.2. Modelo matemático

El transporte de solutos en medios porosos está gobernado por la ecuación de advección-dispersión en una dimensión:

$$\frac{\partial C}{\partial t} + q_x \frac{\partial C}{\partial x} = D \frac{\partial^2 C}{\partial x^2} \quad (5.6)$$

Con esto se asume que el transporte es controlado por advección y dispersión, sin considerar reacciones químicas.

La solución analítica propuesta para resolver la ecuación advección-dispersión está dada por [29]:

$$C(x, t) = C_0 \left\{ \frac{\exp\left(\frac{(q_x - U)x}{2D}\right) + \frac{U - q_x}{U + q_x} \exp\left(\left(\frac{q_x + U}{2D}\right)x - \frac{UL}{D}\right)}{1 + \frac{U - q_x}{U + q_x} \exp\left(-\frac{UL}{D}\right)} \right\} - 2 \exp\left[\frac{q_x x}{2D} - \lambda t - \frac{q_x^2 t}{4D}\right] \sum_{i=1}^{\infty} \frac{\beta_i \sin \frac{\beta_i x}{L} [\beta_i^2 + [\frac{q_x L}{2D}]^2] \exp[-\frac{\beta_i^2 D t}{L^2}]}{[\beta_i^2 + [\frac{q_x L}{2D}]^2 + \frac{q_x L}{2D}][\beta_i^2 + [\frac{q_x L}{2D}]^2 + \frac{\lambda L}{D}]} \quad (5.7)$$

donde

$$U = \sqrt{q_x^2 + 4\lambda D}$$

β_i son las raíces de la ecuación $\beta \cot \beta + \frac{q_x L}{2D} = 0$.

5.2.3. Implementación en MODFLOW 6-FloPy

A partir del análisis previo del flujo subterráneo (realizado en la sección 5.1.3), se incorpora la simulación del transporte de un soluto en una columna horizontal de 12 cm de longitud, manteniendo las mismas condiciones hidráulicas establecidas. La incorporación de los paquetes necesarios para resolver la ecuación de transporte, permitiendo el análisis de la evolución de la concentración del soluto en el medio se detalla en las siguientes líneas de código.

Continuando con la configuración del modelo de transporte se establece nuevamente las unidades de tiempo y distancia, así como también los parámetros físicos necesarios.

```
[27]: ml_units = {
    "time": "seconds",
    "length": "centimeters"
}
xmf6.nice_print(ml_units, 'Units')

ph_par = dict(
```

```

    specific_discharge = 0.1, # descarga específica ( $\text{cm s}^{-1}$ )
    hydraulic_conductivity = 0.01, # conductividad hidráulica
    ( $\text{cm s}^{-1}$ )
    source_concentration = 1.0, # Source concentration
    porosity = 0.1, # porosidad
    initial_concentration = 0.0, # concentración inicial
    longitudinal_dispersivity = 0.1,
    retardation_factor = 1.0,
    decay_rate = 0.0
)
ph_par["dispersion_coefficient"] =
ph_par["longitudinal_dispersivity"] *
ph_par["specific_discharge"] / ph_par["retardation_factor"]

xmf6.nice_print(ph_par, 'Parámetros físicos')

```

En el siguiente diccionario se realiza un cambio en la discretización temporal ya que, a diferencia del modelo de flujo, durante esta simulación se realizan 240 pasos de tiempo en total.

```

[28]: # En este diccionario se ingresan los datos del tiempo para la
      # simulación del transporte

tm_par = dict(
    nper = 1,          # Número de periodos
    total_time = 120.0, # Tiempo de simulación ($s$)
    nstp = 240,       # Número de pasos de tiempo
    tsmult = 1.0      # Multiplier for the length of
    successive time steps.
)
xmf6.nice_print(tm_par, 'Discretización temporal para el
transporte')

```

Como se mencionó al inicio del capítulo, los resultados obtenidos en la ecuación de flujo están estrechamente ligados a la solución de la ecuación de transporte, por lo que es necesario obtener los datos que ya se generaron durante la ejecución de la simulación anterior.

```

[29]: # Obtenemos los resultados de la carga hidráulica
head = flopy.utils.HeadFile(
    os.path.join(os_par['ws'],
                 oc_par['head_file'])).get_data()
# Obtenemos los resultados del BUDGET
bud = flopy.utils.CellBudgetFile(

```

```

    os.path.join(os_par['ws'],
                 oc_par['fbudget_file']),
    precision='double'
)
# Obtenemos las velocidades
spdis = bud.get_data(text='DATA-SPDIS')[0]
qx, qy, qz = flopy.utils.postprocessing.
    ↪get_specific_discharge(spdis, gwf)

```

Para el caso del transporte es necesario utilizar un nuevo objeto de simulación que, como ya se explicó en el paso 3 de la sección anterior se crea con el módulo MFSimulation.

```

[30]: sim_t = flopy.mf6.MFSimulation(
        sim_name=os_par["tran_name"],
        sim_ws=os_par["ws"],
        exe_name=os_par["mf6_exe"]
    )
    print(sim_t)

```

Se configura nuevamente la discretización temporal para la simulación del transporte a la que llamamos `sim_t`

```

[31]: flopy.mf6.ModflowTdis(
        sim_t,
        nper=tm_par["nper"],
        perioddata=((tm_par["total_time"],
                     tm_par["nstp"],
                     tm_par["tsmult"])),
        time_units=ml_units["time"]
    )

```

El paquete del solucionador iterativo mutigrad (IMS) controla la estrategia numérica utilizada para resolver las ecuaciones de flujo y transporte, permitiendo configurar tolerancias, métodos iterativos y parámetros de convergencia.

```

[32]: ims = flopy.mf6.ModflowIms(
        sim_t,
        linear_acceleration="bicgstab" # Método del gradiente
        biconjugado
    )
    print(ims)

```

La instrucción `flopy.mf6.ModflowGwt` se utiliza para definir el modelo de transporte de solutos en aguas subterráneas (GWT Groundwater Transport), dentro de una simulación en MODFLOW 6.

```
[33]: gwt = flopy.mf6.ModflowGwt(
        sim_t,
        modelname=os_par["tran_name"],
        save_flows=True
    )
    print(gwt)
```

Al igual que en la simulación de flujo, se configura la discretización temporal del modelo.

```
[34]: dis = flopy.mf6.ModflowGwtDis(
        gwt,
        length_units=ml_units["length"],
        nlay=mesh.nlay,
        nrow=mesh.nrow,
        ncol=mesh.ncol,
        delr=mesh.delr,
        delc=mesh.delc,
        top=mesh.top,
        botm=mesh.bottom,
    )
    print(dis)
```

Se realiza la configuración de las condiciones iniciales para el modelo GWT

```
[35]: ic = flopy.mf6.ModflowGwtic(
        gwt,
        strt=0 # Concentración inicial
    )
    print(ic)
```

El paquete de almacenamiento y transferencia de masa (Mass Storage and Transfer, MST) es fundamental en la simulación de transporte para definir cómo se almacena la masa del soluto en el sistema y cómo se transfiere entre las fases acuosa y sólida en el medio poroso.

```
[36]: mst = flopy.mf6.ModflowGwtmst(
        gwt,
        porosity=ph_par["porosity"],
        **get_sorption_dict(ph_par["retardation_factor"]),
        **get_decay_dict(ph_par["decay_rate"],
                        ph_par["retardation_factor"] > 1.0),
    )
    print(mst)
```

El paquete de advección para los modelos de transporte de solutos en MODFLOW 6 es esencial para definir cómo se transporta el soluto en el medio poroso debido al movimiento del agua subterránea.

Este paquete es muy importante en cualquier modelo de transporte de solutos, ya que la advección es uno de los principales mecanismos que gobiernan la movilidad de contaminantes y nutrientes en aguas subterráneas.

```
[37]: adv = flopy.mf6.ModflowGwtadv(  
      gwt,  
      scheme="TVD" # esquema utilizado para resolver el término de  
      advección  
      )  
      print(adv)
```

El paquete de dispersión en MODFLOW 6 define los procesos de dispersión hidrodinámica, que incluyen la dispersión mecánica y la difusión molecular.

```
[38]: dsp = flopy.mf6.ModflowGwtdsp(  
      gwt, #  
      xt3d_off=True, # desactiva el método xt3d  
      alh=ph_par["longitudinal_dispersivity"], # dispersividad  
      longitudinal en dirección horizontal  
      ath1=ph_par["longitudinal_dispersivity"], # Dispersión  
      transversal en dirección horizontal.  
      )  
      print(dsp)
```

El siguiente fragmento de código se encarga de interconectar el modelo de flujo (GWF) con el modelo de transporte (GWT) en MODFLOW 6, utilizando el paquete de intercambio de modelos de flujo (FMI) que permite que el modelo de transporte (GWT) use los resultados de un modelo de flujo (GWF) preexistente en la simulación.

```
[39]: pd = [  
      ("GWFHEAD", oc_par["head_file"], None),  
      ("GWFBDGET", oc_par["fbudget_file"], None),  
      ]  
  
      fmi = flopy.mf6.ModflowGwtfmi(  
      gwt,  
      packagedata=pd  
      )  
      print(fmi)
```

A continuación se define una fuente de soluto en el modelo de transporte (GWT) que se configura el paquete de fuentes y sumideros de masa (SSM). Todo esto nos indica que el pozo (WEL-1) definido en el modelo de flujo tendrá una concentración de soluto asociada.

```
[40]: sourcerearray = [{"WEL-1", "AUX", "CONCENTRATION"}]
      ssm = flopy.mf6.ModflowGwtssm(
          gwt,
          sources=sourcerearray
      )
      print(ssm)
```

Comenzando con el análisis de los resultados, se configura un paquete de observaciones (ModflowUtlobs) en MODFLOW 6 para monitorear la concentración del soluto en diferentes ubicaciones de la malla del modelo de transporte (GWT).

```
[41]: obs_data = {
      "transporte.obs.csv": [
          ("X005", "CONCENTRATION", (0, 0, 0)),
          ("X405", "CONCENTRATION", (0, 0, 40)),
          ("X1105", "CONCENTRATION", (0, 0, 110)),
      ],
  }

  obs_package = flopy.mf6.ModflowUtlobs(
      gwt,
      digits=10,
      print_input=True,
      continuous=obs_data
  )

  print(obs_package)
```

Posteriormente se va a definir la salida del modelo de transporte de soluto (GWT)

```
[42]: oc = flopy.mf6.ModflowGwtoc(
      gwt,
      budget_filerecord=oc_par['tbudget_file'],
      concentration_filerecord=oc_par["concentration_file"],
      saverecord=[("CONCENTRATION", "ALL"), ("BUDGET", "LAST")],
      printrecord=[("CONCENTRATION", "LAST"), ("BUDGET", "LAST")],
  )

  print(oc)
```

Nuevamente se generan los archivos de entrada requeridos para ejecutar la simulación en MODFLOW 6, pero en este caso para el transporte.

```
[43]: sim_t.write_simulation()
```

Se ejecuta la simulación

```
[44]: sim_t.run_simulation()
```

Después de ejecutar la simulación es necesario acceder a los datos de salida del modelo de transporte.

mf6gwt_ra contiene los datos de observación del modelo de transporte, en este caso, las concentraciones en ubicaciones específicas del modelo en el obs_data.

```
[45]: mf6gwt_ra = sim_t.get_model("transport").obs.output.obs().data
ucnobj_mf6 = sim_t.transport.output.concentration() # Datos de
concentración de la simulación
```

Las siguientes líneas de código extraen los tiempos de simulación

```
[49]: simtimes = mf6gwt_ra["totim"]
simitimes.shape
#print(type(simitimes), simtimes)
```

Finalmente, definimos una lista de cadenas que contienen identificadores de ubicaciones de observación.

```
[50]: obsnames = ["X005", "X405", "X1105"]
```

Resultados obtenidos.

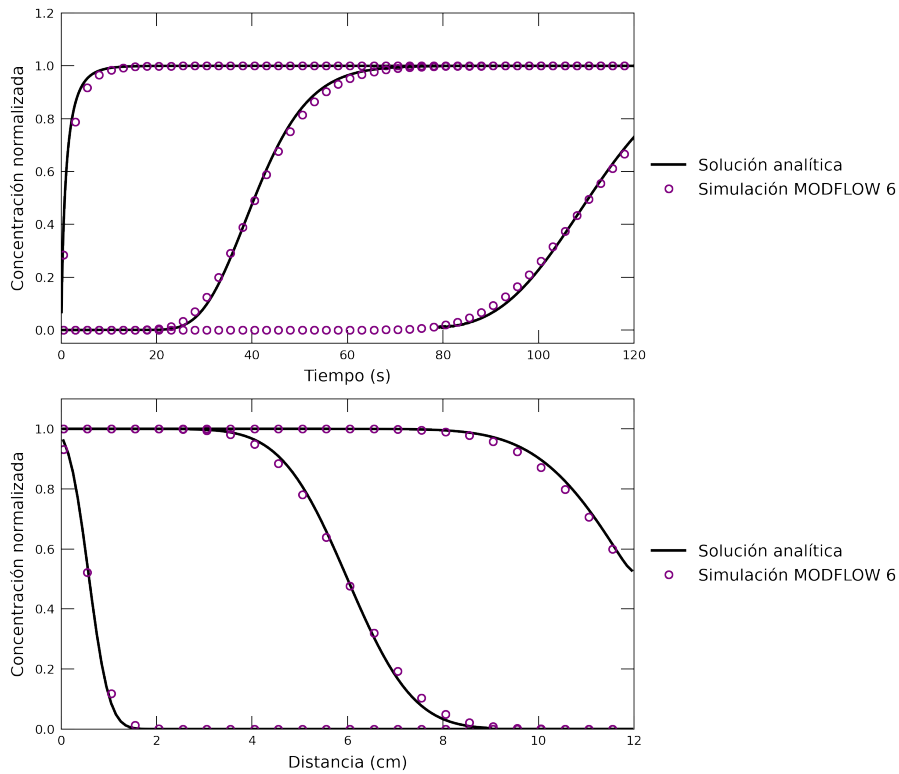


Figura 5.5: Resultados obtenidos de la simulación de transporte.

La gráfica ubicada en la parte superior representa la evolución temporal de la concentración del soluto en tres ubicaciones distintas de la columna.

Por su parte, la gráfica inferior indica la evolución espacial de la concentración en el último instante de tiempo.

En ambas gráficas podemos observar que la solución numérica y la solución analítica muestran el mismo comportamiento por lo que se puede concluir que la ejecución del modelo GWT está mostrando los resultados esperados.

5.3. Ejemplo de transporte advección-dispersión con decaimiento de primer orden.

En el desarrollo de esta sección se utiliza el modelo de advección-dispersión con decaimiento de primer orden para predecir el comportamiento de los nitratos en acuíferos. En este caso, se estudia cómo su concentración varía en el acuífero debido al flujo subterráneo y a procesos de transformación química.

El ciclo del nitrógeno es fundamental para comprender la dinámica de los nitratos en aguas subterráneas, ya que regula las transformaciones químicas que afectan su movilidad y persistencia en los acuíferos. En la siguiente sección explicamos este proceso.

5.3.1. Elementos básicos del ciclo del nitrógeno.

La atmósfera es la fuente inicial y más abundante de nitrógeno, en ella el nitrógeno se encuentra en una proporción del 78 % y en forma de gas. Sin embargo, en esta forma no puede ser captado por las plantas o los vegetales por lo que es aquí donde se inicia un proceso llamado fijación.

En este proceso las llamadas bacterias fijadoras, las cuales se encuentran en interacción con plantas de la familia de las leguminosas, descomponen en el suelo los enlaces del N_2 y lo combinan con átomos de hidrógeno para formar amoníaco (NH_3). Además, existen otras bacterias en el suelo y el agua que convierten este amoníaco en nitrato (NO_3^-). Las descargas eléctricas que se producen en la atmósfera también enriquecen el suelo con distintos compuestos nitrogenados.

El amoníaco, el nitrato y los compuestos nitrogenados son aprovechados por las plantas, y siguiendo el orden de la cadena trófica, el nitrógeno en las plantas pasa a los animales herbívoros y luego a los carnívoros.

Posteriormente, los procesos de descomposición de la materia orgánica incorporan nuevamente nitrógeno al suelo en forma de amoníaco. Esto ocurre mediante la descomposición de cuerpos muertos de animales o plantas y desechos como la orina o heces. A este proceso se le llama amonificación.

El amoníaco proveniente de las bacterias fijadoras, la descomposición de los cuerpos o los desechos, se convierte en amonio el cual es descompuesto en Nitritos y Nitratos.

El ciclo se completa con la desnitrificación, que ocurre mediante bacterias desnitrificantes las cuales descomponen las moléculas de nitritos y nitratos obteniendo energía para vivir y liberando de vuelta a la atmósfera el nitrógeno en estado gaseoso para que el ciclo vuelva a comenzar.

5.3.2. Modelo conceptual

En el siguiente ejemplo, se simula el transporte de nitratos (NO_3^-) en un medio poroso con una porosidad de 0.2. La malla del dominio de simulación tiene dimensiones de 1,010 m de largo, 1 m de ancho y 60 m de altura, discretizada en celdas de $10\text{ m} \times 10\text{ m} \times 1\text{ m}$, formando 6 capas, 101 columnas y una fila. La simulación abarca un periodo total de 600 días, divididos en tres periodos de estrés de 200 días cada uno.

El modelo considera el transporte de una única especie química disuelta ($k = 1$), representando la evolución de la concentración de nitratos en el flujo subterráneo. La densidad del material del suelo se define como $\rho_b = 0.25\text{ kg/m}^3$, mientras que los procesos de degradación y adsorción de los nitratos en la matriz del suelo se modelan mediante reacciones de primer orden con coeficientes de decaimiento:

- ◇ Para la fase disuelta: $\lambda_1 = 0.002$ (desnitrificación en la fase acuosa).
- ◇ Para la fase adsorbida: $\lambda_2 = 0.002$ (interacción con la matriz del suelo).

Además, se considera un coeficiente de dispersión de $D_{ij} = 1.0 \times 10^{-10}$ representando la mezcla del soluto en el medio poroso.

Condiciones iniciales y de frontera

En la primera fila y columna de todas las capas de esta malla se considera una inyección constante de soluto con una concentración constante $= 1\text{ kg/m}^3$.

$$q_s C_s(1, 1, z, 0) = 1$$

Existe una recarga en la primera capa, primera fila y todas las columnas a una concentración de 1.5 kg/m^3

$$q_s C_s(1, y, 1, t) = 1.5 \quad 0 \leq t \leq 200$$

Para el segundo periodo de estrés hay una recarga en la primera capa a una concentración de 2.5 kg/m^3

$$q_s C_s(1, y, 1, t) = 2.5 \quad 201 \leq t \leq 400$$

y una pérdida por evapotranspiración de 1.5 kg/m^3

$$q_s C_s(1, y, 1, t) = -1.5 \quad 201 \leq t \leq 400$$

En el último periodo de estrés la recarga es igual a 0 y la pérdida por evapotranspiración igual a 1 kg/m^3

$$q_s C_s(1, y, 1, t) = -1 \quad 401 \leq t \leq 600$$

Tras la descripción del problema se presenta la siguiente ilustración para un mejor entendimiento del ejemplo estudiado.

En la imagen se considera un medio poroso con una malla como la que se describe en párrafos anteriores, haciendo un acercamiento en las celdas de interés y se observa un pozo, ubicado en la capa número 5, columna 50.

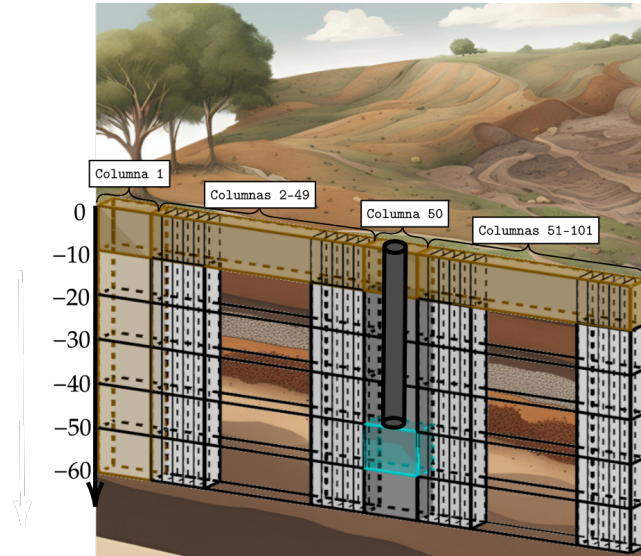


Figura 5.6: Modelo figurativo.

5.3.3. Modelo matemático

En este estudio, se considera un modelo donde el nitrato experimenta los siguientes procesos: advección, dispersión y decaimiento de primer orden, lo que considera una transformación química o biológica del nitrato.

El decaimiento de primer orden suele representar la desnitrificación, que como ya se explicó anteriormente es un proceso biológico que convierte al nitrógeno en formas gaseosas reduciendo así su concentración en el acuífero.

Para este ejemplo, es necesario realizar algunas sustituciones en la ecuación advección-dispersión-reacción.

Para comenzar, el lado izquierdo de la ecuación (3.18) se puede expandir en dos términos

$$\frac{\partial(\theta C^k)}{\partial t} = \theta \frac{\partial C^k}{\partial t} + C^k \frac{\partial \theta}{\partial t} = \theta \frac{\partial C^k}{\partial t} + q'_s C^k \quad (5.8)$$

Considerando que para esta prueba, existen sólo dos tipos de reacciones químicas (sorción y decaimiento de primer orden), el término $\sum R_n$ de la ecuación (3.18) se puede expresar como

$$\Sigma R_n = -\rho_b \frac{\partial \bar{C}^k}{\partial t} - \lambda_1 \theta C^k - \lambda_2 \rho_b \bar{C}^k \quad (5.9)$$

Substituyendo las ecuaciones (5.8) y (5.9) en (3.18) se obtiene

$$\theta \frac{\partial C^k}{\partial t} + \rho_b \frac{\partial \bar{C}^k}{\partial t} = \frac{\partial}{\partial x_i} \left(\theta D_{ij} \frac{\partial C^k}{\partial x_j} \right) - \frac{\partial}{\partial x_i} (\theta v_i C^k) + q_s C_s^k - q'_s C^k - \lambda_1 \theta C^k - \lambda_2 \rho_b \bar{C}^k \quad (5.10)$$

donde

ρ_b : densidad del material del medio

\bar{C}^k : es la concentración de especies k absorbidas en el material del subsuelo

q'_s : es el cambio en el almacenamiento de agua por unidad de volumen

λ_1 : es la velocidad de reacción de primer orden para la fase disuelta

λ_2 : es la velocidad de reacción de primer orden para la fase absorbida (sólida).

Los demás parámetros de la ecuación ya han sido mencionados en la sección 3.3.

Continuando con el ejemplo, sustituimos los datos mencionados en la ecuación (5.10)

$$(0.2) \frac{\partial C^k}{\partial t} + 0.25 \frac{\partial \bar{C}^k}{\partial t} = \frac{\partial}{\partial x_i} \left((0.2)(1.0 \times 10^{-10}) \frac{\partial C^k}{\partial x_j} \right) - \frac{\partial}{\partial x_i} ((0.2)v_i C^k) + q_s C_s^k - q'_s C^k - (0.002)(0.2) C^k - (0.002)(0.2)(0.25) \bar{C}^k \quad (5.11)$$

Este modelo permite evaluar el impacto de la advección, dispersión y degradación en la concentración de nitratos a lo largo del tiempo, proporcionando información clave para comprender su movilidad y persistencia en acuíferos.

5.3.4. Implementación en MODFLOW 6 y MT3D-USGS

Para simular el transporte de nitratos se utiliza MODFLOW 6 para modelar el flujo subterráneo y MT3D-USGS para resolver la ecuación de transporte.

El primer paso consiste nuevamente en la importación de las bibliotecas necesarias para la implementación en Python.

```
[1]: import os
import sys
#from tempfile import TemporaryDirectory

import matplotlib as mpl
import numpy as np
import pandas as pd

import flopy

print(sys.version)
print(f"numpy version: {np.__version__}")
print(f"matplotlib version: {mpl.__version__}")
print(f"flopy version: {flopy.__version__}")
```

También es necesario indicar las rutas en las que se estará trabajando y los directorios en los que se van a almacenar los archivos de salida.

```
[2]: # temporary directory

#temp_dir = TemporaryDirectory()

#model_ws = temp_dir.name

model_ws = '.\\' # Se establece el directorio de trabajo del
modelo (model_ws)

modelpth = os.path.join(model_ws, "p01SpatialStresses_4mf6_mt")
# define la ruta completa del modelo

print(modelpth)

modelname = "p01SpatialStresses" #Define el nombre del modelo
mtexe = "mt3d-usgs_1.1.0_64.exe" # Especifica la ruta del
ejecutable de MT3D-USGS (mtexe)
```

```
[3]: mt = flopy.mt3d.Mt3dms(
    modelname=modelname, # asigna el nombre del modelo definido
    anteriormente.
    model_ws=modelpth, # establece el directorio donde el modelo
    guardará y leerá archivos.
    version="mt3d-usgs", # especifica la versión del software
    MT3D-USGS.
    namefile_ext="onam", # define la extensión del archivo de
    nombre como .nam.
```

```

exe_name="mt3d-usgs_1.1.0_64", # ruta del ejecutable
MT3D-USGS

ftlfree=True # formato libre para la entrada de datos
)

```

Posteriormente es necesario construir el modelo de flujo utilizando MODFLOW 6, donde se define:

- ◊ Geometría del dominio: Dimensiones de 1,010 m × 1 m × 60 m, discretizado en una malla de 6 capas, 101 columnas y 1 fila.
- ◊ Condiciones de frontera: Se establecen condiciones de carga hidráulica fija en los extremos del dominio para generar el flujo subterráneo.
- ◊ Períodos de simulación: Se modelan tres periodos de estrés de 200 días cada uno.

Para definir estos parámetros se utiliza el paquete Basic Transport Package (BTN) de MT3D-USGS. Este paquete establece la configuración inicial del modelo de transporte, incluyendo el dominio de simulación, las condiciones iniciales y las propiedades generales del soluto.

```

[4]: nlay=6 # numero de capas
nrow=1 # numero de filas
ncol=101 # numero de columnas
sconc = np.zeros((nlay,nrow,ncol),dtype=float) # matriz de
concentración inicial del soluto en el dominio.
sconc[:,0,0]=1.0E+03 # concentracion inicial ci

icbund = np.ones((nlay,nrow,ncol),dtype=int) # matriz que indica
en qué celdas se permite el transporte del soluto.
icbund[:,0,0]=-1 #primer columna de cada capa, el valor negativo
indica una concentración fija.

dz = np.ones((nlay,nrow,ncol),dtype=float)
dz=dz * 10.0 # grosor de cada capa, delta z

btn = flopy.mt3d.Mt3dBtn( mt,
    nlay=6,
    nrow=1,
    ncol=101,
    nper=3, # periodos de estrés
    ncomp=1, # Num.total especies quimicas
    mcomp=1, # Num.especies móviles
    laycon= [1,1,1,1,1,1], # concentracion en cada capa
    delr=10, # longitud en cada celda en dirección x

```

```

delc=1, # longitud en cada celda en dirección y
htop=60, # Altura inicial
dz=dz,
perlen=200, # Duración de cada periodo de estrés
nstp=10, # paso de tiempo
dt0=50,
tsmult=1, # multiplicador de paso de tiempo
prcity=.25, # porosidad
sconc = sconc,
unitnumber=31, # número de unidad del archivo
ifmtcn=1, # formato de impresión de la concentración
icbund = icbund
)

```

Seguido del paquete BTN, se configura el Paquete de Advección (ADV), el cual se encarga de definir el esquema numérico utilizado para modelar el proceso de advección del soluto en el medio poroso. El paquete ADV es clave para modelar la movilidad de los nitratos en un acuífero.

```

[5]: mixelm = 0 # indica que se utiliza el esquema de advección
      estándar de diferencias finitas.
percel = 0.50000 # tamaño de la celda de contorno
mxpart = 800000 # define el número máximo de partículas
      permitidas en la simulación,
nadvfd = 1 #

adv = flogy.mt3d.Mt3dAdv(
      mt, mixelm=mixelm, percel=percel, mxpart=mxpart,
nadvfd=nadvfd
)

```

El siguiente paso es configurar el Paquete de Dispersión (DSP), el cual define los parámetros de dispersión hidráulica en el modelo de transporte de solutos. Este paquete incluye tanto la dispersión mecánica como la difusión molecular del soluto.

```

[6]: al = 10 # dispersividad longitudinal
trpt = 0.1 # relación entre la dispersividad transversal y la
      longitudinal
trpv = 0.01 # relación entre la dispersividad vertical y la
      longitudinal.
dmcoef = 1.0000e-9 # coeficiente de difusión molecular

dsp = flogy.mt3d.Mt3dDsp(

```

```

    mt, al=al, trpt=trpt, trpv=trpv, dmcoef=dmcoef,
    multiDiff=False
)

```

Después de configurar los paquetes anteriores, se define el Paquete del Solucionador GCG (Generalized Conjugate Gradient) el cual es responsable de la resolución numérica de la ecuación de transporte, asegurando que la solución sea estable y eficiente.

```

[7]: mxiter = 10000 # número máximo de iteraciones totales del
      solver
      iter1 = 50 # número máximo de iteraciones internas por paso
      temporal
      isolve = 3 # especifica el método de solución
      ncrs = 0 # es un parámetro que controla la estrategia de
      relajación cruzada
      accl = 1.000000 # factor de aceleración para la convergencia del
      solver
      cclose = 1.00e-06 # criterio de convergencia para el solver
      iprgcg = 0 # frecuencia de impresión de la información del
      solver

      gcg = flopy.mt3d.Mt3dGcg(
          mt,
          mxiter=mxiter,
          iter1=iter1,
          isolve=isolve,
          ncrs=ncrs,
          accl=accl,
          cclose=cclose,
          iprgcg=iprgcg,
      )

```

Uno de los paquetes más importantes es el Paquete de Fuentes y Sumideros (SSM - Source and Sink Mixing Package). Este paquete permite definir fuentes y sumideros de soluto en el modelo de transporte, asignando concentraciones a las entradas y salidas de agua subterránea dentro del dominio de simulación.

La variable `itype` es una de la más importantes dentro de este paquete, ya que representa el tipo de fuente o sumidero. Cada uno de ellos se representa como:

- ◇ 1: Pozo (WEL)
- ◇ 2: Recarga (RCH)
- ◇ 3: Drenaje (DRN)

- ◇ 4: Flujo de río (RIV)
- ◇ 5: Frontera de concentración constante (CHD)
- ◇ 7: Evapotranspiración (EVT)
- ◇ 8: Descarga
- ◇ 15: Fuente o sumidero general (GHB).

```
[9]: # Esta es la celda original para construir ssm_data
itype = flopy.mt3d.Mt3dSsm.itype_dict()
print(itype)
print(flopy.mt3d.Mt3dSsm.get_default_dtype())

flota=float(0.0)
flota1 = float(-1.0)

ssm_data = {}
# Con la siguiente instrucción se indica que hay un pozo en la
# capa 5, fila 1, columna 50
ssm_data[0] = [(4, 0, 49, flota, itype['WEL'])]
# Con la siguiente instrucción se indica que hay un proceso de
# evapotranspiración en la capa 1, fila 1, todas las columnas
for j in range(ncol):
    ssm_data[0].append((0, 0, j, 1.5, 7))
# Con la siguiente instrucción se indica que hay un proceso de
# descarga en la capa 1, fila 1, todas las columnas
for j in range(ncol):
    ssm_data[0].append((0, 0, j, flota, 8))

ssm_data[1] = [(4, 0, 49, 0.0, itype['WEL'])]
for j in range(ncol):
    ssm_data[1].append((0, 0, j, 2.5, 7))
for j in range(ncol):
    ssm_data[1].append((0, 0, j, -1.5, 8))

ssm_data[2] = [(4, 0, 49, 0.0, itype['WEL'])]
for j in range(ncol):
    ssm_data[2].append((0, 0, j, flota, 7))
for j in range(ncol):
    ssm_data[2].append((0, 0, j, flota1, 8))

#for k in range(nlay):
#    for i in range(nrow):
#        ssm_data[0].append((k, i, 0, 0.0, itype["GHB"], 0.0, 0.
#        ↪0))
```

```

ssm = flopy.mt3d.Mt3dSsm(
    model = mt,
    dtype=None,
    stress_period_data=ssm_data,
    mxss = 215 )
#ssm.write_file()

```

La última configuración se realiza con el paquete RCT (Reaction Package) usado para modelar reacciones químicas en el transporte de solutos dentro del flujo subterráneo. Este paquete permite simular procesos como degradación, adsorción y descomposición de contaminantes en el medio poroso.

```

[11]: isothm = 1 # Isoterma de adsorción lineal
       ireact = 1 # Reacciones de primer orden
       irtop = 2 # controla la lectura de los coeficientes de
           reacción.
       igetsc = 0 # no se obtendrán concentraciones escalares
           adicionales
       ireaction = 0 # controla opciones de reacción adicionales
       rhob = 0.25 # Densidad de masa del suelo
       rc1 = 0.002 # Tasa de reacción de primer orden para la fase
           disuelta
       rc2 = 0.002 # Tasa de decaimiento en la capa del suelo

       rct = flopy.mt3d.Mt3dRct(
           mt, isothm=isothm, ireact=ireact, igetsc=igetsc, rhob =
           rhob, rc1=rc1, rc2=rc2
       )

```

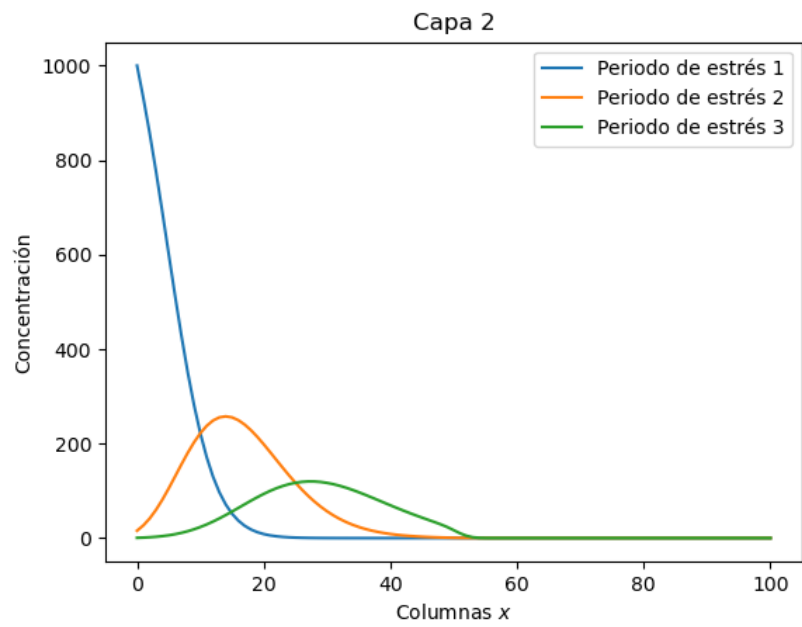
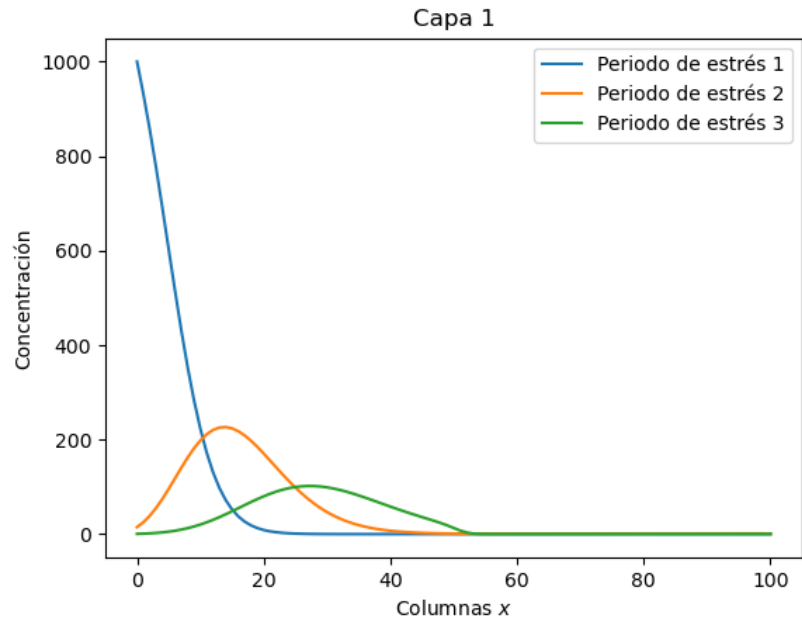
Finalmente, al ejecutar el siguiente comando FloPy genera los archivos de entrada correspondientes a los paquetes activados anteriormente en el modelo, lo que permite ejecutar una simulación de transporte con MT3D-USGS.

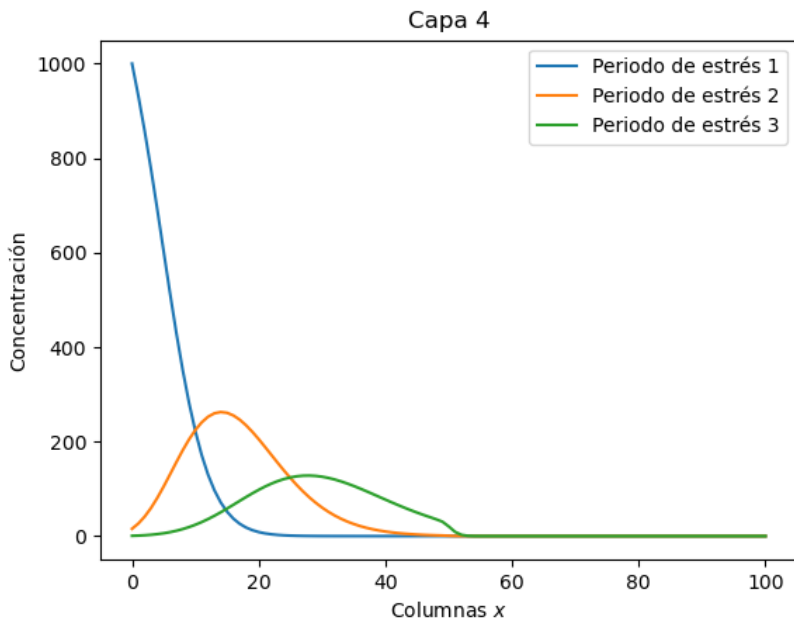
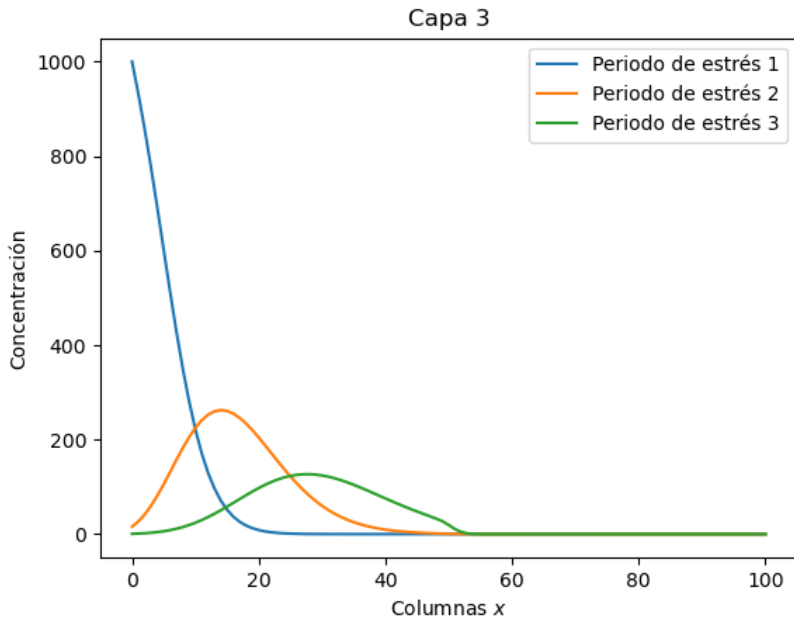
```

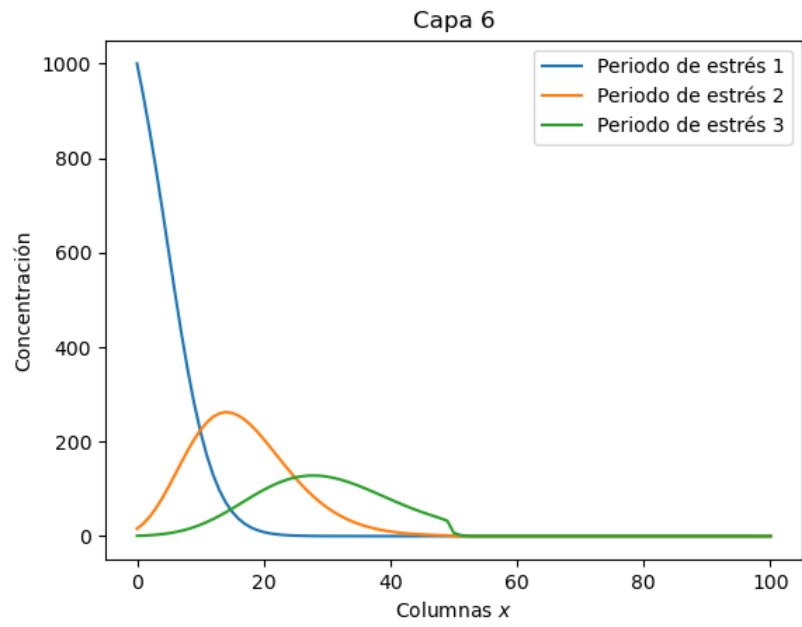
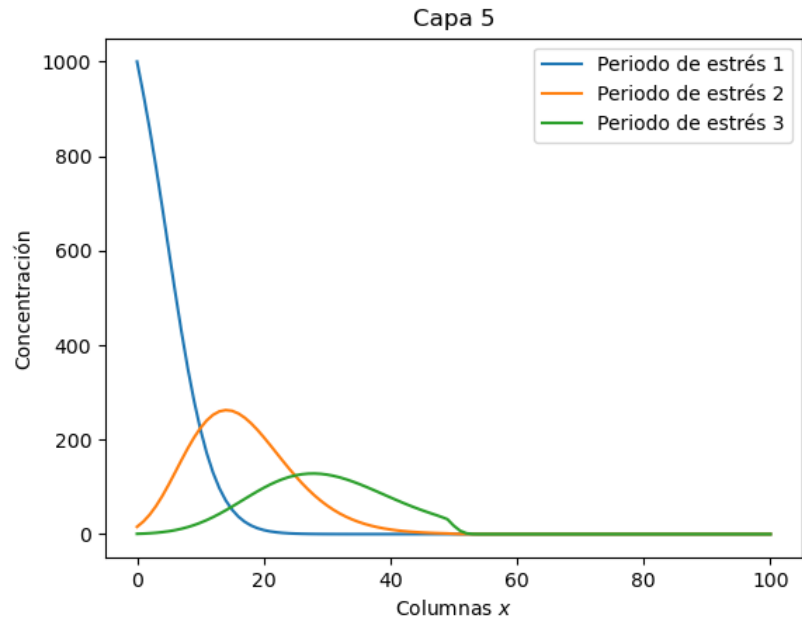
[12]: mt.write_input()

```

Después de crear estos archivos de entrada, se substituyeron en los archivos proporcionados por la documentación de MT3D-USGS para la ejecución del modelo de flujo y transporte del modelo. Obteniendo los siguientes resultados.







Debido a que las gráficas de los resultados presentan tendencias similares en todas las capas, los comentarios respecto a ellos se harán de manera general.

En cada capa, la concentración de nitratos disminuye cuando $x \rightarrow 120$ en todos los periodos de estrés.

La condición inicial $C_0 = 1,000$ se cumple en todas las capas en el primer periodo de estrés, sin embargo esta concentración disminuye rápidamente. Para los siguientes periodos de estrés, la concentración es menor cerca del inicio del dominio pero a lo largo de él comienza a distribuirse.

Las formas de las gráficas para los periodos de estrés 2 y 3 indican que el nitrato se transporta y se dispersa. Al ser la curva más extendida en el periodo de estrés 3 podemos decir que el efecto de dispersión está más marcado conforme transcurre el tiempo.

Conclusiones

El estudio de los fundamentos teóricos de los modelos matemáticos de flujo en medios porosos y del transporte de especies de nitrógeno permitió reforzar los conocimientos teóricos y numéricos para el entendimiento e implementación de su resolución numérica. A través del análisis de los principios que rigen estos procesos, se logró una comprensión más profunda de los mecanismos involucrados en el movimiento del agua y la dispersión de solutos en acuíferos.

El desarrollo de estos conceptos facilitó la construcción de una base teórica y conceptual adecuada para la aplicación de métodos numéricos en la simulación de escenarios hidrogeológicos, aspecto fundamental para evaluar la dinámica del flujo y la evolución de contaminantes en medios porosos.

Asimismo, el análisis del ciclo del nitrógeno y de las ecuaciones que describen su transporte en medios porosos contribuyó a profundizar en la dinámica de este elemento en sistemas subterráneos.

Por otra parte, el estudio de problemas de referencia que incorporan nuevas características y paquetes del software utilizado en el desarrollo de esta tesis permitió verificar la precisión de nuevos códigos mediante la comparación con soluciones analíticas y con los resultados de otros modelos existentes.

La participación en el taller del Instituto de Geofísica fue un elemento clave en el desarrollo y culminación de este trabajo. Durante dicho proceso, se adquirieron habilidades prácticas en el uso del software empleado para realizar simulaciones computacionales de modelos de flujo y transporte.

Además de elaborar modelos conceptuales y representaciones figurativas de los casos estudiados, se colaboró de forma conjunta con otros participantes en el desarrollo de nuevos códigos utilizados en el tercer ejemplo presentado en el capítulo 5. Esta experiencia interdisciplinaria destacó la importancia de abordar los problemas desde diferentes perspectivas y puso en evidencia el valor de la modelación matemática como herramienta de análisis y gestión en sistemas hidrogeológicos.

Considerando que uno de los objetivos de esta tesis es servir como apoyo para

futuras investigaciones sobre la contaminación por nitrógeno, el contenido fue redactado con la intención de proporcionar una guía clara para quienes deseen iniciar un proyecto de investigación en este campo.

Los códigos desarrollados fueron explicados de manera detallada para facilitar su implementación y servir como ejercicios introductorios en el uso de MODFLOW 6 y MT3D-USGS.

Estos conocimientos no solo fortalecen la base teórica y numérica para la modelación de sistemas hidrogeológicos, sino que también contribuyen a la toma de decisiones en la gestión y conservación de los recursos hídricos subterráneos, particularmente en lo referente al control y mitigación de la contaminación por nitrógeno.

Anexos

Anexo 1. ruta_temporal.py

Código utilizado para agregar las rutas temporales correctas en cada simulación.

```
import os, sys
c_path = os.getcwd()
l_path = c_path.split(sep="\\")
i_wma = l_path.index('FyT')
a_path = '\\'.join(l_path[:i_wma+1])
src_path = 'C:\\Users\\ADMON\\FyT\\src\\ft1D'
if not(src_path in sys.path[0]):
    sys.path.insert(0, os.path.abspath(a_path + src_path))
```

Anexo 2. xmf6.py

Código utilizado para personalizar el estilo de los datos de salida y para la simplificación de los parámetros de entrada en la simulación de flujo y transporte.

```
import os, sys # Interfaces con el sistema operativo.
import numpy as np # Manejo de arreglos numéricos
multidimensionales
import matplotlib.pyplot as plt # Graficación

# Biblioteca y módulos de flopy
import flopy
from flopy.plot.styles import styles

from colorama import Fore, Style, Back

class MeshDis():
    def __init__(self, nrow = 1, ncol = 1, nlay = 1,
                 column_length = 1.0, row_length = 1.0,
                 top = 0.0, bottom = 0.0):
        self.__nx = ncol # Number of columns
```

```

self.__ny = nrow # Number of rows
self.__nz = nlay # Number of layers
self.__lx = row_length
self.__ly = column_length
self.__lz = np.abs(top - bottom)
self.__dx = self.__calc_dx()
self.__dy = self.__calc_dy()
self.__dz = 1.0
self.__top = top # Top of the model
self.__bottom = bottom # Layer bottom elevation

def print(self):
    print('NX : {:8d} (ncol)'.format(self.__nx))
    print('NY : {:8d} (nrow)'.format(self.__ny))
    print('NZ : {:8d} (nlay)'.format(self.__nz))
    print('LX : {:8.5} (row)'.format(self.__lx))
    print('LY : {:8.5} (col)'.format(self.__ly))
    print('LZ : {:8.5} (lay)'.format(self.__lz))
    print('DX : {:8.5} (delr)'.format(self.__dx))
    print('DY : {:8.5} (delc)'.format(self.__dy))
    print('DZ : {:8.5} (top-botm)'.format(self.__dz))

@property
def ncol(self):
    return self.__nx

@ncol.setter
def ncol(self, ncol):
    if ncol > 0:
        self.__nx = ncol
        self.__calc_dx()
    else:
        print('El valor {} no es válido para ncol, debe
ser mayor que cero'.format(ncol))

@property
def nrow(self):
    return self.__ny

@nrow.setter
def nrow(self, nrow):
    if nrow > 0:
        self.__ny = nrow
        self.__calc_dy()

```

```

        else:
            print('El valor {} no es válido para nrow, debe
ser mayor que cero'.format(nrow))

@property
def nlay(self):
    return self.__nz

@nlay.setter
def nlay(self, nlay):
    if nlay > 0:
        self.__nz = nlay
        ## TODO: calcular un delta para la direccion z
    else:
        print('El valor {} no es válido para nlay, debe
ser mayor que cero'.format(nlay))

@property
def row_length(self):
    return self.__lx

@row_length.setter
def row_length(self, r_l):
    if r_l > 0:
        self.__lx = r_l
    else:
        print('El valor {} no es válido para row_length,
debe ser mayor que cero'.format(r_l))

@property
def col_length(self):
    return self.__ly

@col_length.setter
def col_length(self, c_l):
    if c_l > 0:
        self.__ly = c_l
    else:
        print('El valor {} no es válido para col_length,
debe ser mayor que cero'.format(c_l))

@property
def lay_length(self):
    return self.__lz

```

```

@lay_length.setter
def lay_length(self, l_1):
    if l_1 > 0:
        self.__lz = l_1
    else:
        print('El valor {} no es válido para lay_length,
debe ser mayor que cero'.format(l_1))

@property
def delr(self):
    return self.__dx

@property
def delc(self):
    return self.__dy

@property
def delz(self):
    return self.__dz

@property
def top(self):
    return self.__top

@property
def bottom(self):
    return self.__bottom

def __calc_dx(self):
    return self.__lx / self.__nx

def __calc_dy(self):
    return self.__ly / self.__ny

def __calc_dz(self):
    return self.__lz / self.__nz

def get_coords(self):
    return (np.linspace(0.5 * self.__dx, self.__lx -
0.5 * self.__dx, self.__nx),
            np.linspace(0.5 * self.__dy, self.__ly -
0.5 * self.__dy, self.__ny),
            np.linspace(0.5 * self.__dz, self.__lz -
0.5 * self.__dz, self.__nz))

```

```

        def get_grid(self):
            xg, yg, zg = np.meshgrid(np.linspace(0, self.__lx,
self.__nx),
                                     np.linspace(0, self.__ly,
self.__ny),
                                     np.linspace(0, self.__lz,
self.__nz))

        def get_dict(self):
            return {'row_length': self.__lx, 'col_length': self._
↪_ly, 'lay_length': self.__lz,
                    'ncol': self.__nx, 'nrow': self.__ny, 'nlay':
self.__nz,
                    'delr': self.__dx, 'delc': self.__dy, 'dell':
self.__dz,
                    'top': self.__top, 'bottom': self.__bottom}

    def plot_flow_1D(gwf, mesh, os_par, oc_par, savefig = False):
        """
        Función para graficar los resultados.

        Paramaters
        -----
        gwf: ModflowGwf
        Objeto del modelo de flujo GWF

        mesh: MeshDis
        Objeto que gestiona atributos y métodos de una malla
        rectangular
        estructurada y uniforme.

        os_par: dict
        Parámetros para ejecución de MODFLOW 6, archivos de
        salida y
        path del workspace.
        """
        # Obtenemos los resultados de la carga hidráulica
        head = flopy.utils.HeadFile(
            os.path.join(os_par["ws"],
                oc_par["head_file"])).get_data()

        print('head_L = {} \t head_R = {}'.format(head[0,0,0],
head[0,0,-1]))

        # Obtenemos los resultados del BUDGET

```

```

bud = flopy.utils.CellBudgetFile(
    os.path.join(os_par["ws"],
                 oc_par["fbudget_file"]),
    precision='double'
)
# Obtenemos las velocidades
spdis = bud.get_data(text='DATA-SPDIS')[0]
qx, qy, qz = flopy.utils.postprocessing.get_specific_
↪ discharge(spdis, gwf)

with styles.USGSPlot():
    plt.rcParams['font.family'] = 'DeJavu Sans'
    x, _, _ = mesh.get_coords()
    plt.figure(figsize=(10,3))
    plt.plot(x, head[0, 0], marker=".", ls="--", mec=
↪ "blue", mfc="none", markersize="1", label = 'Head')
    plt.xlim(0, 12)
    plt.xticks(ticks=np.linspace(0, mesh.row_length,13))
    plt.xlabel("Distance (cm)")
    plt.ylabel("Head (unitless)")
    plt.legend()
    plt.grid()

    if savefig:
        plt.savefig('head.pdf')
    else:
        plt.show()

from wexler1 import sol_analytical_t
def plot_tran_1D(sim, mesh, tm_par, ph_par, os_par, oc_par,
savefig = False):
    """
    Función para graficar los resultados.

    Paramaters
    -----
    gwf: ModflowGwf
    Objeto del modelo de flujo GWF

    mesh: MeshDis
    Objeto que gestiona atributos y métodos de una malla
    rectangular
    estructurada y uniforme.

    os_par: dict

```

```

        Parámetros para ejecución de MODFLOW 6, archivos de
salida y
path del workspace.
"""
mf6gwt_ra = sim.get_model("transport").obs.output.obs().
↪data
ucnobj_mf6 = sim.transport.output.concentration()
simtimes = mf6gwt_ra["totim"]
obsnames = ["X005", "X405", "X1105"]

with styles.USGSPlot():
    plt.rcParams['font.family'] = 'DeJavu Sans'

    fig, axs = plt.subplots(2, 1, figsize=(5,6), tight_
↪layout=True)

    iskip = 5

    atimes = np.arange(0, tm_par["total_time"], 0.1)

    for i, x in enumerate([0.05, 4.05, 11.05]):
        a1, idx_filter = sol_analytical_t(i, x,
atimes, mesh, ph_par)

        axs[0].plot(atimes[idx_filter], a1[idx_filter],
color="k", label="ANALYTICAL")

        axs[0].plot(simtimes[:, :iskip], mf6gwt_
↪ra[obsnames[i]][: :iskip],
                    marker="o", ls="none", mec="blue",
mfc="none", markersize="4",
                    label="MODFLOW 6")
        axs[0].set_ylim(-0.05, 1.2)
        axs[0].set_xlim(0, 120)
        axs[0].set_xlabel("Time (seconds)")
        axs[0].set_ylabel("Normalized Concentration
(unitless)")

    ctimes = [6.0, 60.0, 120.0]
    x, _, _ = mesh.get_coords()
    for i, t in enumerate(ctimes):
        a1, idx_filter = sol_analytical_t(i, x, t, mesh,
ph_par, False)

        axs[1].plot(x, a1, color="k", label="ANALYTICAL")

```

```

        simconc = ucnoobj_mf6.get_data(totim=t).flatten()
        axs[1].plot(x[:, :iskip], simconc[:, :iskip],
                    marker="o", ls="none", mec="blue",
mfc="none", markersize="4",
                    label="MODFLOW 6")
        axs[1].set_ylim(0, 1.1)
        axs[1].set_xlim(0, 12)
        axs[1].set_xlabel("Distance (cm)")
        axs[1].set_ylabel("Normalized Concentration
(unitless)")

    if savefig:
        plt.savefig('conc.pdf')
    else:
        plt.show()

def nice_print(dic, message = ''):
    print(Fore.GREEN)
    print(message)
    print('{:~30}'.format(30*'-') + Style.RESET_ALL)
    for k,v in dic.items():
        print('{:>20} = {:<10}'.format(k, v))

if __name__ == '__main__':
    mesh = MeshDis(
        nrow = 1,      # Number of rows
        ncol = 120,    # Number of columns
        nlay = 1,      # Number of layers
        row_length = 12.0, # Length of rows
        column_length = 0.1, # Length of columns
        top = 1.0,     # Top of the model
        bottom = 0,    # Layer bottom elevation
    )
    mesh.print()
    nice_print(mesh.get_dict(), 'Space discretization')

```

Anexo 3. wexler1.py

Código utilizado para calcular la solución analítica del ejemplo de transporte.

```

import numpy as np

class Wexler1d:

```

```

    """
    Analytical solution for 1D transport with inflow at a
    concentration of 1.
    at x=0 and a third-type bound at location l.
    Wexler Page 17 and Van Genuchten and Alves pages 66-67
    """

    def betaeqn(self, beta, d, v, l):
        return beta / np.tan(beta) - beta**2 * d / v / l + v * l
        / 4.0 / d

    def fprimebetaeqn(self, beta, d, v, l):
        """
        f1 = cotx - x/sinx2 - (2.0D0*C*x)
        """
        c = v * l / 4.0 / d
        return 1.0 / np.tan(beta) - beta / np.sin(beta) ** 2 - 2.
        ↪ 0 * c * beta

    def fprime2betaeqn(self, beta, d, v, l):
        """
        f2 = -1.0D0/sinx2 - (sinx2-x*DSIN(x*2.0D0))/(sinx2*sinx2)
        - 2.0D0*C
        """
        c = v * l / 4.0 / d
        sinx2 = np.sin(beta) ** 2
        return (
            -1.0 / sinx2
            - (sinx2 - beta * np.sin(beta * 2.0)) / (sinx2 *
            sinx2)
            - 2.0 * c
        )

    def solvebetaeqn(self, beta, d, v, l, xtol=1.0e-12):
        from scipy.optimize import fsolve

        t = fsolve(
            self.betaeqn,
            beta,
            args=(d, v, l),
            fprime=self.fprime2betaeqn,
            xtol=xtol,
            full_output=True,

```

```

)
result = t[0][0]
infod = t[1]
isoln = t[2]
msg = t[3]
if abs(result - beta) > np.pi:
    raise Exception("Error in beta solution")
err = self.betaeqn(result, d, v, l)
fvec = infod["fvec"][0]
if isoln != 1:
    print("Error in beta solve", err, result, d, v, l,
msg)

return result

def root3(self, d, v, l, nval=1000):
    b = 0.5 * np.pi
    betalist = []
    for i in range(nval):
        b = self.solvebetaeqn(b, d, v, l)
        err = self.betaeqn(b, d, v, l)
        betalist.append(b)
        b += np.pi
    return betalist

def analytical(self, x, t, v, l, d, tol=1.0e-20, nval=5000):
    sigma = 0.0
    betalist = self.root3(d, v, l, nval=nval)
    concold = None
    for i, bi in enumerate(betalist):
        denom = bi**2 + (v * l / 2.0 / d) ** 2 + v * l / d
        x1 = (
            bi
            * (bi * np.cos(bi * x / l) + v * l / 2.0 / d *
np.sin(bi * x / l))
            / denom
        )

        denom = bi**2 + (v * l / 2.0 / d) ** 2
        x2 = np.exp(-1 * bi**2 * d * t / l**2) / denom

        sigma += x1 * x2
        term1 = 2.0 * v * l / d * np.exp(v * x / 2.0 / d -
v**2 * t / 4.0 / d)
        conc = 1.0 - term1 * sigma
        if i > 0:

```

```

        assert concold is not None
        diff = abs(conc - concold)
        if np.all(diff < tol):
            break
        concold = conc
    return conc

def analytical2(self, x, t, v, l, d, e=0.0, tol=1.0e-20,
nval=5000):
    """
        Calculate the analytical solution for one-dimension
        advection and
        dispersion using the solution of Lapidus and Amundson
        (1952) and
        Ogata and Banks (1961)

        Parameters
        -----
        x : float or ndarray
            x position
        t : float or ndarray
            time
        v : float or ndarray
            velocity
        l : float
            length domain
        d : float
            dispersion coefficient
        e : float
            decay rate

        Returns
        -----
        result : float or ndarray
            normalized concentration value

    """
    u = v**2 + 4.0 * e * d
    u = np.sqrt(u)
    sigma = 0.0
    denom = (u + v) / 2.0 / v - (u - v) ** 2.0 / 2.0 / v / (u
+ v) * np.exp(
        -u * l / d
    )

```

```

        term1 = np.exp((v - u) * x / 2.0 / d) + (u - v) / (u + v)
* np.exp(
    (v + u) * x / 2.0 / d - u * l / d
)
    term1 = term1 / denom
    term2 = 2.0 * v * l / d * np.exp(v * x / 2.0 / d - v**2 *
t / 4.0 / d - e * t)
    betalist = self.root3(d, v, l, nval=nval)
    concold = None
    for i, bi in enumerate(betalist):
        denom = bi**2 + (v * l / 2.0 / d) ** 2 + v * l / d
        x1 = (
            bi
            * (bi * np.cos(bi * x / l) + v * l / 2.0 / d *
np.sin(bi * x / l))
            / denom
        )

        denom = bi**2 + (v * l / 2.0 / d) ** 2 + e * l**2 / d
        x2 = np.exp(-1 * bi**2 * d * t / l**2) / denom

        sigma += x1 * x2

    conc = term1 - term2 * sigma
    if i > 0:
        assert concold is not None
        diff = abs(conc - concold)
        if np.all(diff < tol):
            break
        concold = conc
    return conc

def sol_analytical_t(i, x, atimes, mesh, pparams, x_axis_
    ↪time=True):

    a1 = Wexler1d().analytical2(x, atimes,
        pparams["specific_discharge"] /
pparams["retardation_factor"],
        mesh.row_length,
        pparams["dispersion_coefficient
    ↪"],
        pparams["decay_rate"])

    idx = 0

    if x_axis_time:

```

```

if idx == 0:
    idx_filter = a1 < 0
    a1[idx_filter] = 0
    idx_filter = a1 > 1
    a1[idx_filter] = 0
    idx_filter = atimes > 0
    if i == 2:
        idx_filter = atimes > 79
elif idx > 0:
    idx_filter = atimes > 0
else:
    if idx == 0:
        idx_filter = x > mesh.row_length
        if i == 0:
            idx_filter = x > 6
        if i == 1:
            idx_filter = x > 9
        a1[idx_filter] = 0.0

return a1, idx_filter

```

Índice de figuras

1.1. Figura que representa el experimento realizado por Darcy para formular su ley. Fuente: Elaboración propia.	17
3.1. Volumen de control elemental utilizado para explicar la deducción de la ecuación de flujo. Fuente: Elaboración propia.	28
5.1. Figura que representa el modelo figurativo del ejemplo trabajado en esta sección. Fuente: Elaboración propia.	44
5.2. Gráfica de la solución analítica	46
5.3. Comparación de la solución analítica vs la solución obtenida con MODFLOW 6 de la ecuación de flujo en una dimensión, con las condiciones y características mencionadas en el desarrollo del modelo conceptual explicado en esta sección. Fuente: Elaboración propia.	55
5.4. Figura que representa el modelo figurativo del ejemplo trabajado en esta sección. Fuente: Elaboración propia.	57
5.5. Resultados obtenidos de la simulación de transporte.	65
5.6. Figura que representa el modelo figurativo del ejemplo trabajado en esta sección. Fuente: Elaboración propia.	68

Bibliografía

- [1] M Alexander. Biodegradation and bioremediation. 2nd edn academic press. *San Diego, USA*, 1999.
- [2] R. Alvarez. Estimación de parámetros en un modelo de acuíferos. Master's thesis, Universidad Nacional Autónoma de México, 1997.
- [3] M.P. Anderson and W.W. Woessner. *Applied Groundwater Modeling: Simulation of Flow and Advective Transport*. Academic Press, 2015.
- [4] C Anthony J Appelo and Dieke Postma. *Geochemistry, groundwater and pollution*. CRC press, 2004.
- [5] Teresa Audesirk. *Biología : La Vida En La Tierra*. Pearson, 2017.
- [6] J. Bear. *Dynamics of Fluids in Porous Media*. Elsevier, 1972.
- [7] Jacob Bear. *Modeling groundwater flow and pollution*. Theory and Applications of Transport in Porous Media. Springer, 1987.
- [8] Ralf Conrad. Microbial ecology of methanogens and methanotrophs. *Advances in agronomy*, 96:1–63, 2007.
- [9] Instituto Mexicano de Tecnología del Agua. Aguas subterráneas. <https://www.gob.mx/imta/articulos/aguas-subterraneas>, 2019.
- [10] G. Echeverri. Aspectos teóricos sobre el fenómeno de contaminación de aguas subterráneas. *Revista Universidad Eafit.*, 1998.
- [11] EPTISA. Actualización del modelo matemático de flujo del acuífero crestax (mallorca). *Instituto Geológico y Minero de España*, 2009.
- [12] C. W. Fetter. *Applied hydrogeology*. Prentice Hall, 2001.
- [13] L. Fonseca, E. Trujillo, and G. Peña. Tortuosidad y permeabilidad de materiales cerámicos mesoporosos de caolín y diatomita. *Revista UIS Ingenierías*, 2019.
- [14] R. Freeze and J. Cherry. *Groundwater*. Prentice-Hall, Inc., 1979.

- [15] M. Fuillerat. Revisión de los modelos matemáticos aplicados a la gestión de recursos hídricos o cuencas. Master's thesis, Universidad de Alcalá, 2021.
- [16] Carlos Humberto Galeano Urueña. *Técnicas de solución numérica de la ecuación de Difusión-Advección-Reacción para el estudio de dispersión de contaminantes*. Universidad Nacional de Colombia, 2009.
- [17] R. Galán. Modelación matemática de nitratos en el agua subterránea en la región sur de la ciudad de México. Master's thesis, Universidad Nacional Autónoma de México, 2019.
- [18] J. Hoffman. *Numerical Methods for Engineers and Scientists*. CRC Press, 2001.
- [19] IMCO. Situación del agua en México. <https://imco.org.mx/situacion-del-agua-en-mexico/>, 2023.
- [20] L. F. Konikow, D. J. Goode, and G. Z. Hornberger. A three-dimensional method-of-characteristics solute-transport model (moc3d). *U. S. GEOLOGICAL SURVEY*, 1996.
- [21] C. D. Langevin, J. D. Hughes, E.R. Banta, R. G. Niswonger, S. Panday, and A. M. Provost. *Documentation for the MODFLOW 6 Groundwater Flow Model*. US Geological Survey., 2017.
- [22] P. Loera. Modelos matemáticos para pruebas de bombeo en acuíferos. caso del campo experimental de la unison. *EPISTEMUS*, 2015.
- [23] Francois MM Morel and Janet G Hering. *Principles and applications of aquatic chemistry*. John Wiley & Sons, 1993.
- [24] A. Palomares. Contaminación del agua por nitratos y técnicas para su tratamiento. <https://www.esferadelagua.es/agua-y-tecnologia/contaminacion-del-agua-por-nitratos-y-tecnicas-para-su-tratamiento>, 2013.
- [25] S.B Patil. Contaminant transport through porous media: An overview of experimental and numerical studies. *Korea Science*, 2014.
- [26] Werner Stumm and James J Morgan. *Aquatic chemistry: chemical equilibria and rates in natural waters*. John Wiley & Sons, 2013.
- [27] B. Swistock. Nitratos en el agua potable. *Penn State Extension*, 2023.
- [28] F.J. Sánchez. *Hidrología superficial y subterránea*. Kindle Direct Publishing, 2022.
- [29] E. J. Wexler. *Analytical solutions for one-, two-, and three-dimensional solute transport in ground-water systems with uniform flow*. U. S. GEOLOGICAL SURVEY, 1992.

- [30] C. Zheng and P. P. Wang. *MT3DMS: A modular three-dimensional transport model for simulation of advection, dispersion, and chemical reactions of contaminants in groundwater systems; documentation and user's guide*. US Army Engineer Research and Development Center., 1999.
- [31] Chunmiao Zheng and Gordon D Bennett. *Applied Contaminant Transport Modeling*. Wiley-Interscience, 2002.