



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIAS DE LA TIERRA

INSTITUTO DE GEOFÍSICA

EXPLORACIÓN, AGUAS SUBTERRÁNEAS, MODELACIÓN Y PERCEPCIÓN REMOTA

Modelo de flujo y transporte en medios porosos en FEniCS
usando el método de elementos finitos mixtos

Aplicación a un caso de estudio para la simulación de un proceso
de inyección de agua de baja salinidad a escala de laboratorio

T E S I S

QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN CIENCIAS DE LA TIERRA
(MODELACIÓN)

P R E S E N T A:

Juan Eduardo Linares Pérez

DIRECTOR DE TESIS:

Dr. Martín Alberto Díaz Viera

Instituto Mexicano del Petróleo

COMITÉ TUTOR:

Dr. Ernesto Rubio Acosta

Instituto de Investigaciones

en Matemáticas Aplicadas y en Sistemas, UNAM

Dr. David Parra Guevara

Centro de Ciencias de la Atmósfera, UNAM

Ciudad Universitaria, Febrero de 2018.

Resumen

En este trabajo de tesis se desarrolló un modelo general de flujo y transporte en medios porosos a escala de laboratorio, el cual fue aplicado para la simulación del desprendimiento, migración y bloqueo por finos durante un proceso de inyección de agua de baja salinidad (LSWF, por sus siglas en inglés).

El proceso de inyección de agua de baja salinidad es considerado como un método de recuperación mejorada y consiste en inyectar agua con una salinidad menor al agua presente en el yacimiento provocando una serie de efectos entre los que se incluye los fenómenos de desprendimiento, migración y bloqueo por finos que pueden modificar la permeabilidad de la roca y por consiguiente alterar el flujo en el yacimiento.

En el desarrollo del modelo se utilizó una metodología de modelación sistemática y desde el punto de vista metodológico, se describen cada una de las etapas, que consiste en los modelos conceptual, matemático, numérico y computacional.

Se obtuvo un modelo matemático general de flujo y transporte en medio porosos, aplicando la formulación axiomática de la modelación de medios continuos, donde el modelo de flujo es monofásico ligeramente compresible, mientras que el modelo de transporte es multicomponente e incluye varios fenómenos físico-químicos relevantes tales como advección, difusión, dispersión, adsorción-desorción y diversas reacciones.

El sistema de ecuaciones diferenciales parciales no-lineal se resolvió numéricamente en el espacio mediante el método de elementos finitos; mientras que en el tiempo se aplicó el método de diferencias finitas hacia atrás, resultando un esquema numérico completamente implícito. Las tres diferentes formulaciones del método de elementos finitos: clásica (CL), el mixta (MX) y el mixta-dual (MD), fueron validadas para un caso de referencia y se realizó una comparación en términos del orden de convergencia y su desempeño, resultando ser más óptimo el método mixto-dual, mostrando una mejor aproximación tanto en el cálculo de la velocidad como en el de la presión.

La implementación computacional se realizó mediante el lenguaje de programación Python 2.7 utilizando el proyecto FEniCS 2017.1, el cual es un software de código abierto y multiplataforma (Windows/Linux).

Finalmente, el modelo de flujo y transporte acoplado fue aplicado para un caso de estudio de desprendimiento, migración y bloqueo por finos. Las soluciones numéricas obtenidas con la implementación en FEniCS 2017.1 son comparadas con las obtenidas con el software comercial COMSOL Multiphysics®.

Abstract

In this thesis was developed a general model of flow and transport in porous media at laboratory scale, which was applied for the simulation of detachment, migration and blockage by fines during an injection process, of low salinity water (LSWI).

The process of injecting low salinity water, considered as an improved recovery method, consists of injecting water with a salinity lower than the water present in the reservoir causing a series of effects including the phenomena of detachment, Migration and blockage by fines that can modify the permeability of the rock and consequently alter the flow in the reservoir.

In the development of the model, a systematic modeling methodology was used and from the methodological point of view, each stage is described, consisting of conceptual, mathematical, érico and computacional.

A general mathematical model of flow and transport in porous medium was obtained, applying the axiomatic formulation of the continuous media modeling, where the flow model is slightly compressible monofasic, while the model of Multicomponent transport includes several relevant physicochemical phenomena such as advection, diffusion, dispersion, adsorption-desorption and reactions.

The system of partial differential equations was solved numerically in space by the finite element method using three different formulations: the classical method (CL), the mixed (MX) and the mixed-dual (MD). Whereas in time the method of finite differences was applied backwards, resulting in a completely implicit numerical scheme. The computational implementation was done in the Python 2.7 programming language using the FEniCS 2017.1 project, which is open source and multiplatform software (Windows / Linux).

The flow model was validated for a reference case and a comparison was made between the three finite element formulations in terms of the order of convergence and their performance, resulting in the optimization of the dual-mixed.

Finally, the coupled flow and transport model was applied in two cases of fine migration study. Numerical solutions obtained with the implementation in FEniCS 2017.1 are compared with those obtained with commercial software COMSOL Multiphysics®.

Índice general

Resumen	I
Abstract	III
Índice general	VIII
Lista de figuras	X
Lista de cuadros	XI
Introducción	XIII
1. Metodología de la modelación	1
1.1. Modelación conceptual	3
1.2. Modelación matemática	3
1.3. Modelación numérica	3
1.4. Modelación computacional	4
1.5. Validación	4
1.6. Casos de estudio	4
2. Introducción al Método de Elementos Finitos	5
2.1. Definición de elemento finito	6
2.2. Notación	7
2.3. Definición de malla	7
2.4. Elementos de Lagrange	8
2.5. Elementos de Brezzi-Douglas-Marini	8
2.6. Elementos discontinuos de Lagrange	9
2.7. Elementos de Raviart-Thomas	10
2.8. Ejemplo: la ecuación de Poisson	11
2.8.1. Formulación variacional clásica	12
2.8.2. Formulación variacional mixta	14
2.8.3. Formulación variacional mixta-dual	17
3. El proyecto FEniCS	21
3.1. Componentes	22
3.2. Proceso de solución con FEniCS	23
3.3. Ejemplo: la ecuación de Poisson	24

3.3.1.	Implementación con la forma variacional clásica	25
3.3.2.	Implementación con la forma variacional mixta	27
3.3.3.	Implementación con la forma variacional mixta-dual	29
4.	Modelo de flujo monofásico	33
4.1.	Modelo conceptual	34
4.2.	Modelo matemático	35
4.3.	Modelo numérico	37
4.3.1.	Modelo numérico con la formulación clásica	38
4.3.2.	Modelo numérico con la formulación mixta	40
4.3.3.	Modelo numérico con la formulación mixta-dual	42
4.4.	Modelo computacional	44
4.4.1.	Implementación con la formulación clásica	48
4.4.2.	Implementación con la formulación mixta	49
4.4.3.	Implementación con la formulación mixta-dual	50
4.5.	Resultados	52
4.6.	Análisis de las tasas de convergencia	56
5.	Modelo de transporte multicomponente	61
5.1.	Modelo conceptual	62
5.2.	Modelo matemático	62
5.3.	Modelo numérico	63
5.4.	Modelo computacional	66
6.	Modelo de flujo y transporte	71
6.1.	Modelo conceptual	72
6.2.	Modelo matemático	73
6.3.	Modelo numérico	73
6.4.	Modelo computacional	74
7.	Caso de estudio: Proceso LSWF	77
7.1.	Modelo conceptual	79
7.2.	Modelo matemático	80
7.2.1.	Ecuación de transporte de la salinidad C_s	80
7.2.2.	Ecuación de finos adheridos σ_a	81
7.2.3.	Ecuación de transporte de finos móviles C_f	81
7.2.4.	Ecuación de finos atorados σ_c	82
7.2.5.	Ecuación de flujo	82
7.2.6.	Ecuaciones de Estado	83
7.2.7.	Diagrama causal de la dinámica del sistema	84
7.3.	Modelo numérico	85
7.4.	Modelo computacional	86
7.4.1.	Espacios de funciones	86
7.4.2.	Funciones de base y de peso	86
7.4.3.	Fronteras naturales y esenciales	87
7.4.4.	Forma variacional	87

7.4.5.	Condiciones iniciales	87
7.4.6.	Solución en el tiempo	88
7.5.	Simulaciones	89
7.5.1.	Simulaciones de la salinidad del fluido	91
7.5.2.	Simulaciones de la concentración crítica de finos adheridos	93
7.5.3.	Simulaciones de la concentración de finos adheridos	94
7.5.4.	Simulaciones de la concentración de finos móviles	96
7.5.5.	Simulaciones de la concentración de finos atorados	98
7.5.6.	Simulaciones de la permeabilidad	100
7.5.7.	Simulaciones de la presión	102
7.5.8.	Simulaciones de la velocidad	104
7.5.9.	Simulaciones de la permeabilidad efectiva	106
7.6.	Análisis de resultados	106
Conclusiones		107
Apéndices		109
A. Proceso de inyección de agua de baja salinidad a escala de laboratorio		111
A.1.	Mecanismos del proceso LSWF	112
A.1.1.	Desprendimiento y migración de finos	112
A.1.2.	Contribución del pH	112
A.1.3.	Efecto Salting-in	113
A.1.4.	Intercambio iónico multicomponente	113
A.1.5.	Alteración de la mojabilidad	113
A.2.	Condiciones necesarias para el efecto LSWF	113
A.3.	Mecanismos presentes en el efecto LSWF	114
B. Modelación matemática de los sistemas continuos		117
B.1.	Teoría de los sistemas continuos	118
B.2.	Propiedades intensivas	119
B.2.1.	Representación Lagrangiana	119
B.2.2.	Representación Euleriana	120
B.2.3.	La derivada material	120
B.3.	Propiedades extensivas	121
B.4.	Ecuación de balance global	121
B.5.	Condiciones de balance local	122
B.5.1.	Ecuación diferencial de balance local	123
B.5.2.	Condición de salto de balance local	123
B.6.	Teoremas	124
B.6.1.	Teorema de Gauss extendido	124
B.6.2.	Teorema de transporte de Reynolds extendido	124
B.6.3.	Lema de Dubois-Reymond extendido	124
B.7.	De la ecuación de balance global a las ecuaciones de balance local	125
B.8.	Ejemplo	125
B.9.	Ecuaciones básicas de la MMC	127

B.9.1. Ecuaciones de balance de masa	127
B.9.2. Ecuaciones de balance de momento lineal	127
B.9.3. Ecuaciones de balance de energía	129
B.10. Modelos de fases múltiples	131
B.11. Modelos completos	133
B.11.1. Condiciones iniciales	134
B.11.2. Condiciones de frontera	134
Bibliografía	137

Lista de figuras

1.1. Proceso de la metodología de la modelación.	2
2.1. Ejemplos de celdas de elementos finitos	7
2.2. Elemento de Lagrange CG_q en un triángulo para $q = 1, 2, 3, 4, 5, 6$	8
2.3. Elemento de Lagrange CG_q en un tetraedro para $q = 1, 2, 3, 4, 5, 6$	9
2.4. Elementos Brezzi-Douglas-Marini de primero, segundo y tercer grado sobre triángulos y tetraedros.	9
2.5. Elementos de Lagrange discontinuos de cero, primero, segundo y tercer grado sobre triángulos y tetraedros.	10
2.6. Elemento de Lagrange discontinuo cuadrático. Todos los grados de libertad de un elementos finito de Lagrange discontinuo son internos al elemento, lo que significa que no se impone ninguna continuidad global por estos elementos.	10
2.7. Elementos Raviart-Thomas de primero, segundo y tercer grado sobre triángulos y tetraedros.	11
3.1. Comunicación entre los diversos componentes de FEniCS. Las líneas continuas indican las dependencias y las líneas discontinuas indican el flujo de datos.	24
3.2. Solución de la ecuación de Poisson mediante la formulación clásica y polinomios de Lagrange de orden 2 obtenida con FEniCS	26
3.3. Solución de la ecuación de Poisson mediante la formulación clásica y polinomios de Lagrange de orden 4 obtenida con FEniCS	26
3.4. Distribución sobre el cuadrado unitario de la solución numérica aproximada de la velocidad y presión en la ecuación de Poisson mediante la forma variacional mixta, obtenida con FEniCS, sobre una malla triangular uniforme de 16×16	29
3.5. Solución de la ecuación de elíptica con el MEF Mixto-Dual	31
4.1. Dominio del modelo de flujo monofásico	38
4.2. Mallado del modelo de flujo monofásico.	46
4.3. Presión obtenida con COMSOL Multiphysics del modelo de flujo monofásico.	52
4.4. Perfiles de Presión del Flujo Monofásico	53
4.5. Perfiles de la velocidad del flujo monofásico	54
4.6. Cambio de la presión obtenida con COMSOL Multiphysics y FEniCS del modelo de flujo monofásico.	55
4.7.	58
4.8.	58
4.9. Análisis del error con el MEF Mixto-Dual	59

7.1.	Proceso de desprendimiento y migración de finos	78
7.2.	Núcleo vertical de longitud L y radio r	80
7.3.	Concentración crítica de finos adheridos	83
7.4.	Diagrama causal del caso de estudio	84
7.5.	Malla generada con COMSOL en 3D para el caso de estudio.	90
7.6.	Malla generada con FEniCS para el caso de estudio.	90
7.7.	Salinidad del fluido en la entrada del núcleo vs. tiempo.	91
7.8.	Salinidad del fluido en la salida del núcleo vs. tiempo.	91
7.9.	Perfiles de la salinidad del fluido con FEniCS	92
7.10.	Perfiles de la salinidad del fluido con COMSOL	92
7.11.	Concentración crítica de finos adheridos vs. salinidad	93
7.12.	Concentración crítica de finos adheridos vs. tiempo.	93
7.13.	Concentración de finos adheridos en la entrada del núcleo vs. tiempo.	94
7.14.	Concentración de finos adheridos en la salida del núcleo vs. tiempo.	94
7.15.	Perfiles de la conc. de finos adheridos con FEniCS	95
7.16.	Perfiles de la conc. de finos adheridos con COMSOL	95
7.17.	Concentración de finos móviles en la entrada del núcleo vs. tiempo.	96
7.18.	Concentración de finos móviles en la salida del núcleo vs. tiempo.	96
7.19.	Perfiles de la conc. de finos móviles con FEniCS	97
7.20.	Perfiles de la conc. de finos móviles con COMSOL	97
7.21.	Conc. de finos atorados en la entrada del núcleo vs. tiempo.	98
7.22.	Conc. de finos atorados en la salida del núcleo vs. tiempo.	98
7.23.	Perfiles de la conc. de finos atorados con FEniCS	99
7.24.	Perfiles de la conc. de finos atorados con COMSOL	99
7.25.	Permeabilidad en la entrada del núcleo vs. tiempo.	100
7.26.	Permeabilidad en la salida del núcleo vs. tiempo.	100
7.27.	Perfiles de la permeabilidad con FEniCS	101
7.28.	Perfiles de la permeabilidad con COMSOL	101
7.29.	Presión en la entrada del núcleo vs. tiempo.	102
7.30.	Caída de presión en el núcleo vs. tiempo.	102
7.31.	Perfiles de Presión con FEniCS	103
7.32.	Perfiles de Presión con COMSOL	103
7.33.	Velocidad del fluido en la entrada del núcleo vs. tiempo.	104
7.34.	Velocidad del fluido en la salida del núcleo vs. tiempo.	104
7.35.	Perfiles de la velocidad con FEniCS	105
7.36.	Perfiles de la velocidad con COMSOL	105
7.37.	Permeabilidad efectiva en la entrada del núcleo vs. tiempo.	106
B.1.	Esquema de un cuerpo material $B(t)$ en movimiento	119
B.2.	Esquema de un cuerpo material $B(t)$	122

Lista de cuadros

2.1. Elementos finitos utilizados	6
2.2. Notación esquemática utilizada para los grados de libertad.	7
4.1. Propiedades del núcleo del modelo de flujo monofásico	44
4.2. Análisis de la tasa de convergencia de la presión con el MEF Clásico	57
7.1. Coeficientes de la ecuación de cada componente del sistema.	85
7.2. Características del núcleo, fluido y experimento.	89
7.3. Datos de la escalera de reducción de salinidad de la salmuera de inyección. .	89

Introducción

La motivación de la presente tesis fue desarrollar un modelo general de flujo monofásico y transporte multicomponente en medios porosos a escala de laboratorio con condiciones de frontera generales de tipo Neumann y Dirichlet, usando el método de elementos finitos mixtos e implementarlo computacionalmente en un script mediante el lenguaje de programación Python 2.7 usando FEniCS Project 2017.1 el cual es libre, de código abierto y multiplataforma (Linux, Windows, Mac), con el objetivo de simular numéricamente un proceso de desprendimiento, migración y bloqueo de poros por finos mediante la técnica de recuperación mejorada de hidrocarburos de inyección de agua de baja salinidad, la cual consiste en inyectar agua con una salinidad menor al agua presente en el yacimiento.

El sistema de ecuaciones diferenciales parciales del caso de estudio es no-lineal debido a que está acoplado dinámicamente, ya que al reducir lo suficiente la salinidad de la salmuera de inyección, provoca que los finos adheridos a la roca comiencen a desprenderse, generando finos móviles que migran con el fluido, los cuales dependiendo de su tamaño y de la distribución de los poros, se van quedando atorados en el trayecto, tapando los poros y bloqueando el flujo, por lo que se reduce la permeabilidad absoluta del medio y que a su vez se manifiesta en un aumento de la presión para mantener el gasto constante que se le impuso, lo que implica la velocidad advectiva, la cual es un ingrediente importante en la ecuación de advección-dispersión de la salinidad y de los finos móviles. Dicho sistema de ecuaciones se resolvió numéricamente en el espacio mediante el método de elementos finitos mixtos; mientras que en el tiempo se aplicó el método de diferencias finitas hacia atrás, resultando un esquema numérico completamente implícito.

En el **capítulo uno** se explica brevemente la metodología general para desarrollar sistemáticamente un modelo y cada una de sus etapas correspondientes, las cuales consisten en el modelo conceptual, matemático, numérico y computacional.

En el **capítulo dos** se hace una introducción al Método de Elementos Finitos con el cual se hace la aproximación numérica de los modelos matemáticos a desarrollar para que se puedan implementar en una plataforma computacional. Se describe el desarrollo de la formulación débil para la ecuación de Poisson por el Método de Elementos Finitos, clásico, mixto y mixto-dual.

En el **capítulo tres**, se describe el proyecto FEniCS, el cual es una colección de componentes de software libre de código abierto, con el objetivo común de permitir la solución automatizada de ecuaciones diferenciales. Los componentes proporcionan herramientas de computación científica para trabajar con mallas computacionales, formulaciones variacionales de elementos finitos de ecuaciones diferenciales ordinarias y parciales y álgebra lineal

numérica [18]. Se describe brevemente cada uno de sus módulos y su interacción. Se muestra como ejemplo, la forma de implementar la ecuación de Poisson en su formulación variacional clásica, mixta y mixta-dual.

En el **capítulo cuatro** se desarrolla un modelo de flujo monofásico ligeramente compresible y se muestra el desarrollo de las formulaciones variacionales con el método de elementos finitos clásico, mixto y mixto-dual, para discretizar el espacio y el método de Euler hacia atrás para discretizar el tiempo. Se hace un análisis de las tasas de las convergencias de los tres métodos de elementos finitos: clásico, mixto y mixto-dual.

En el **capítulo cinco** se desarrolla el modelo de transporte el cual mantiene una velocidad constante y se resuelve numéricamente en el espacio mediante el método de elementos finitos estándar, mientras que para para discretizar el tiempo se aplica el método de Euler hacia atrás.

En el **capítulo seis** se desarrolla el modelo de flujo y transporte y se resuelve numéricamente mediante el método de elementos finitos para discretizar el espacio, donde se derivan la formulación variacional mixta-dual para la ecuación de flujo y las formulaciones variacionales estándar de las ecuaciones de transporte, mientras que para para discretizar el tiempo se aplica el método de Euler hacia atrás. Finalmente se resuelven simultáneamente en un sistema de ecuaciones no-lineales.

En el **capítulo siete** se realiza un caso de estudio para la simulación del proceso de recuperación mejorada de hidrocarburos a escala de laboratorio, mediante la técnica de inyección de agua de baja salinidad (LSWF por sus siglas en inglés) y se realizan las simulaciones numéricas en dos dimensiones de un problema de migración de finos usando datos tomados de la literatura y se compararon los resultados obtenidos con los publicados y obtenidos con COMSOL Multiphysics®.

Se demostró que la metodología de la modelación matemática y computacional es una metodología adecuada y apropiada para desarrollar modelos. Por otro lado, se implementó un modelo de flujo y transporte usando la formulación mixta-dual del método de elementos finitos para simular pruebas de laboratorio a escala de núcleo desarrollando un software con el lenguaje de programación Python basado en FEniCS Project 2017.1. Este modelo se aplicó exitosamente para el problema de migración de finos en el caso de estudio de inyección de agua de baja salinidad usando datos tomados de la literatura.

Como trabajo futuro, se desea desarrollar modelos multifásicos y multicomponentes de flujo y transporte multiescala, con un gran potencial en su aplicación a la modelación de yacimientos heterogéneos. Obtener un módulo de optimización para realizar el ajuste de datos, un módulo de conversión de unidades y una interfaz gráfica para los usuarios.

Capítulo 1

Metodología de la modelación

En el presente capítulo se explica brevemente la metodología general para desarrollar sistemáticamente un modelo y cada una de sus etapas correspondientes.

El medio más efectivo para predecir el comportamiento de un sistema de la naturaleza y anticiparse a los acontecimientos es la metodología de la modelación, la cual consiste en construir y desarrollar modelos de los sistemas de interés. Un modelo de un sistema es un sustituto del sistema original cuyo comportamiento imita y reproduce, con cierto grado de aproximación, el correspondiente comportamiento del sistema original.

La manera en que se construyeron los modelos en esta tesis, fue siguiendo *la metodología de la modelación matemática y computacional*, la cual es una metodología conveniente para modelar y que formalmente consta de cuatro etapas: la modelación conceptual, matemática, numérica y computacional, como se puede observar en la figura (1.1).

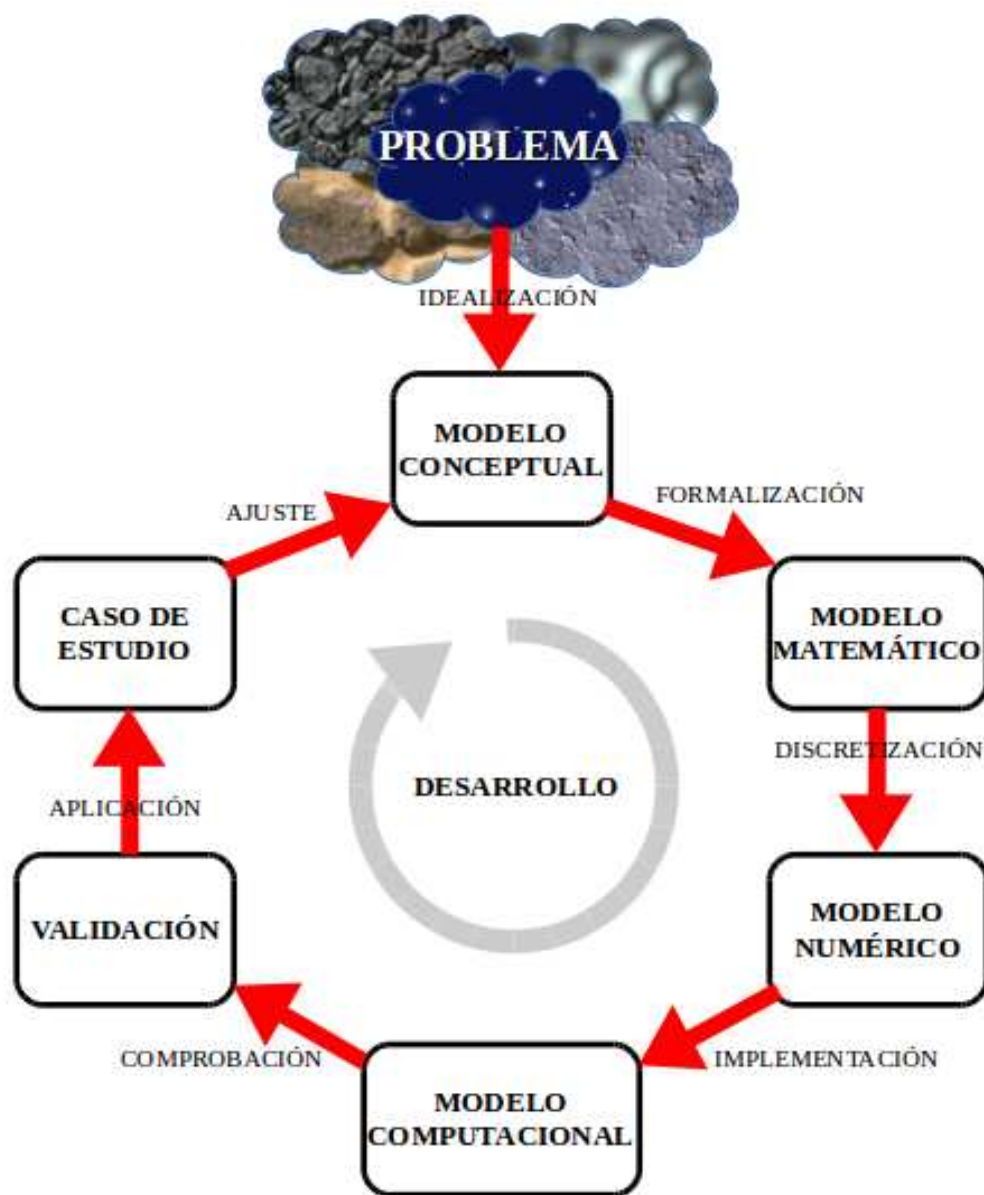


Figura 1.1: Proceso de la metodología de la modelación.

1.1. Modelación conceptual

El primer paso en el proceso de la modelación es la construcción de un *modelo conceptual* del sistema y su comportamiento. En el *modelo conceptual* del problema a resolver, se establecen los supuestos, los alcances y las limitaciones del modelo. Este paso incluye una descripción verbal de la composición del sistema, los fenómenos físicos involucrados, los mecanismos que lo gobiernan y las propiedades relevantes del medio en donde ocurren. La descripción toma la forma de una serie de suposiciones, seleccionadas subjetivamente por el modelador, para expresar en palabras su entendimiento y aproximación del sistema real y los procesos que ocurren dentro de éste, con el objetivo de proveer la información requerida. Por su cualidad de ser una versión subjetiva y simplificada del sistema real, no existen modelos únicos para un sistema poroso dado. Distintos conjuntos de suposiciones simplificadas, cada uno adecuado para obtener la información requerida, resultará en un modelo distinto.

1.2. Modelación matemática

La etapa de *la modelación matemática* consiste en obtener una representación matemática del comportamiento del sistema a estudiar, lo cual implica una descripción satisfactoria de cada una de las hipótesis del modelo conceptual en forma de relaciones matemáticas.

En el presente trabajo, para desarrollar los modelos matemáticos se utilizó el enfoque de la formulación axiomática para la modelación de los sistemas de medios continuos, explicado con más detalle en el apéndice B, donde se describe el comportamiento de las propiedades intensivas en correspondencia con las propiedades extensivas en las ecuaciones de balance global y después mediante ecuaciones de balance local, se desarrolla el conjunto de ecuaciones diferenciales parciales. Esto se realiza para cada componente en cada fase, posteriormente, se especifican leyes constitutivas que estén ligadas con la naturaleza del problema. Finalmente, el modelo se completa especificando las condiciones iniciales y de frontera para que resulte en un problema bien planteado y tenga solución única.

1.3. Modelación numérica

El *modelo numérico* es una versión discretizada del modelo matemático, cuyas soluciones son aproximadas hasta cierto nivel de error pero que a su vez deben de ser consistentes con la solución del modelo matemático, porque salvo para los problemas más sencillos, no es posible obtener por métodos analíticos las soluciones de tales ecuaciones, que son las que permiten predecir el comportamiento de los sistemas. En cambio, la diversidad y complejidad de problemas que pueden ser tratados con métodos numéricos es impresionante.

En el presente trabajo, las ecuaciones diferenciales parciales se resolvieron numéricamente en el espacio mediante el método de elementos finitos usando tres diferentes formulaciones: el método clásico, el mixto y el mixto-dual. Mientras que en el tiempo se aplicó el método de diferencias finitas hacia atrás, resultando un esquema numérico completamente implícito. Los tres esquemas de los elementos finitos se detallan más en el capítulo 2.

1.4. Modelación computacional

En la cuarta etapa se realiza la *modelación computacional* donde se traduce el modelo numérico a cierto lenguaje de programación para ser implementado en un sistema de cómputo específico. Son los resultados de éste, el producto final del proceso de la modelación. Las ventajas de este modelo frente a los modelos físicos es que se pueden realizar distintas simulaciones únicamente variando los parámetros de entrada del modelo fácilmente. En la actualidad, la simulación computacional permite estudiar sistemas complejos como fenómenos naturales, lo que sería muy costoso, peligroso o incluso imposible de estudiar por experimentación directa.

En el presente trabajo, la implementación computacional se realizó mediante el lenguaje de programación Python 2.7 utilizando el Proyecto FEniCS 2017.1, que es una colección de bibliotecas libres de código abierto y multiplataforma (Windows/Linux), para la resolución automatizada de ecuaciones diferenciales por el método de elementos finitos. Se puede encontrar información más detallada sobre el Proyecto FEniCS en el capítulo 3.

1.5. Validación

La *validación* del modelo es la etapa donde se corrobora que los resultados obtenidos con el modelo tienen una buena aproximación del fenómeno en cuestión que se desea describir, haciendo comparaciones entre las soluciones obtenidas por el modelo computacional y las soluciones analíticas del modelo matemático realizadas con ciertas consideraciones, o también, con resultados publicados de pruebas de laboratorio.

1.6. Casos de estudio

Los *casos de estudio* se desarrollan para estudiar y analizar el fenómeno específico de interés que se desea reproducir, donde se proponen datos de entrada particulares con los cuales se resolverá el modelo propuesto para poder predecir lo más aproximado posible el comportamiento del sistema bajo las condiciones supuestas.

Capítulo 2

Introducción al Método de Elementos Finitos

En el presente capítulo se hace una introducción al Método de Elementos Finitos con el cual se hace la aproximación numérica de los modelos matemáticos a desarrollar para que se puedan implementar en una plataforma computacional. Se describe el desarrollo de la formulación débil para la ecuación de Poisson por el Método de Elementos Finitos, clásico, mixto y mixto-dual.

El método de elementos finitos es actualmente uno de los más populares y poderosos para resolver problemas diferenciales con dominios de forma compleja [27]. Fue desarrollado a partir de 1950 [9], por ingenieros y matemáticos como un método general para la solución numérica de ecuaciones diferenciales parciales en su forma variacional, las cuales caracterizan el comportamiento de un fenómeno físico sobre un medio continuo, dividiendo el dominio en subdominios no intersectantes entre sí denominados *elementos finitos*. El conjunto de elementos finitos forma una partición del dominio, también denominada *discretización*. Dentro de cada elemento se distinguen una serie de puntos representativos llamados *nodos* y al conjunto de nodos considerando su adyacencia se le denomina *mall*a. El conjunto de relaciones entre el valor de una determinada variable en los nodos, se puede escribir como un sistema de ecuaciones lineales (o linealizadas). El número de ecuaciones de dicho sistema es proporcional al número de nodos. La solución obtenida es sólo aproximada, coincidiendo con la solución exacta en los nodos, mientras que en el resto de puntos que no son nodos, la solución aproximada se obtiene interpolando los resultados obtenidos en los nodos.

2.1. Definición de elemento finito

Una vez dividido el dominio Ω en celdas, se puede definir un espacio de funciones locales \mathcal{V} en cada celda T y utilizar estos espacios de funciones locales para construir el espacio de funciones global V_h . Una celda T junto con un espacio de funciones local \mathcal{V} y un conjunto de reglas para describir las funciones en \mathcal{V} es llamado *elemento finito*. Esta definición fue formalizada por primera vez por Ciarlet [10] y permanece hasta hoy la formulación estándar [6]. La definición formal dice lo siguiente: un elemento finito está definido por una tripleta $(T, \mathcal{V}, \mathcal{L})$, donde

- El dominio T es un subconjunto cerrado y acotado de \mathbb{R}^d (para $d = 1, 2, 3, \dots$) con interior no vacío y suave a tramos en la frontera.
- El espacio $\mathcal{V} = \mathcal{V}(T)$ es un espacio de funciones sobre T de dimensión finita n .
- El conjunto de los grados de libertad (nodos) $\mathcal{L} = \{l_1, l_2, \dots, l_n\}$ es una base para el espacio dual \mathcal{V}' ; es decir, el espacio de funcionales lineales acotados sobre \mathcal{V} .

Los elementos que se utilizan en la presente tesis se encuentran en la tabla (2.1)

Elemento Finito	Abreviatura	Espacio de Sóvlev	Conforme
Lagrange	CG	H^1	si
Discontinuous Lagrange	DG	L^2	si
Brezzi-Douglas-Marini	BDM	$H(div)$	si
Raviart-Thomas	RT	$H(div)$	si







Cuadro 2.1: Tipos de elementos finitos utilizados en la presente tesis, incluyendo su nombre completo, abreviatura y el respectivo espacio de Sobolev en que los elementos son conformes o no-conformes.

2.2. Notación

El espacio de polinomios de grado superior q sobre un dominio $T \subset \mathbb{R}^d$ está denotado por $\mathcal{P}_q(T)$ y los correspondientes campos del d -vector por $[\mathcal{P}_q(T)]^d$.

Un espacio de elementos finitos E es llamado *V-conforme* si $E \subseteq V$. Si no, este es llamado *(V-) no conforme*.

Los elementos de \mathcal{L} se denominan generalmente como los *grados de libertad* del elemento $(T, \mathcal{V}, \mathcal{L})$. Cuando se describen familias de elementos finitos, usualmente se ilustran los grados de libertad con una cierta notación esquemática como se puede apreciar en la tabla (2.2)

	Evaluación en un punto.
	Evaluación de todas las primeras derivadas.
	Evaluación de todas las segundas derivadas.
	Evaluación de la componente direccional.
	Evaluación de la derivada direccional.
	Evaluación de los momentos interiores.

Cuadro 2.2: Notación esquemática utilizada para los grados de libertad.

2.3. Definición de malla

Para definir V_h , primero se divide el dominio Ω en un conjunto finito de celdas $\mathcal{T}_h = \{T\}$ con interiores disjuntos tales que

$$\bigcup_{T \in \mathcal{T}_h} T = \Omega \quad (2.1)$$

Juntas, estas celdas forman una *malla* del dominio Ω . Las celdas típicamente son formas poligonales simples como intervalos, triángulos, cuadriláteros, tetraedros o hexaedros como se muestra en la figura (2.1), pero también son posibles otras formas, en particular las celdas curvadas para capturar correctamente la frontera de un dominio no-poligonal.

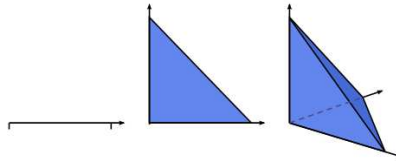


Figura 2.1: Ejemplos de celdas de elementos finitos en una, dos y tres dimensiones.

2.4. Elementos de Lagrange

El elemento finito más conocido y más utilizado es el elemento de Lagrange P_1 , sin embargo hay toda una familia de elementos parametrizados por el grado polinomial. Los elementos de Lagrange de grado superior ofrecen mejores propiedades de orden de aproximación [25].

El elemento de Lagrange (CG_q) está definido para $q = 1, 2, \dots$ por:

$$T \in \{\text{intervalo}, \text{triángulo}, \text{tetraedro}\} \quad (2.2)$$

$$\mathcal{V} = \mathcal{P}_q(T) \quad (2.3)$$

$$l_i(v) = v(x^i), \quad i = 1, \dots, n(q) \quad (2.4)$$

donde $\{x^i\}_{i=1}^{n(q)}$ es una enumeración de puntos en T definida por

$$x = \begin{cases} i/q & 0 \leq i \leq q, & \text{intervalo } T \\ (i/q, j/q) & 0 \leq i+j \leq q, & \text{triángulo } T \\ (i/q, j/q, k/q) & 0 \leq i+j+k \leq q, & \text{tetraedro } T \end{cases} \quad (2.5)$$

La dimensión de los elementos finitos de Lagrange corresponde a la dimensión de los polinomios completos de grado q sobre T y es:

$$n(q) = \begin{cases} q+1, & \text{intervalo } T \\ \frac{1}{2}(q+1)(q+2), & \text{triángulo } T \\ \frac{1}{6}(q+1)(q+2)(q+3), & \text{tetraedro } T \end{cases} \quad (2.6)$$

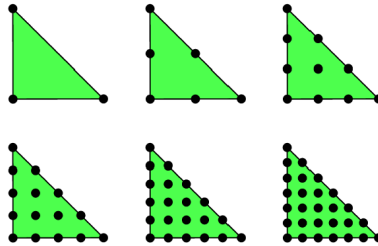


Figura 2.2: Elemento de Lagrange CG_q en un triángulo para $q = 1, 2, 3, 4, 5, 6$.

2.5. Elementos de Brezzi-Douglas-Marini

Los elementos *Brezzi-Douglas-Marini* (BDM_q) fueron introducidos por [7] para formulaciones mixtas de ecuaciones elípticas de segundo orden y están definidos para $q = 1, 2, \dots$ por

$$T \in \{\text{triángulo}, \text{tetraedro}\} \quad (2.7)$$

$$\mathcal{V} = [\mathcal{P}_q(T)]^d \quad (2.8)$$

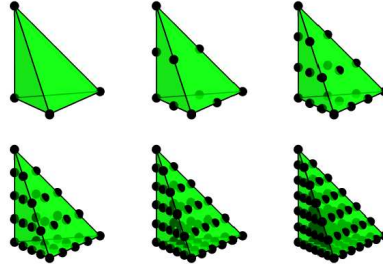


Figura 2.3: Elemento de Lagrange CG_q en un tetraedro para $q = 1, 2, 3, 4, 5, 6$.

$$\mathcal{L} = \begin{cases} \int_f \underline{v} \cdot \underline{n} p ds, & \text{para un conjunto de func. base } p \in \mathcal{P}_q(f) \text{ para cada faceta } f \\ \int_T \underline{v} \cdot \underline{p} d\mathbf{x}, & \text{para un conjunto de func. base } p \in NED_{q-1}^1(T) \text{ para } q \geq 2 \end{cases} \quad (2.9)$$

donde NED_{q-1}^1 se refiere a los elementos de Nédélec $H(\text{curl})$ de primer orden.

La dimensión de los elementos BDM_q es:

$$n(q) = \begin{cases} (q+1)(q+2), & \text{triángulo } T \\ \frac{1}{2}(q+1)(q+2)(q+3), & \text{tetraedro } T \end{cases} \quad (2.10)$$

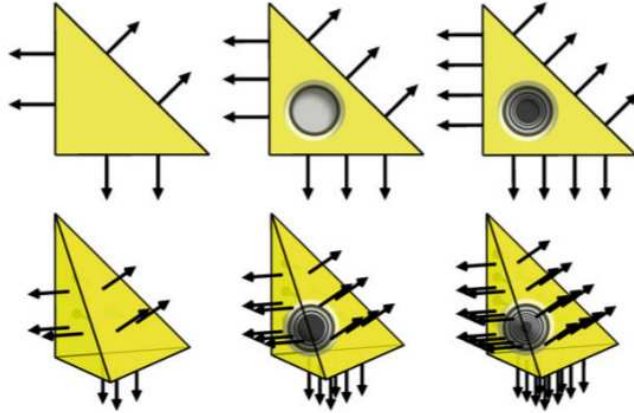


Figura 2.4: Elementos Brezzi-Douglas-Marini de primero, segundo y tercer grado sobre triángulos y tetraedros.

2.6. Elementos discontinuos de Lagrange

Los elementos *Lagrange discontinuos* (DG_q) están definidos para $q = 0, 1, 2, \dots$ por

$$T \in \{\text{intervalo}, \text{triángulo}, \text{tetraedro}\} \quad (2.11)$$

$$\mathcal{V} = \mathcal{P}_q(T) \quad (2.12)$$

$$l_i(v) = v(x^i) \quad (2.13)$$

donde $\{x^i\}_{i=1}^{n(q)}$ es una enumeración de puntos en T definida por

$$x = \begin{cases} i/q & 0 \leq i \leq q, & \text{intervalo } T \\ (i/q, j/q) & 0 \leq i + j \leq q, & \text{triángulo } T \\ (i/q, j/q, k/q) & 0 \leq i + j + k \leq q, & \text{tetraedro } T \end{cases} \quad (2.14)$$

La dimensión de los elementos DG_q es:

$$n(q) = \begin{cases} q + 1, & \text{intervalo } T \\ \frac{1}{2}(q + 1)(q + 2), & \text{triángulo } T \\ \frac{1}{6}(q + 1)(q + 2)(q + 3), & \text{tetraedro } T \end{cases} \quad (2.15)$$

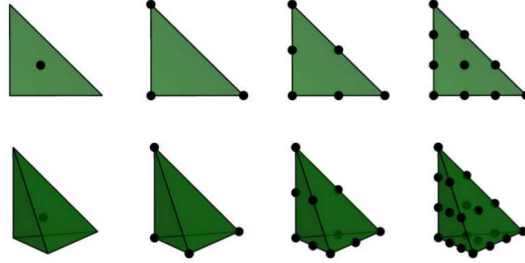


Figura 2.5: Elementos de Lagrange discontinuos de cero, primero, segundo y tercer grado sobre triángulos y tetraedros.

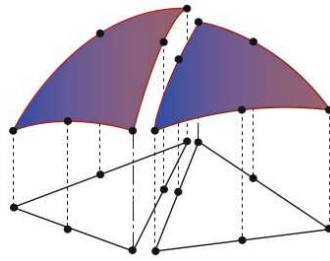


Figura 2.6: Elemento de Lagrange discontinuo cuadrático. Todos los grados de libertad de un elementos finito de Lagrange discontinuo son internos al elemento, lo que significa que no se impone ninguna continuidad global por estos elementos.

2.7. Elementos de Raviart-Thomas

Los elementos de *Raviart-Thomas* fueron introducidos por Raviart y Thomas en 1977 [23] y fueron los primeros elementos para discretizar la forma mixta de ecuaciones elípticas

de primero y segundo orden sobre triángulos. Su espacio de elementos \mathcal{V} está diseñado de modo que sea el espacio polinomial más pequeño $\mathcal{V} \subset \mathcal{P}_q(T)$, para $q = 1, 2, \dots$ a partir de la cual la divergencia se mapea con $\mathcal{P}_{q-1}(T)$. La siguiente definición se basa en una presentada por Nédélec [21] y en ocasiones este es referido como el elemento de *Raviart-Thomas-Nédélec*.

El elemento de *Raviart-Thomas* (RT_q) está definido para $q = 1, 2, \dots$ por

$$T \in \{\text{triángulo}, \text{tetraedro}\} \quad (2.16)$$

$$\mathcal{V} = [\mathcal{P}_{q-1}(T)]^d + x\mathcal{P}_{q-1}(T) \quad (2.17)$$

$$\mathcal{L} = \begin{cases} \int_f \underline{v} \cdot \underline{n} p ds, & \text{para un conjunto de func. base } p \in \mathcal{P}_{q-1}(f) \text{ para cada faceta } f \\ \int_T \underline{v} \cdot \underline{p} d\underline{x}, & \text{para un conjunto de func. base } p \in [\mathcal{P}_{q-2}(T)]^d \text{ para } q \geq 2 \end{cases} \quad (2.18)$$

La dimensión de los elementos RT_q es:

$$n(q) = \begin{cases} q(q+2), & \text{triángulo } T \\ \frac{1}{2}q(q+1)(q+3), & \text{tetraedro } T \end{cases} \quad (2.19)$$

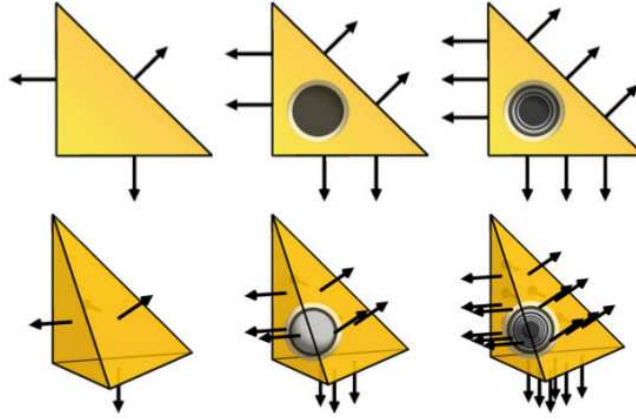


Figura 2.7: Elementos Raviart-Thomas de primero, segundo y tercer grado sobre triángulos y tetraedros.

2.8. Ejemplo: la ecuación de Poisson

El objetivo de esta sección es mostrar cómo obtener la forma variacional de la ecuación de Poisson (2.20), con condiciones de frontera generales del tipo Dirichlet y Neumann, mediante tres variantes del método de elementos finitos: Clásica, Mixta y Mixta-Dual.

$$-\nabla \cdot (\nabla p) = f, \quad \forall \underline{x} \in \Omega \quad (2.20)$$

$$p(\underline{x}) = p_\partial(\underline{x}), \quad \forall \underline{x} \in \partial\Omega_D \quad (2.21)$$

$$-(\nabla p) \cdot \underline{n} = g_\partial(\underline{x}), \quad \forall \underline{x} \in \partial\Omega_N \quad (2.22)$$

Aquí, $p = p(\underline{x})$ es la función desconocida, $f = f(\underline{x})$ es una función dada, Ω es el dominio espacial, $p_\partial = p_\partial(\underline{x})$ es un valor (que también puede ser una función) impuesto sobre la frontera de Ω de tipo Dirichlet $\partial\Omega_D$ donde la solución p es conocida, \underline{n} es el vector normal a la frontera de $\partial\Omega$ y $g_\partial = g_\partial(\underline{x})$ es un valor (que también puede ser una función) impuesto sobre la frontera de Ω de tipo Neumann $\partial\Omega_N$, donde $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$ y $\partial\Omega_D \cap \partial\Omega_N = \emptyset$.

2.8.1. Formulación variacional clásica

Sea $p \in W = \{p \in H^1(\Omega) : p(\underline{x}) = p_\partial(\underline{x}), \forall \underline{x} \in \partial\Omega_D\}$ la función incógnita a ser aproximada llamada *función de prueba* donde:

$H^1(\Omega) = \{w \in L^2(\Omega) : \nabla w \in L^2(\Omega)\}$ es el espacio de Sobolev

$L^2(\Omega) = \{w : \int_\Omega ||w||^2 d\underline{x} < \infty\}$ es un espacio vectorial normado.

El primer paso para obtener la formulación variacional clásica es multiplicar la ecuación (2.20) por una función $w \in \hat{W} = \{w \in H^1(\Omega) : w(\underline{x}) = 0, \forall \underline{x} \in \partial\Omega_D\}$, conocida como *función de peso* e integrar la ecuación resultante sobre el dominio Ω .

$$-\int_\Omega (\nabla \cdot (\nabla p)) w d\underline{x} = \int_\Omega f w d\underline{x} \quad (2.23)$$

Las funciones de prueba y de peso pertenecen a ciertos espacios W y \hat{W} llamados *espacios de funciones de prueba y de peso* respectivamente que especifican las propiedades de las funciones.

Una regla común cuando se deducen formulaciones variacionales es transformar una derivada espacial de segundo orden de p , en una primera derivada de p y w , aplicando la fórmula de integración por partes:

$$-\int_\Omega (\nabla \cdot (\nabla p)) w d\underline{x} = \int_\Omega \nabla p \cdot \nabla w d\underline{x} - \int_{\partial\Omega_N} w(\nabla p) \cdot \underline{n} d\underline{x} - \int_{\partial\Omega_D} w(\nabla p) \cdot \underline{n} d\underline{x} \quad (2.24)$$

donde $(\nabla p) \cdot \underline{n}$ es la derivada de p en la dirección del vector normal \underline{n} sobre la frontera $\partial\Omega$.

Sustituyendo la ecuación (2.24) en la ecuación (2.23) y reacomodando términos

$$\int_\Omega \nabla p \cdot \nabla w d\underline{x} = \int_\Omega f w d\underline{x} + \int_{\partial\Omega_N} w(\nabla p) \cdot \underline{n} d\underline{x} + \int_{\partial\Omega_D} w(\nabla p) \cdot \underline{n} d\underline{x} \quad (2.25)$$

Una característica de la formulación variacional clásica es que dado que $w \in \hat{W}$ entonces la función de peso $w = 0$ en las partes de la frontera donde la solución p es conocida, es decir, en toda la frontera de tipo Dirichlet $\partial\Omega_D$.

Por lo tanto, al aplicar las condiciones de frontera de Dirichlet (4.76) y Neumann (2.22), la integral sobre la frontera de Dirichlet de la ecuación (2.25) es igual a cero, obteniendo finalmente la *forma variacional clásica* del problema original (2.20)-(2.22):

$$\int_{\Omega} \nabla p \cdot \nabla w d\underline{x} = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_N} g w d\underline{x} \quad (2.26)$$

Para mantener esta ecuación para todas las funciones de peso w en un espacio adecuado \hat{W} , se obtiene un problema matemático bien definido que únicamente determina la solución p que se encuentra en algún espacio de funciones W posiblemente diferente.

La afirmación correcta del problema variacional clásico es: Encontrar $p \in W$ tal que

$$\int_{\Omega} \nabla p \cdot \nabla w d\underline{x} = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_N} g w d\underline{x}, \quad \forall w \in \hat{W} \quad (2.27)$$

El problema variacional (2.27) es un problema continuo, pues define la solución p en el espacio de funciones de dimensión infinita W .

Nótese que la condición de Dirichlet $p = p_{\partial}$ en $\partial\Omega_D$ entra directamente en la definición del espacio W (esta es una condición de frontera esencial), mientras que la condición de Neumann $-(\nabla p) \cdot \underline{n} = g_{\partial}$ en $\partial\Omega_N$, entra en el problema variacional (esta es una condición de frontera natural).

El método de elementos finitos clásico para la ecuación de Poisson encuentra una solución aproximada del problema variacional (2.27) reemplazando el espacio de funciones de dimensión infinita W y \hat{W} por espacios discretos de dimensión finita $W_h \subset W$ y $\hat{W}_h \subset \hat{W}$. La literatura en matemáticas sobre problemas variacionales utiliza la notación p_h para referirse a la solución del problema discreto, mientras que p se utiliza para referirse a la solución del problema continuo. Entonces el problema variacional discreto se lee como sigue:

Encontrar $p_h \in W_h \subset W$ tal que

$$\int_{\Omega} \nabla p_h \cdot \nabla w d\underline{x} = \int_{\Omega} f w d\underline{x} \quad \forall w \in \hat{W}_h \subset \hat{W} \quad (2.28)$$

Este problema variacional, junto con una adecuada definición de los espacios de funciones W_h y \hat{W}_h , definen la solución numérica aproximada de la ecuación de Poisson (2.20)-(2.22).

Introduciendo la siguiente notación canónica para los problemas variacionales discretos:

Encontrar $p_h \in W_h \subset W$ tal que

$$a(p_h, w) = L(w), \quad \forall w \in \hat{W}_h \subset \hat{W} \quad (2.29)$$

donde para la ecuación de Poisson, se tiene:

$$a(p_h, w) = \int_{\Omega} \nabla p_h \cdot \nabla w d\underline{x} \quad (2.30)$$

$$L(w) = \int_{\Omega} f w d\underline{x} \quad (2.31)$$

En la literatura matemática, $a(p_h, w)$ se conoce como la *forma bilineal*, $a : W_h \times \hat{W}_h \rightarrow \mathbb{R}$, mientras que $L(w)$ se conoce como la *forma lineal*, $L : \hat{W}_h \rightarrow \mathbb{R}$. En cada problema lineal que se resuelve, se deben identificar los términos con la incógnita p_h y recolectarse en $a(p_h, w)$, al igual que se recogen los términos con las funciones conocidas en $L(w)$.

Suponiendo por ahora que se tiene una base $\{\phi_j\}_{j=1}^N$ para W_h y una base $\{\hat{\phi}_i\}_{i=1}^N$ para \hat{W}_h . Aquí N denota la dimensión de los espacios W_h y \hat{W}_h . Para resolver el problema variacional discreto (2.29), se puede hacer una solución estimada p_h en términos de las funciones base

$$p_h(\underline{x}) = \sum_{j=1}^N P_j \phi_j(\underline{x}) \quad (2.32)$$

donde $P_j \in \mathbb{R}^N$ es el vector de grados de libertad a ser calculado. Sustituyendo la ecuación (2.32) en la formulación variacional discreta (2.29) y tomando $w = \hat{\phi}_i$ para $i = 1, 2, \dots, N$, se obtiene

$$\sum_{j=1}^N P_j \int_{\Omega} \nabla \phi_j \cdot \nabla \hat{\phi}_i d\underline{x} = \int_{\Omega} f \hat{\phi}_i d\underline{x} - \int_{\partial\Omega_N} g \hat{\phi}_i d\underline{x}, \quad \forall i = 1, 2, \dots, N. \quad (2.33)$$

de donde se sigue que

$$\sum_{j=1}^N P_j a(\phi_j, \hat{\phi}_i) = L(\hat{\phi}_i), \quad \forall i = 1, 2, \dots, N. \quad (2.34)$$

Los grados de libertad \underline{P} de la solución de elementos finitos p_h pueden calcularse resolviendo el sistema lineal de ecuaciones $\underline{A} \cdot \underline{P} = \underline{b}$, donde

$$A_{ij} = \int_{\Omega} \nabla \phi_j \cdot \nabla \hat{\phi}_i d\underline{x} = a(\phi_j, \hat{\phi}_i) \quad (2.35)$$

$$b_i = \int_{\Omega} f \hat{\phi}_i d\underline{x} - \int_{\partial\Omega_N} g \hat{\phi}_i d\underline{x} = L(\hat{\phi}_i) \quad (2.36)$$

con $i, j = 1, 2, \dots, N$.

2.8.2. Formulación variacional mixta

Ahora se ilustrará la aplicación del método de elementos finitos mixtos para ecuaciones diferenciales de segundo orden. Una razón, entre otras, para usar el método mixto es que en algunas aplicaciones, una variable vectorial (por ejemplo, la velocidad del fluido) es la variable primaria en la que uno está interesado. A continuación, el método mixto se desarrolla para aproximar a la vez esta variable y una variable escalar (por ejemplo, la presión) simultáneamente y dar una aproximación de orden alto de ambas variables. En lugar de un único espacio de elementos finitos utilizado en el método de elementos finitos clásico, el método de elementos finitos mixtos emplea dos espacios diferentes, lo que sugiere el nombre mixto. Estos dos espacios deben satisfacer una condición inf-sup para que el método mixto sea estable [8].

Para deducir la forma variacional mixta (con dos campos), se utiliza una formulación alternativa de la ecuación de Poisson (2.20), introduciendo un vector variable adicional, llamado flujo: $\underline{u} = -\nabla p$ como sigue

$$\underline{u} = -\nabla p \quad \forall \underline{x} \in \Omega \quad (2.37)$$

$$\nabla \cdot \underline{u} = f \quad \forall \underline{x} \in \Omega \quad (2.38)$$

con condiciones de frontera

$$p = p_{\partial}, \quad \forall \underline{x} \in \partial\Omega_D \quad (2.39)$$

$$\underline{u} \cdot \underline{n} = g_{\partial} \quad \forall \underline{x} \in \partial\Omega_N \quad (2.40)$$

Sean

$$p \in W = \{p \in H^1(\Omega) : p(\underline{x}) = p_{\partial}(\underline{x}), \forall \underline{x} \in \partial\Omega_D\}$$

$$\underline{u} \in \underline{V} = \{\underline{u} \in H(\text{div}, \Omega) : \underline{u}(\underline{x}) \cdot \underline{n}(\underline{x}) = g_{\partial}(\underline{x}), \forall \underline{x} \in \partial\Omega_N\}$$

$$\underline{v} \in \hat{\underline{V}} = \{\underline{v} \in H(\text{div}, \Omega) : \underline{v}(\underline{x}) \cdot \underline{n}(\underline{x}) = 0, \forall \underline{x} \in \partial\Omega_N\}$$

Multiplicando la ecuación (2.37) por una función vectorial de peso $\underline{v} \in \hat{\underline{V}}$ e integrando la ecuación resultante sobre el dominio Ω , se obtiene lo siguiente

$$\int_{\Omega} \underline{u} \cdot \underline{v} d\underline{x} = - \int_{\Omega} (\nabla p) \cdot \underline{v} d\underline{x} \quad (2.41)$$

integrando por partes la integral de la derecha de la igualdad en la ecuación (2.41)

$$\int_{\Omega} \underline{u} \cdot \underline{v} d\underline{x} = \int_{\Omega} p \nabla \cdot \underline{v} d\underline{x} - \int_{\partial\Omega_D} p \underline{v} \cdot \underline{n} ds - \int_{\partial\Omega_N} p \underline{v} \cdot \underline{n} ds \quad (2.42)$$

Una característica de la formulación variacional mixta es que dado que $\underline{v} \in \hat{\underline{V}}$ entonces la función vectorial de peso $\underline{v} = 0$ en las partes donde la velocidad \underline{u} es conocida, es decir, en toda la frontera de tipo Neumann.

Por lo tanto, al aplicar las condiciones de frontera de Dirichlet (2.39) y Neumann (2.40), la integral sobre la frontera de Neumann de la ecuación (2.42) es igual a cero, obteniendo

$$\int_{\Omega} \underline{u} \cdot \underline{v} d\underline{x} = \int_{\Omega} p \nabla \cdot \underline{v} d\underline{x} - \int_{\partial\Omega_D} p \underline{v} \cdot \underline{n} ds \quad (2.43)$$

Sea $\hat{W} = \{w \in H^1(\Omega) : w(\underline{x}) = 0, \forall \underline{x} \in \partial\Omega_D\}$. Multiplicando la ecuación (2.38) por una función de peso $w \in \hat{W}$ e integrando la ecuación resultante sobre el dominio Ω , se tiene que

$$\int_{\Omega} (\nabla \cdot \underline{u}) w d\underline{x} = \int_{\Omega} f w d\underline{x} \quad (2.44)$$

Sumando las ecuaciones (2.43) y (2.44), se obtiene la formulación variacional mixta:

Encontrar $(\underline{u}, p) \in M = \underline{V} \times W$ tal que

$$a((\underline{u}, p), (\underline{v}, w)) = L(\underline{v}, w), \quad \forall (\underline{v}, w) \in \hat{M} = \hat{\underline{V}} \times \hat{W} \quad (2.45)$$

donde la forma bilineal a y la forma lineal L , se definen de la siguiente manera

$$a((\underline{u}, p), (\underline{v}, w)) = \int_{\Omega} ((\nabla \cdot \underline{u})w + \underline{u} \cdot \underline{v} - p \nabla \cdot \underline{v}) d\underline{x} \quad (2.46)$$

$$L(\underline{v}, w) = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_D} p_{\partial} \underline{v} \cdot \underline{n} ds \quad (2.47)$$

Nótese que la condición de tipo Dirichlet $p = p_{\partial}$ en $\partial\Omega_D$ entra en la formulación variacional (esta es una condición de frontera natural) y la condición de tipo Neumann $\underline{u} \cdot \underline{n} = g_{\partial}$ en $\partial\Omega_N$ entra en la definición del espacio \underline{V} (esta es una condición de frontera esencial).

Se puede discretizar la formulación mixta, restringiendo el problema variacional (2.45) a un par de espacios mixtos discretos $M_h \subset M$ y $\hat{M}_h \subset \hat{M}$.

Encontrar $(\underline{u}_h, p_h) \in M_h = \underline{V}_h \times W_h$ tal que

$$a((\underline{u}_h, p_h), (\underline{v}, w)) = L(\underline{v}, w), \quad \forall (\underline{v}, w) \in \hat{M}_h = \hat{\underline{V}}_h \times \hat{W}_h \quad (2.48)$$

donde las forma bilineal a y la forma lineal L , se definen como

$$a((\underline{u}_h, p_h), (\underline{v}, w)) = \int_{\Omega} ((\nabla \cdot \underline{u}_h)w + \underline{u}_h \cdot \underline{v} - p_h \nabla \cdot \underline{v}) d\underline{x} \quad (2.49)$$

$$L(\underline{v}, w) = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_D} p_{\partial} \underline{v} \cdot \underline{n} ds \quad (2.50)$$

Para resolver el problema variacional mixto discreto (2.48), se va a suponer que se tiene una base $\{(\phi_j, \psi_j)\}_{j=1}^N$ del espacio M_h y una base $\{(\hat{\phi}_j, \hat{\psi}_j)\}_{i=1}^N$ del espacio \hat{M}_h . Aquí N denota la dimensión de los espacios M_h y \hat{M}_h .

Se puede hacer una estimación para la solución de elementos finitos mixtos de la forma

$$(\underline{u}_h, p_h) = \sum_{j=1}^N P_j(\phi_j, \psi_j) \quad (2.51)$$

resolviendo un sistema lineal $\underline{\underline{A}} \cdot \underline{\underline{P}} = \underline{\underline{b}}$, para los grados de libertad $\underline{\underline{P}} \in \mathbb{R}^N$, donde

$$A_{ij} = \int_{\Omega} [(\nabla \cdot \psi_j) \hat{\phi}_i + \psi_j \cdot \hat{\psi}_i - \phi_j \nabla \cdot \hat{\psi}_i] d\underline{x}$$

$$b_i = \int_{\Omega} f \hat{\phi}_i d\underline{x} - \int_{\partial\Omega_D} p_{\partial} \hat{\psi}_i \cdot \underline{n} ds$$

Las formulaciones mixtas requieren cierta atención en la selección de espacios que discretizan los diferentes espacios de funciones. Aquí, una combinación estable de espacios de elementos finitos mixtos es, para \underline{V}_h los elementos de *Brezzi-Douglas-Marini* (BDM) de orden polinomial k , mientras que para W_h se recomienda utilizar los elementos *Discontinuous Galerkin* (DG) de orden polinomial $k - 1$. [18]

Las discretizaciones estables de los métodos de elementos finitos mixtos deben satisfacer las condiciones denominadas “inf-sup” o de *Ladyszshenskaya-Babuska-Brezzi* (LBB) [4]. Esta teoría explica porqué muchos de los espacios de elementos finitos para los métodos mixtos parecen complicados en comparación con los de los elementos estándar.

Para que este problema tenga solución única, es natural imponer una condición discreta inf-sup como la siguiente [8]:

$$\sup_{0 \neq \underline{v} \in \underline{V}_h} \frac{|(\nabla \cdot \underline{v}, w)|}{\|\underline{v}\|_{\underline{V}}} \geq C_2 \|w\|, \quad \forall w \in W_h \quad (2.52)$$

donde $C_2 > 0$ es una constante independiente de h .

2.8.3. Formulación variacional mixta-dual

Sea W un espacio de Banach. Un mapeo $L : W \rightarrow \mathbb{R}$ es llamado *funcional lineal* si

$$L(\alpha x + \beta y) = \alpha L(x) + \beta L(y), \quad \alpha, \beta \in \mathbb{R}, \quad x, y \in W. \quad (2.53)$$

Se dice que L es acotado en la norma $\|\cdot\|_W$ si existe una constante $\tilde{L} > 0$ tal que

$$|L(w)| \leq \tilde{L} \|w\|_W, \quad \forall w \in W. \quad (2.54)$$

El conjunto de funcionales lineales acotados en W se denomina espacio dual de W y es denotado por W' [8].

Para $L \in W'$, se define

$$\|L\|_{W'} = \sup_{0 \neq w \in W} \frac{L(w)}{\|w\|_W} \quad (2.55)$$

Puesto que L es acotado, siempre es finita y de hecho induce una norma sobre W , llamada la *norma dual* y W' es un espacio de Banach con respecto a esta.

Una formulación mixta-dual de la ecuación de Poisson 2.20, implica la introducción de un vector variable adicional, llamado flujo: $\underline{u} = -\nabla p$ y reescribiéndola como sigue:

$$\underline{u} = -\nabla p \quad \forall \underline{x} \in \Omega \quad (2.56)$$

$$\nabla \cdot \underline{u} = f \quad \forall \underline{x} \in \Omega \quad (2.57)$$

con sus condiciones de frontera

$$p = p_\partial, \quad \forall \underline{x} \in \partial\Omega_D \quad (2.58)$$

$$\underline{u} \cdot \underline{n} = g_\partial, \quad \forall \underline{x} \in \partial\Omega_N \quad (2.59)$$

Sean

$$\underline{u} \in \underline{V} = \{\underline{u} \in H(\text{div}, \Omega) : \underline{u}(\underline{x}) \cdot \underline{n}(\underline{x}) = g_\partial, \quad \forall \underline{x} \in \partial\Omega_N\}$$

$$\underline{v} \in \hat{\underline{V}} = \{\underline{v} \in H(\text{div}, \Omega) : \underline{v}(\underline{x}) \cdot \underline{n}(\underline{x}) = 0, \quad \forall \underline{x} \in \partial\Omega_N\}$$

$$p \in W = \{p \in H^1(\Omega) : p(\underline{x}) = p_\partial(\underline{x}), \forall \underline{x} \in \partial\Omega_D\}$$

Multiplicando la ecuación (2.56) por una función vectorial de peso $\underline{v} \in \hat{\underline{V}}$ e integrando la ecuación resultante sobre el dominio Ω y reacomodando términos, se tiene que

$$\int_{\Omega} (\underline{u} \cdot \underline{v} + \nabla p \cdot \underline{v}) d\underline{x} = 0 \quad (2.60)$$

Sea $w \in \hat{W} = \{w \in H^1(\Omega) : w(\underline{x}) = 0, \forall \underline{x} \in \partial\Omega_D\}$

Multiplicando la ecuación (2.57) por una función de peso $w \in \hat{W}$ e integrando la ecuación resultante sobre el dominio Ω , se tiene que

$$\int_{\Omega} (\nabla \cdot \underline{u}) w d\underline{x} = \int_{\Omega} f w d\underline{x} \quad (2.61)$$

integrando por partes en la ecuación (2.61), la integral de la izquierda de la igualdad se tiene

$$\int_{\partial\Omega_D} w \underline{u} \cdot \underline{n} ds + \int_{\partial\Omega_N} w \underline{u} \cdot \underline{n} ds - \int_{\Omega} \underline{u} \cdot \nabla w d\underline{x} = \int_{\Omega} f w d\underline{x} \quad (2.62)$$

Una característica de la formulación variacional mixta-dual es que dado que $w \in \hat{W}$, entonces la función de peso $w = 0$ en las partes donde la presión p es conocida, es decir, en toda la frontera de tipo Dirichlet y en consecuencia, la integral sobre la frontera de Dirichlet en la ecuación (2.62) es igual a cero; mientras que por otro lado, como $\underline{u} \in \underline{V}$, entonces $\underline{u} \cdot \underline{n} = g_\partial$ en toda la frontera de tipo Neumann. Por lo tanto, al aplicar las condiciones de frontera de Dirichlet (2.58) y Neumann (2.59) en la ecuación (2.62) y reacomodando términos, se obtiene

$$- \int_{\Omega} \underline{u} \cdot \nabla w d\underline{x} = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_N} g w ds \quad (2.63)$$

Sumando las ecuaciones (2.60) y (2.63), se obtiene la formulación variacional mixta-dual:

Encontrar $(\underline{u}, p) \in M = \underline{V} \times W$ tal que

$$a((\underline{u}, p), (\underline{v}, w)) = L((\underline{v}, w)), \quad \forall (\underline{v}, w) \in \hat{M} = \hat{\underline{V}} \times \hat{W} \quad (2.64)$$

donde las formas bilineal a y lineal L se definen como:

$$a((\underline{u}, p), (\underline{v}, w)) = \int_{\Omega} (\underline{u} \cdot \underline{v} + (\nabla p) \cdot \underline{v} - \underline{u} \cdot \nabla w) d\underline{x} \quad (2.65)$$

$$L((\underline{v}, w)) = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_N} g w ds \quad (2.66)$$

Nótese que la condición de Dirichlet $p = p_{\partial}$ en $\partial\Omega_D$ entra directamente en la definición del espacio mixto $M = \underline{V} \times W$ (esta es una condición de frontera esencial) y que la condición de Neumann $\underline{u} \cdot \underline{n} = g_{\partial}$ entra en el problema variacional (esta es una condición de frontera natural).

Se puede discretizar la formulación mixta-dual (2.64), restringiendo el problema a un par de espacios mixtos discretos $M_h \subset M$ y $\hat{M}_h \subset M$.

Encontrar $(\underline{u}_h, p_h) \in M_h = \underline{V}_h \times W_h$ tal que

$$a((\underline{u}_h, p_h), (\underline{v}, w)) = L((\underline{v}, w)), \quad \forall (\underline{v}, w) \in \hat{M}_h = \hat{\underline{V}}_h \times \hat{W}_h \quad (2.67)$$

donde la forma bilineal a y la forma lineal L se definen como:

$$a((\underline{u}_h, p_h), (\underline{v}, w)) = \int_{\Omega} (\underline{u}_h \cdot \underline{v} + (\nabla p_h) \cdot \underline{v} - \underline{u}_h \cdot \nabla w) d\underline{x} \quad (2.68)$$

$$L((\underline{v}, w)) = \int_{\Omega} f w d\underline{x} - \int_{\partial\Omega_N} g w ds \quad (2.69)$$

Para resolver el problema variacional discreto mixto -dual (2.67), se va a suponer que se tiene una base $\{(\phi_j, \psi_j)\}_{j=1}^N$ del espacio M_h y una base $\{(\hat{\phi}_j, \hat{\psi}_j)\}_{j=1}^N$ del espacio \hat{M}_h . Aquí N denota la dimensión de los espacios M_h y \hat{M}_h .

Se puede hacer una estimación para la solución mixta-dual de elementos finitos de la forma

$$(\underline{u}_h, p_h) = \sum_{j=1}^N Pj(\phi_j, \psi_j) \quad (2.70)$$

resolviendo un sistema lineal $\underline{\underline{A}} \cdot \underline{\underline{P}} = \underline{\underline{b}}$ para los grados de libertad $\underline{\underline{P}} \in \mathbb{R}^N$, donde

$$A_{ij} = \int_{\Omega} [\psi_j \cdot \hat{\psi}_i + (\nabla \phi_j) \cdot \hat{\psi}_i - \psi_j \cdot \nabla \hat{\phi}_i] d\underline{x} \quad (2.71)$$

$$b_i = \int_{\Omega} f \hat{\phi}_i - \int_{\partial\Omega_N} g \hat{\phi}_i ds \quad (2.72)$$

Las formulaciones mixtas duales requieren cierta atención en la selección de espacios que discretizan los diferentes espacios de funciones. Aquí, una combinación estable de espacios de elementos finitos mixta-dual es, para \underline{V}_h los elementos de *Discontinuous Raviart-Thomas* (DRT) de orden polinomial k , mientras que para W_h se recomienda utilizar los elementos de *Lagrange* (CG) de orden polinomial $k+1$ [18]. Recordemos que los métodos mixtos de elementos finitos estables deben satisfacer la condición de Ladyzhenskaya-Babuska-Brezzi [4].

Capítulo 3

El proyecto FEniCS

En el presente capítulo se describe el proyecto FEniCS, el cual es una colección de componentes de software libre de código abierto, con el objetivo común de permitir la solución automatizada de ecuaciones diferenciales. Los componentes proporcionan herramientas de computación científica para trabajar con mallas computacionales, formulaciones variacionales de elementos finitos de ecuaciones diferenciales ordinarias y parciales y álgebra lineal numérica [18]. Se describe brevemente cada uno de sus módulos y su interacción. Se muestra como ejemplo, la forma de implementar la ecuación de Poisson en su formulación variacional clásica, mixta y mixta-dual.

El proyecto FEniCS es un proyecto de investigación con el objetivo común de crear software libre, de código abierto, que funcione en los sistemas operativos Linux, Mac y Windows, para resolver computacionalmente modelos matemáticos basados en ecuaciones diferenciales parciales, mediante el método de elementos finitos con un código corto, eficiente e intuitivo, que se puede implementar en los lenguajes de programación *Python* y *C++* y ejecutar en varias plataformas que van desde laptops hasta clusters de alto rendimiento, ofreciendo otras capacidades muy poderosas a programadores experimentados.[18]

El proyecto FEniCS se creó inicialmente en el año 2003 como una colaboración entre la Universidad de Chicago y la Universidad Tecnológica de Chalmers. Actualmente cuenta con la colaboración de una gran cantidad de Universidades y prestigiosos centros de investigación alrededor del Mundo, que participan activamente en su desarrollo.

3.1. Componentes

La biblioteca **FEniCS**, está conformada por varios bloques de construcción (componentes de software) que se describen a continuación y se muestran en la figura (3.1).

DOLFIN es una biblioteca de C++ y Python que funciona como el principal entorno de resolución de problemas y la interfaz de usuario de FEniCS. Su funcionalidad integra los otros componentes de FEniCS y maneja la comunicación entre los diversos componentes de FEniCS y el software externo, implementa el entorno de resolución de problemas, proporciona estructuras de datos tales como mallas de elementos finitos, funciones, espacios de funciones, algoritmos de cómputo y cálculo, ensamblaje de elementos finitos automatizado, refinamientos de la malla, solucionadores de álgebra lineal numérica, etc.

FIAT es el tabulador automático de elementos finitos, responsable de generar la construcción numérica de funciones de base de elementos finitos para FFC, trabajando principalmente en términos de álgebra lineal numérica utilizando el paquete NumPy, para obtener eficiencia en Python.

UFL (Unified Form Language) implementa el lenguaje matemático abstracto mediante el cual los usuarios pueden expresar ecuaciones diferenciales en términos de formas variacionales de elementos finitos. Es un lenguaje específico para la declaración de discretizaciones de elementos finitos de formas variacionales y funcionales. Más precisamente, el lenguaje define una interfaz de usuario flexible para definir espacios de elementos finitos y expresiones de formas variacionales en una notación cercana a la notación matemática.

FFC (FEniCS Form Compiler) es el motor de generación de código de FEniCS, responsable de generar código de C++ eficiente a partir de abstracciones matemáticas de alto nivel. Es un compilador para formas variacionales de elementos finitos que toma el código UFL como entrada y genera la salida UFC. Una de las características clave de FEniCS es la generación automatizada de código para la solución general y eficiente de problemas variacionales de elementos finitos. Esta generación automatizada de código se basa en un compilador de código "just-in-time".

UFC (Unified Form-Assembly Code) es una interfaz de C++ que consiste en funciones de bajo nivel para evaluar y ensamblar formas variacionales de elementos finitos. UFC es una interfaz entre componentes específicos y de propósito general de programas de elementos finitos. En particular, la interfaz UFC define la estructura y la firma del código generado por los compiladores de formulario FFC y SFC para DOLFIN. La interfaz de UFC se aplica a una amplia gama de problemas de elementos finitos (incluyendo elementos finitos mixtos y métodos discontinuos de Galerkin) y puede usarse con bibliotecas que difieren ampliamente en su diseño. Para ello, la interfaz no depende de otros componentes FEniCS (u otras bibliotecas) y consiste sólo en un conjunto mínimo de clases abstractas de C++ que usan arrays C simples para la transferencia de datos.

Instant es un pequeño módulo de Python para la compilación "just-in-time" del código de C y C++ en Python. Instant acepta código C y C++ y por lo tanto es convenientemente combinado con las herramientas de generación de código en DOLFIN, FFC y SFC.

SyFi y SFC: SyFi (Symbolic Finite elements) es una biblioteca de elementos finitos y su compilador es SFC (SyFi Form Compiler). SyFi es un marco para definir simbólicamente elementos finitos, en muchos aspectos, SyFi es el equivalente de FIAT, mientras que SFC corresponde a FFC. SFC se puede utilizar en FEniCS como un compilador de formularios. De forma similar a FFC, se traduce el código UFL en el código UFC, que puede ser utilizado por el ensamblador DOLFIN. El código UFC es compilado con JIT mediante Instant.

FErari proporciona una opción dentro del compilador de formularios FFC para aplicar optimizaciones basadas en gráficos en tiempo de compilación. FErari proporciona un beneficio práctico al reducir el tiempo de ejecución sin reducir el orden general de complejidad de los cálculos de elementos finitos.

3.2. Proceso de solución con FEniCS

La solución de un problema físico con FEniCS consta de los siguientes pasos:

1. Identificar el dominio, las ecuaciones diferenciales parciales que caracterizan el fenómeno físico, sus condiciones iniciales y de frontera.
2. Reformular las ecuaciones diferenciales parciales como un problema variacional de elementos finitos (discreto).
3. Escribir un programa en Python o C++ donde se defina el dominio computacional (la malla), el problema variacional, los espacios de funciones (grado polinomial y tipo), los datos de entrada, las condiciones iniciales y de frontera, etc. utilizando las abstracciones correspondiente con la sintaxis de FEniCS.
4. Añadir declaraciones al programa, llamando a FEniCS para resolver el problema variacional con valores en la frontera, opcionalmente calcular flujos, promedios, etc. y finalmente visualizar los resultados.

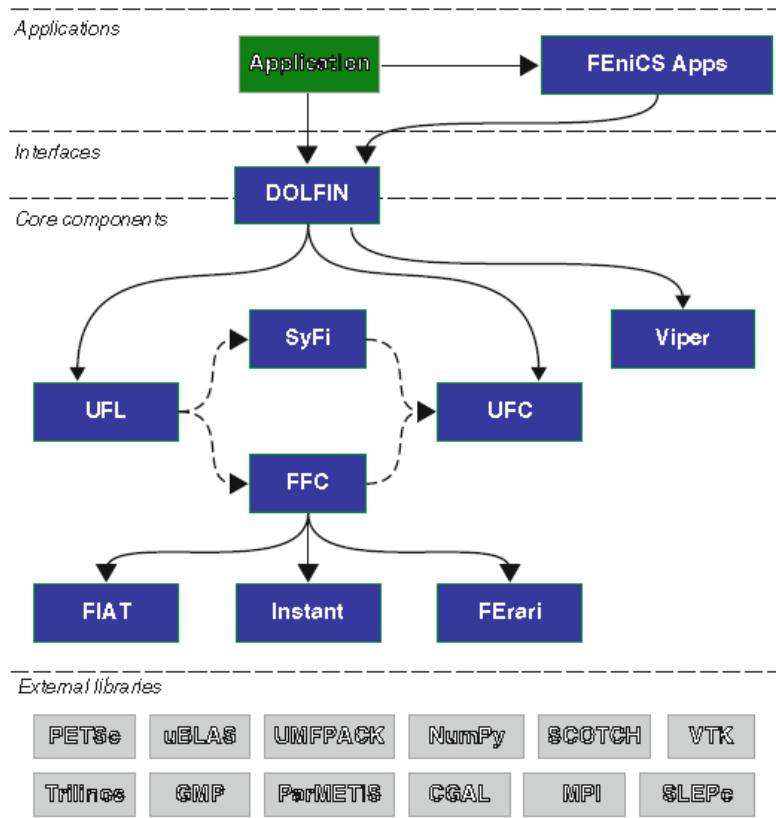


Figura 3.1: Comunicación entre los diversos componentes de FEniCS. Las líneas continuas indican las dependencias y las líneas discontinuas indican el flujo de datos.

3.3. Ejemplo: la ecuación de Poisson

A continuación se describe detalladamente paso a paso la implementación computacional mediante el lenguaje de programación Python 2.7 utilizando FEniCS 2017.1 para resolver la ecuación de Poisson 3.1 con cada uno de los tres tipos de formulaciones variacionales: Clásica, Mixta y Mixta-Dual, vistas en la sección (2.8).

$$-\nabla \cdot (\nabla p) = f, \quad \forall \underline{x} \in \Omega \quad (3.1)$$

$$p(\underline{x}) = p_0(\underline{x}), \quad \forall \underline{x} \in \partial\Omega \quad (3.2)$$

Para mostrar la implementación computacional, se necesitará tomar valores específicos para Ω , p_0 y f . Será prudente construir un problema con una solución analítica propuesta para que posteriormente se pueda calcular el orden de aproximación de la solución numérica obtenida con FEniCS con la solución analítica propuesta.

Se propone la función $p_e(x, y) = \sin(\pi x)\sin(\pi y)$ como la solución analítica en $2D$, la cual, al ser sustituida en la ecuación de Poisson (3.1), se deduce que $f(x, y) = 2\pi^2 p(x, y)$; además se considera $p_0 = 0$ sobre la frontera y el dominio Ω igual al cuadrado unitario, obteniendo el siguiente problema con valores en la frontera:

$$-\nabla \cdot (\nabla p) = 2\pi^2 p(x, y), \quad \forall \underline{x} \in [0, 1] \times [0, 1] \quad (3.3)$$

$$p(\underline{x}) = 0, \quad \forall \underline{x} \in \partial[0, 1] \times [0, 1] \quad (3.4)$$

3.3.1. Implementación con la forma variacional clásica

Lo primero que se debe hacer para trabajar con FEniCS Project es importarlo escribiendo la siguiente declaración en la primera línea en el código de Python:

```
from fenics import *
```

Se puede crear una malla triangular uniforme que sea un cuadrado unitario mediante la función `UnitSquareMesh()`, donde para este caso será de 8 divisiones en el eje x (primer argumento) por 8 divisiones en el eje y (segundo argumento)

```
mesh = UnitSquareMesh(8, 8)
```

Mediante la función `FunctionSpace()` se puede definir un espacio de funciones V sobre la malla `mesh` definida anteriormente con elementos finitos tipo *Lagrange* de orden 2

```
V = FunctionSpace(mesh, 'Lagrange', 2)
```

Se puede definir la condición de frontera de tipo Dirichlet, como en la ecuación (3.4), mediante la función `DirichletBC()` sobre el espacio de funciones V donde $p(\underline{x}) = 0$ sobre toda la frontera de la siguiente manera:

```
def boundary(x, on_boundary):
    return on_boundary
bc = DirichletBC(V, Constant(0.0), boundary)
```

Se puede definir la solución exacta $p_e = \sin(\pi x)\sin(\pi y)$ mediante la función `Expression()` de la siguiente manera:

```
p_e = Expression('sin(pi*x[0])*sin(pi*x[1])', degree=2)
```

Las funciones de base p se definen mediante la función `TrialFunction()` sobre el espacio de funciones V

```
p = TrialFunction(V)
```

Las funciones de peso w se definen mediante la función `TestFunction()` sobre el espacio de funciones V

```
w = TestFunction(V)
```

Se puede definir el valor del término fuente f de la siguiente manera:

```
f = 2*pi**2*p_e
```

Se define la forma bilineal a de la formulación variacional como en la ecuación (2.30)

```
a = inner(grad(p), grad(w))*dx
```

Se define la forma lineal L de la formulación variacional como en la ecuación (2.31)

```
L = f*w*dx
```

Se define una función p sobre el espacio de funciones V , mediante la función `Function()`

```
p = Function(V)
```


Se resuelve la ecuación mediante la función `solve()` de la siguiente manera:

```
solve(a == L, p, bc)
```

finalmente se grafica la malla y la solución respectivamente mediante la función `plot()`

```
plot(mesh)
plot(p)
```

después de compilar el programa, se obtienen las gráficas de la figura (3.2).

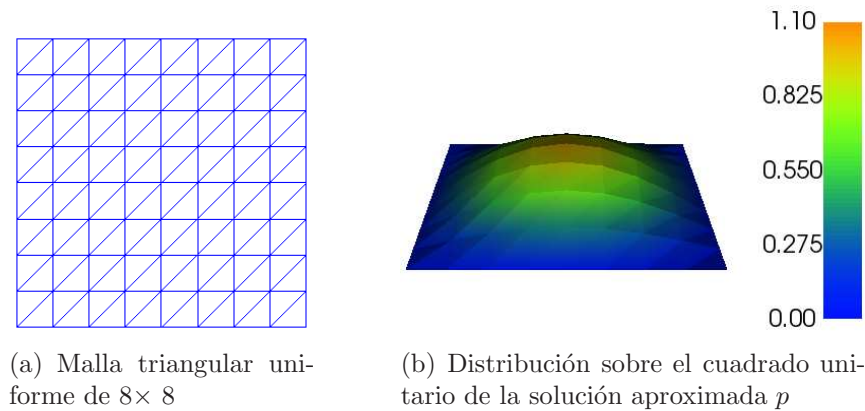


Figura 3.2: Solución de la ecuación de Poisson mediante la formulación clásica y polinomios de Lagrange de orden 2 obtenida con FEniCS

También se puede refinar la malla tanto como se desee y aumentar el orden del grado del polinomio para obtener una mejor aproximación, pero el tiempo de cálculo es mayor.

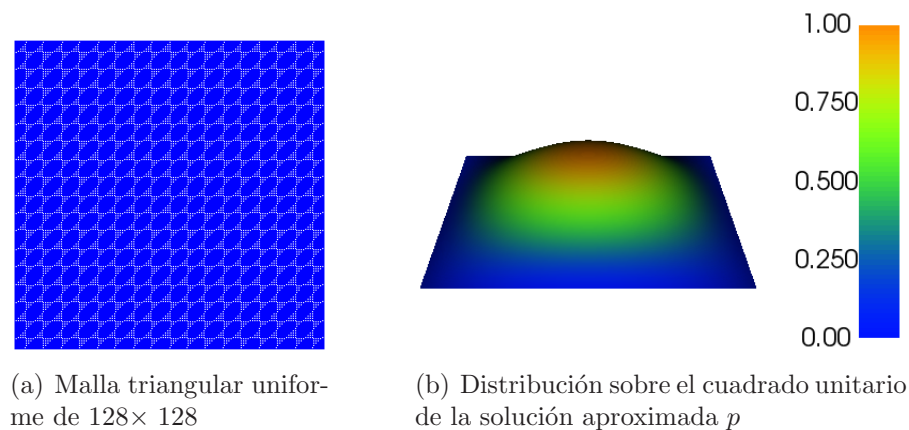


Figura 3.3: Solución de la ecuación de Poisson mediante la formulación clásica y polinomios de Lagrange de orden 4 obtenida con FEniCS

3.3.2. Implementación con la forma variacional mixta

Como se vio anteriormente, lo primero que se debe hacer para trabajar con FEniCS es importarlo escribiendo la siguiente declaración en la primera línea del código:

```
from fenics import *
```

Se crea la malla triangular uniforme, que cubra el cuadrado unitario del ejemplo

```
mesh = UnitSquareMesh(16, 16)
```

Se definen de elementos triangulares de tipo *Brezzi-Douglas-Marini* de orden 2, que se usarán para el cálculo de la velocidad mediante la clase `FiniteElement()`, en la cual, el primer argumento especifica el tipo de elementos finitos, el segundo elemento, la forma de los elementos y el tercer argumento especifica el grado polinomial.

```
BDM = FiniteElement("BDM", 'triangle', 2)
```

De la misma forma, se definen de elementos triangulares de tipo *Discontinuous-Galerkin* de orden 1, que se usarán para el cálculo de la presión y debe ser de un menor grado que los de la velocidad

```
DG = FiniteElement("DG", 'triangle', 1)
```

Se definen los elementos mixtos utilizando los elementos definidos anteriormente, mediante la clase `MixedElement()`, es decir, $W = \{(\underline{u}, p) | \underline{u} \in BDM \text{ y } p \in DG\}$

```
mixed_element = MixedElement([BDM, DG])
```

Se define el espacio de funciones W de elementos finitos mixtos sobre la malla *mesh* y los elementos mixtos definidos anteriormente

```
W = FunctionSpace(mesh, mixed_element)
```

Se definen las funciones de base u para la velocidad y p para la presión sobre el espacio de funciones W , mediante la clase `TrialFunctions()`

```
(u, p) = TrialFunctions(W)
```

Se definen las funciones de peso v para la velocidad y w para la presión sobre el espacio de funciones W , mediante la clase `TestFunctions()`

```
(v, w) = TestFunctions(W)
```

Se define el vector normal a la frontera, que se utilizará en la formulación variacional

```
n = FacetNormal(mesh)
```

Se define la solución exacta $p_e = \sin(\pi x)\sin(\pi y)$

```
p_exact = Expression('sin(pi*x[0])*sin(pi*x[1])', degree=1)
```

Se define el valor del término fuente f de la siguiente manera:

```
f = 2*pi**2*p_exact
```

Se define la condición de frontera de tipo Dirichlet como en la ecuación (3.4). Para aplicar la condición de frontera al segundo subespacio del espacio mixto W , se debe acceder a éste mediante el método `sub()`, que en este caso, se hace con la declaración `W.sub(1)` haciendo referencia al espacio de funciones **DG** del espacio mixto W , mientras que en caso de querer hacer referencia al espacio de funciones **BDM**, se debe usar `W.sub(0)`. También se declara el valor de **p0** como en la ecuación (3.4) porque se utilizará más adelante en la formulación variacional. En el método de elementos finitos mixto, las condiciones de frontera para el flujo son condiciones de frontera *esenciales* (que deben indicarse en el espacio de funciones), mientras que las condiciones de frontera de tipo Dirichlet, son condiciones de frontera *naturales* (que deben aplicarse a la forma variacional). Las condiciones de frontera esenciales se especifican a través de la clase `DirichletBC()` que toma tres argumentos: el espacio de funciones al que se va a aplicar la condición de frontera, los datos de la condición de frontera y la parte de la frontera.

```
def boundary(x, on_boundary):
    return on_boundary
p0 = Constant(0.0)
bc=DirichletBC(W.sub(1), p0, boundary)
```

Se define la forma bilineal a de la formulación variacional mixta como en la ec. (2.46)

```
a = dot(u, v)*dx - div(v)*p*dx + div(u)*w*dx
```

Se define la forma lineal L de la formulación variacional mixta como en la ec. (2.47)

```
L = f*w*dx - p0*inner(v,n)*ds
```

Se define la función *sol* en el espacio de funciones V para almacenar las soluciones numéricas tanto de la presión p como de la velocidad \underline{u} .

```
sol = Function(W)
```

El cálculo de la solución se realiza llamando a la función `solve()`, utilizando las formas bilineal a , la forma lineal L , la función *sol* definida anteriormente que almacena las soluciones de \underline{u} y p , la condición de frontera *bc* de la siguiente manera:

```
solve(a == L, sol, bc)
```

Los componentes \underline{u} y p de la solución se pueden extraer llamando a la función `split()`

```
(u, p) = sol.split()
```

finalmente se grafican las soluciones \underline{u} y p para analizar el resultado.

```
plot(u)
plot(p)
```

después de compilar el programa, se obtienen las gráficas de la figura (3.4).

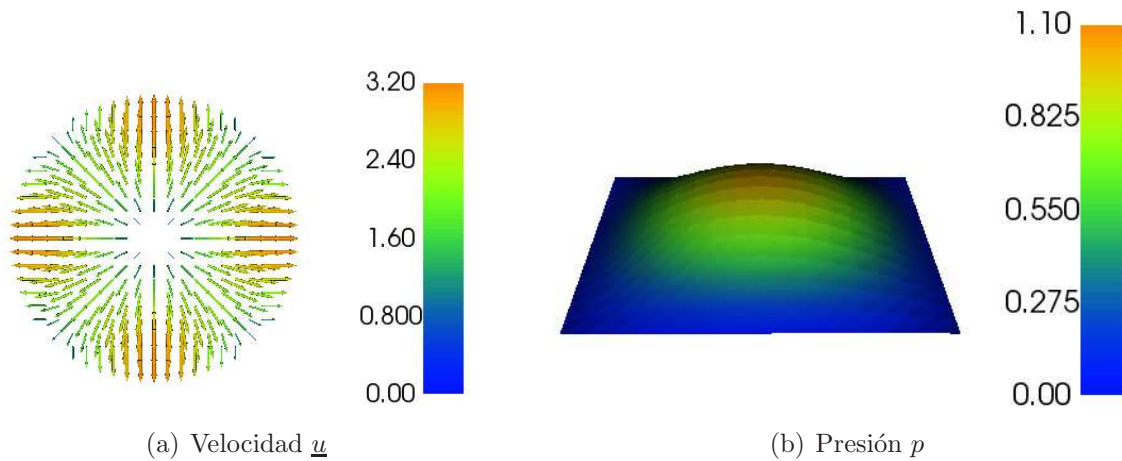


Figura 3.4: Distribución sobre el cuadrado unitario de la solución numérica aproximada de la velocidad y presión en la ecuación de Poisson mediante la forma variacional mixta, obtenida con FEniCS, sobre una malla triangular uniforme de 16×16

3.3.3. Implementación con la forma variacional mixta-dual

Como se vió en los ejemplos anteriores, primero se debe importar FEniCS escribiendo la siguiente línea en el código de Python:

```
from fenics import *
```

después se crea una malla triangular uniforme que cubra el cuadrado unitario del dominio del ejemplo, de 8×8 cuadrados divididos en dos triángulos cada uno.

```
mesh = UnitSquareMesh(8, 8)
```

luego se definen sobre la malla los elementos triangulares de tipo *DRT* de orden 2, que se usarán para el cálculo de la velocidad

```
DRT = FiniteElement('DRT', 'triangle', 2)
```

de la misma manera se definen sobre la malla los elementos triangulares de tipo *CG* de orden 3, que se usarán para el cálculo de la presión y deben ser de un grado mayor que los de la velocidad

```
CG = FiniteElement('CG', 'triangle', 3)
```

Se definen los elementos mixtos utilizando los elementos definidos anteriormente

```
mixed_element = MixedElement([DRT, CG])
```

Se define el espacio de funciones W de elementos finitos mixtos sobre la malla

```
W = FunctionSpace(mesh, mixed_element)
```

se definen las funciones de base u para la velocidad y p para la presión sobre W

```
(u, p) = TrialFunctions(W)
```

se definen las funciones de peso v para la velocidad y w para la presión sobre W

```
(v, w) = TestFunctions(W)
```

Se define la solución exacta $p_e = \sin(\pi x)\sin(\pi y)$

```
p_exact = Expression('sin(pi*x[0])*sin(pi*x[1])', degree=2)
```

Se define el valor del término fuente f de la siguiente manera:

```
f = 2*pi**2*p_exact
```

Se aplica la condición de frontera para p , al segundo subespacio del espacio mixto-dual, accediendo a éste con el método `sub()`. En el método de elementos finitos mixto-dual, las condiciones de frontera de flujo son las condiciones de frontera *naturales* (que deben aplicarse a la forma variacional), mientras que las condiciones de frontera de tipo Dirichlet son condiciones de frontera *esenciales* (que deben indicarse en el espacio de funciones).

```
def boundary(x, on_boundary):
    return on_boundary
bc = DirichletBC(W.sub(1), Constant(0.0), boundary)
```

Se define la forma bilineal a de la formulación mixta-dual como en la ecuación ec. (2.65)

```
a = (dot(u, v) + dot(grad(p), v) + dot(u, grad(w)))*dx
```

Se define la forma lineal L de la formulación mixta-dual como en la ecuación ec. (2.66)

```
L = - f*v*dx
```

Se crea un objeto de la clase `Function()` en el espacio de funciones W , llamado *sol* para almacenar las soluciones numéricas tanto de la velocidad \underline{u} como de la presión p .

```
sol = Function(W)
```

Se calcula la solución

```
solve(a == L, sol, bc)
```

Se extraen los componentes \underline{u} y p de la función *sol* que los almacena

```
(u, p) = sol.split()
```

finalmente se grafican las soluciones numéricas \underline{u} y p para analizar el resultado.

```
plot(u)
plot(p)
```

después de compilar el programa, se obtienen las gráficas de la velocidad \underline{u} y la presión p respectivamente como se muestra en la figura 3.5..

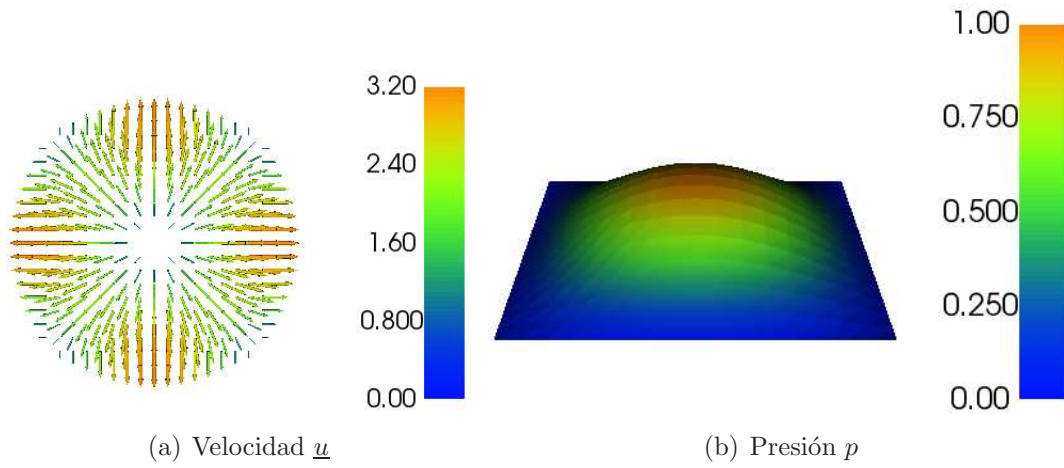


Figura 3.5: Distribución sobre el cuadrado unitario de la solución numérica aproximada de la velocidad y presión en la ec. de Poisson mediante la forma variacional mixta-dual, obtenida con FEniCS, sobre una malla triangular uniforme de 16×16

Capítulo 4

Modelo de flujo monofásico

En el presente capítulo se desarrolla un modelo de flujo monofásico ligeramente compresible y se muestra el desarrollo de las formulaciones variacionales con el método de elementos finitos clásico, mixto y mixto-dual, para discretizar el espacio y el método de Euler hacia atrás para discretizar el tiempo. Se hace un análisis de las tasas de las convergencias de los tres métodos de elementos finitos: clásico, mixto y mixto-dual.

4.1. Modelo conceptual

1. El fluido es ligeramente compresible.
2. El flujo del fluido en el medio poroso es monofásico con un componente.
3. Consideraremos un fluido Newtoniano que ocupa el espacio intersticial del medio poroso, bajo condiciones isotérmicas.
4. Se considera la ley de Darcy.
5. El flujo de la masa debido a la dispersión y difusión son tan pequeños con relación al flujo de la masa que se pueden despreciar.
6. La interfase fluido-sólido es una superficie material con respecto a la masa del fluido tal que ninguna masa del fluido puede cruzarla.
7. Hay conservación en la masa del fluido.
8. El medio es isótropo.

Notación

- ϕ - porosidad en el medio poroso
- ρ - densidad del fluido por unidad de volumen
- \underline{u} - velocidad de Darcy
- \underline{k} - tensor de permeabilidad absoluta del medio poroso
- μ - viscosidad del fluido
- γ - magnitud de la aceleración gravitacional
- z - profundidad
- p - presión
- c_f - Compresibilidad del fluido
- c_R - Compresibilidad de la roca
- c_t - Compresibilidad total
- r_w - Radio del pozo

4.2. Modelo matemático

Los problemas básicos que deben abordarse en el modelado y la simulación de flujos de fluidos tanto en el petróleo como en los depósitos de agua subterránea son análogos. Un depósito de petróleo es un medio poroso que contiene una serie de hidrocarburos. El objetivo principal de la simulación del yacimiento es predecir el rendimiento futuro de un yacimiento y encontrar las vías y medios para optimizar la recuperación de algunos de los hidrocarburos[8].

Los modelos matemáticos de yacimientos petroleros han sido utilizados desde el siglo XIX y consisten en un conjunto de ecuaciones que describen el flujo de fluidos en un yacimiento petrolero, junto con un conjunto apropiado de condiciones iniciales y de frontera.

Usando el enfoque sistemático para la modelación de sistemas continuos derivaremos a continuación las ecuaciones que constituyen el modelo matemático de flujo monofásico a través de un medio poroso [2]. El sistema está formado por dos fases; pero el movimiento de la fase sólida está en reposo, trataremos solamente la fase fluida que consta de una sola componente [15].

El modelo del flujo se construye basándose en una **propiedad extensiva**: la masa del fluido M_f y por la hipótesis de que el medio poroso está saturado, la **propiedad intensiva** es el producto de la densidad por la porosidad $\rho(\underline{x}, t)\phi(\underline{x}, t)$, donde ϕ es la porosidad, es decir, la fracción del volumen del espacio físico ocupado por los poros y como el material poroso está saturado, es igual a la fracción de volumen ocupado por el fluido.

$$M_f(t) = \int_{\Omega} \rho(\underline{x}, t)\phi(\underline{x}, t) \, d\underline{x} \quad (4.1)$$

Al considerar que hay conservación en la masa del fluido, entonces no hay una fuente y no existen discontinuidades, por lo tanto, la ecuación de balance global está expresada de la siguiente manera:

$$\frac{d}{dt}M_f(t) = 0 \quad (4.2)$$

y la ecuación diferencial de balance local esta determinada por:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla \cdot (\rho\phi \underline{v}) = 0, \quad \forall \underline{x} \in \Omega \quad (4.3)$$

La velocidad en el medio poroso o velocidad de Darcy se define como $\underline{u} = \phi \underline{v}$, entonces la ecuación de balance local en un medio poroso se define como:

$$\frac{\partial}{\partial t}(\phi\rho) = -\nabla \cdot (\rho \underline{u}), \quad \forall \underline{x} \in \Omega \quad (4.4)$$

Efecto de la elasticidad del sistema fluido-sólido

Podemos expresar el término $\partial\phi\rho/\partial t$ en una forma ampliamente utilizada en la aplicaciones de este tipo. Desarrollando éste término, obtenemos que

$$\frac{\partial}{\partial t}(\phi\rho) = \phi\frac{\partial\rho}{\partial t} + \rho\frac{\partial\phi}{\partial t} \quad (4.5)$$

Esta ecuación está integrada por la contribución de la compresibilidad del fluido y por la contribución de la compresibilidad de la roca. **La compresibilidad del fluido** c_f se define

como

$$c_f = \frac{1}{\rho} \frac{\partial\rho}{\partial p} \quad (4.6)$$

La compresibilidad de la roca c_R se define como

$$c_R = \frac{1}{\phi} \frac{\partial\phi}{\partial p} \quad (4.7)$$

La compresibilidad total c_t se define como

$$c_t = c_f + c_R = \frac{1}{\rho} \frac{\partial\rho}{\partial p} + \frac{1}{\phi} \frac{\partial\phi}{\partial p} \quad (4.8)$$

La densidad del fluido satisface una ecuación de estado de acuerdo con la cual la densidad del agua es una función de su presión

$$\frac{\partial\rho}{\partial t} = \frac{\partial\rho}{\partial p} \frac{\partial p}{\partial t} \quad (4.9)$$

por definición de compresibilidad del fluido (4.6), tenemos que

$$\frac{\partial\rho}{\partial t} = c_f \rho \frac{\partial p}{\partial t} \quad (4.10)$$

Análogamente, la porosidad de la roca satisface una ecuación de estado de acuerdo con la cual la porosidad de la roca es una función de su presión

$$\frac{\partial\phi}{\partial t} = \frac{\partial\phi}{\partial p} \frac{\partial p}{\partial t} \quad (4.11)$$

por definición de compresibilidad de la roca (4.7), tenemos que

$$\frac{\partial\phi}{\partial t} = c_R \phi \frac{\partial p}{\partial t} \quad (4.12)$$

sustituyendo (4.10) y (4.12) en (4.5), obtenemos que

$$\frac{\partial\rho\phi}{\partial t} = \phi\rho c_f \frac{\partial p}{\partial t} + \rho\phi c_R \frac{\partial p}{\partial t} = \phi\rho (c_f + c_R) \frac{\partial p}{\partial t} \quad (4.13)$$

por definición de compresibilidad total (4.8), obtenemos que

$$\frac{\partial\rho\phi}{\partial t} = \phi\rho c_t \frac{\partial p}{\partial t} \quad (4.14)$$

sustituyendo en la ecuación (4.4) obtenemos que

$$\phi\rho c_t \frac{\partial p}{\partial t} = -\nabla \cdot (\rho \underline{u}), \quad \forall \underline{x} \in \Omega \quad (4.15)$$

La ley de Darcy

Esta ley es una ecuación constitutiva, empírica, que relaciona a la velocidad efectiva del fluido con su presión, indicando una relación lineal entre la velocidad del fluido en el medio poroso y el gradiente de la presión, siendo utilizada ampliamente para la modelación de flujo en medios porosos.

$$\underline{u} = -\frac{1}{\mu} \underline{k} \cdot (\nabla p - \rho \gamma \nabla z) \quad (4.16)$$

donde: ϕ — es la porosidad del medio, ρ — es la densidad del fluido, c_t — es la compresibilidad

total $\equiv c_f + c_R$, p — es la presión del fluido, μ — es la viscosidad del fluido, \underline{k} — es el tensor de permeabilidad del medio poroso, γ — es la magnitud de la aceleración gravitacional y z — es la altura.

Modelo resultante

Aplicando la ley de Darcy (4.16) a la ecuación (4.15), obtenemos la ecuación diferencial básica que describe el flujo monofásico en medios porosos con condiciones iniciales y de frontera. [9].

$$(\rho \phi c_t) \frac{\partial p}{\partial t} - \nabla \cdot \left(\frac{\rho}{\mu} \underline{k} \cdot (\nabla p - \rho \gamma \nabla z) \right) = q \quad , \quad \forall \underline{x} \in \Omega \quad y \quad t \geq t_0 \quad (4.17)$$

$$p = p_0 \quad , \quad \forall \underline{x} \in \Omega \quad y \quad \forall t = t_0 \quad (4.18)$$

$$p = p_\partial \quad , \quad \forall \underline{x} \in \partial_D \Omega \quad y \quad \forall t > t_0 \quad (4.19)$$

$$\underline{n} \cdot \left(\frac{\rho}{\mu} \underline{k} \cdot (\nabla p - \rho \gamma \nabla z) \right) = g_\partial \quad , \quad \forall \underline{x} \in \partial_N \Omega \quad y \quad \forall t > t_0 \quad (4.20)$$

donde \underline{k} es un tensor simétrico, positivo definido, acotado y suave a tramos en Ω de la forma $\frac{\rho}{\mu} \underline{k} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix}$, con $k_{ij} \equiv k_{ij}(\underline{x}, t)$, mientras que $\rho \equiv \rho(\underline{x}, t)$, $\mu \equiv \mu(\underline{x}, t)$, $\phi \equiv \phi(\underline{x}, t)$ y $q \equiv q(\underline{x}, t)$. Ω es un dominio acotado con frontera $\partial\Omega$, a su vez la frontera está formada por dos partes, una con condiciones de tipo Dirichlet $\partial_D \Omega$ y otra con condiciones de tipo Neumann $\partial_N \Omega$, donde $\partial\Omega = \partial_D \Omega \cup \partial_N \Omega$, con $\partial_D \Omega \cap \partial_N \Omega = \emptyset$.

4.3. Modelo numérico

Se desarrollarán las formulaciones variacionales con tres variantes del Método de Elementos Finitos (MEF): Clásico, Mixto y Mixto-Dual. Los modelos numéricos van a considerar un dominio en forma de cilindro parado verticalmente, cuya numeración de las fronteras se muestra a continuación:



Figura 4.1: Dominio del modelo de flujo monofásico y sus respectivas fronteras con la indexación convenida. En la frontera Γ_1 se inyecta a una velocidad constante, en la frontera Γ_3 se produce a presión constante y la superficie lateral curva del cilindro es impermeable.

4.3.1. Modelo numérico con la formulación clásica

Usualmente se prefiere para resolver las ecuaciones diferenciales parciales dependientes del tiempo mediante el método de elementos finitos realizar una semidiscretización en el tiempo de la ecuación diferencial, primero discretizando la derivada temporal mediante una aproximación de diferencias finitas, produciendo una secuencia de problemas estacionarios y luego convirtiendo cada problema estacionario en una formulación variacional.

Supongamos que el superíndice n denota una cantidad en el tiempo t_n , donde n es un número entero que cuenta los niveles del tiempo, por ejemplo, p^n significa p en el nivel de tiempo n , entonces por simplicidad y por razones de estabilidad, se utilizará un esquema de diferencias finitas hacia atrás para el término del tiempo.

$$\left(\frac{\partial p}{\partial t}\right)^n = \frac{p^n - p^{n-1}}{\Delta t_n} + O(\Delta t_n) \quad (4.21)$$

donde $p^n \equiv p(\underline{x}, t_n)$ y el tamaño de paso del tiempo es $\Delta t_n \equiv t_n - t_{n-1}$ con $n = 0, 1, \dots, N$. Esta sustitución, resulta en un error de discretización de orden $O(\Delta t_n)$.

Entonces, para un tiempo dado t_n , podemos reescribir la ecuación (4.17) como sigue:

$$(\rho \phi c_t)^n \frac{p^n - p^{n-1}}{\Delta t_n} = \nabla \cdot \left(\left(\frac{\rho \underline{k}}{\mu} \right)^n \cdot \nabla p^n - \rho^n \gamma \nabla z \right) + q^n \quad (4.22)$$

donde $(\underline{k})^n \equiv \underline{k}(\underline{x}, t_n)$, $\rho^n \equiv \rho(\underline{x}, t_n)$, $\phi^n \equiv \phi(\underline{x}, t_n)$, $\mu^n \equiv \mu(\underline{x}, t_n)$ y $q^n \equiv q(\underline{x}, t_n)$.

Multiplicando ambos lados de la igualdad en la ecuación (4.22) por Δt_n resulta:

$$(\rho\phi c_t)^n p^n - (\rho\phi c_t)^n p^{n-1} = \Delta t_n \nabla \cdot \left(\left(\frac{\rho k}{\mu} \right)^n \cdot \nabla p^n - \rho^n \gamma \nabla z \right) + \Delta t_n q^n \quad (4.23)$$

Reacomodando términos en la ecuación (4.23) para que el lado izquierdo de la igualdad contenga los términos con la incógnita p^n y el lado derecho de la igualdad contenga los términos calculados solamente. El resultado es una sucesión de problemas espaciales (estacionarios) para p^n , suponiendo que p^{n-1} se conoce desde el paso de tiempo anterior:

$$(\rho\phi c_t)^n p^n - \Delta t_n \nabla \cdot \left(\left(\frac{\rho k}{\mu} \right)^n \cdot \nabla p^n - \rho^n \gamma \nabla z \right) = (\rho\phi c_t)^n p^{n-1} + \Delta t_n q^n \quad (4.24)$$

Entonces, dada la condición inicial p_0 , se puede calcular p^1, p^2, p^3, \dots y así sucesivamente.

Para resolver la ecuación (4.24) utilizando el método de elementos finitos, se requiere convertir las ecuaciones en su forma débil, entonces como es costumbre, se multiplica la ecuación por una función de peso $w \in \hat{W}$ y se integra sobre el dominio Ω , resultando

$$\begin{aligned} & \int_{\Omega} (\rho\phi c_t)^n p^n w d\underline{x} - \Delta t_n \int_{\Omega} \nabla \cdot \left(\left(\frac{\rho k}{\mu} \right)^n \cdot \nabla p^n - \rho^n \gamma \nabla z \right) w d\underline{x} \\ &= \int_{\Omega} ((\rho\phi c_t)^n p^{n-1} + \Delta t_n q^n) w d\underline{x} \end{aligned} \quad (4.25)$$

aplicando a la segunda integral de la izquierda de la igualdad, el teorema de la divergencia, luego las condiciones de frontera naturales y esenciales y reacomodando términos

$$\begin{aligned} & \int_{\Omega} (\rho\phi c_t)^n p^n w d\underline{x} + \Delta t_n \int_{\Omega} \nabla w \cdot \left(\left(\frac{\rho k}{\mu} \right)^n \cdot \nabla p^n - \rho^n \gamma \nabla z \right) d\underline{x} \\ &= \int_{\Omega} ((\rho\phi c_t)^n p^{n-1} + \Delta t_n q^n) w d\underline{x} + \Delta t_n \int_{\partial_N \Omega} w g_{\partial}^n d\underline{x} \end{aligned} \quad (4.26)$$

Considerando que el efecto de gravedad se puede despreciar y que la densidad del fluido ρ es constante, entonces la formulación variacional de la ecuación diferencial con condiciones iniciales y de frontera (4.17) del modelo matemático de flujo monofásico a través de un medio poroso se puede reescribir de la siguiente manera

$$a^n(p^n, w) = L^n(w) \quad (4.27)$$

donde

$$a^n(p^n, w) = \int_{\Omega} (\phi c_t)^n p^n w d\underline{x} + \Delta t_n \int_{\Omega} \nabla w \cdot \left(\left(\frac{1}{\mu} k \right)^n \cdot \nabla p^n \right) d\underline{x} \quad (4.28)$$

$$L^n(w) = \int_{\Omega} ((\phi c_t)^n p^{n-1} + \Delta t_n q^n / \rho^n) w d\underline{x} + \Delta t_n \int_{\partial_N \Omega} w g_{\partial}^n d\underline{x} \quad (4.29)$$

Considerando la indexación convenida de las fronteras (ver figura 4.1), tenemos que en la frontera de salida Γ_3 se establece una condición tipo Dirichlet, $p(\underline{x}, t) = p_{out}(\underline{x}, t)$. Por otro lado, sea $\partial_N \Omega = \Gamma_1 \cup \Gamma_2$, donde Γ_2 es la superficie lateral curva del cilindro, entonces

$$\int_{\partial_N \Omega} w g_{\partial}^n d\underline{x} = \int_{\Gamma_1} w g_1 d\underline{x} + \int_{\Gamma_2} w g_2 d\underline{x} \quad (4.30)$$

Sean $g_1 = u_{in}$ en Γ_1 con $u_{in} = cte.$, $g_2 = 0$ en Γ_2 dado que es impermeable, entonces

$$\int_{\partial_N \Omega} w g_{\partial}^n d\underline{x} = \int_{\Gamma_1} w u_{in} d\underline{x} \quad (4.31)$$

Por lo tanto, la formulación variacional de la ecuación (4.17) queda de la siguiente manera:

Para cualquier $0 \leq n \leq N$, encontrar $p_h^n \in W_h \subset W$ tal que

$$a^n(p_h^n, w_h) = L^n(w_h) \quad \forall w_h \in \hat{W}_h \text{ y } t_n \in (t_0, t_N] \quad (4.32)$$

donde

$$a^n(p_h^n, w_h) = \int_{\Omega} (\phi c_t)^n p_h^n w_h d\underline{x} + \Delta t_n \int_{\Omega} \nabla w_h \cdot \left(\left(\frac{1}{\mu} \underline{k} \right)^n \cdot \nabla p_h^n \right) d\underline{x} \quad (4.33)$$

$$L^n(w_h) = \int_{\Omega} ((\phi c_t)^n p_h^{n-1} + \Delta t_n q^n / \rho^n) w_h d\underline{x} + \Delta t_n \int_{\Gamma_1} w_h u_{in} d\underline{x} \quad (4.34)$$

$W = \{w \in H^1(\Omega) : w|_{\partial_D \Omega} = p_{\partial}\}$ es el espacio de las funciones de base.

$\hat{W} = \{w \in H^1(\Omega) : w|_{\partial_D \Omega} = 0\}$ es el espacio de las funciones de peso.

H^1 es el espacio de Sobolev.

El método de diferencias finitas hacia atrás, también conocido como el método de Euler hacia atrás, es incondicionalmente estable. Esta es una característica muy deseable de un esquema de discretización del tiempo para un problema parabólico [8].

4.3.2. Modelo numérico con la formulación mixta

Reescribiendo el modelo matemático (ver ecuaciones 4.17-4.20) como un sistema de dos ecuaciones diferenciales en términos de la presión p y de la velocidad \underline{u} , considerando que la densidad es constante y despreciando la fuerza de gravedad de la siguiente manera:

$$\phi c_t \frac{\partial p}{\partial t} + \nabla \cdot \underline{u} = q / \rho \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (4.35)$$

$$\underline{u} = -\frac{1}{\mu} \underline{k} \cdot \nabla p \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (4.36)$$

$$p = p_0 \quad , \quad \forall \underline{x} \in \Omega \text{ y } t = t_0 \quad (4.37)$$

$$p = p_{\partial} \quad , \quad \forall \underline{x} \in \partial_D \Omega \text{ y } t > t_0 \quad (4.38)$$

$$-\underline{u} \cdot \underline{n} = g_{\partial} \quad , \quad \forall \underline{x} \in \partial_N \Omega \text{ y } t > t_0 \quad (4.39)$$

Por un lado, de la ecuación (4.36), se obtiene

$$\mu \underline{\underline{k}}^{-1} \cdot \underline{u} = -\nabla p \quad (4.40)$$

haciendo un producto interior con \underline{v} e integrando sobre el dominio

$$\int_{\Omega} (\mu \underline{\underline{k}}^{-1} \cdot \underline{u}) \cdot \underline{v} d\underline{x} = - \int_{\Omega} (\nabla p) \cdot \underline{v} d\underline{x} = \int_{\Omega} p \nabla \cdot \underline{v} d\underline{x} - \int_{\partial\Omega} p \underline{v} \cdot \underline{n} ds \quad (4.41)$$

Aplicando la condición de frontera de tipo Dirichlet (4.38), la cual es una condición de frontera natural debido a que las funciones de peso $\underline{v} \in \underline{\hat{V}} = \{\underline{v} \in H(\text{div}, \Omega) : \underline{v} \cdot \underline{n}|_{\partial_N \Omega} = 0\}$ y reacomodando términos

$$\int_{\Omega} ((\mu \underline{\underline{k}}^{-1} \cdot \underline{u}) \cdot \underline{v} - p \nabla \cdot \underline{v}) d\underline{x} = - \int_{\partial_D \Omega} p \underline{v} \cdot \underline{n} ds \quad (4.42)$$

Considerando la indexación convenida de las fronteras (ver fig. 4.1), se establece una condición tipo Dirichlet en la frontera de salida Γ_3 , donde $p(\underline{x}, t) = p_{out}(\underline{x}, t)$

$$\int_{\Omega} ((\mu \underline{\underline{k}}^{-1} \cdot \underline{u}) \cdot \underline{v} - p \nabla \cdot \underline{v}) d\underline{x} = - \int_{\Gamma_3} p_{out} \underline{v} \cdot \underline{n} ds \quad (4.43)$$

Ahora, para las condiciones de frontera esenciales, se define $\partial_N \Omega = \Gamma_1 \cup \Gamma_2$, donde Γ_2 es la superficie lateral curva del cilindro, $g_1 = u_{in}$ en Γ_1 con $u_{in} = \text{cte.}$ y $g_2 = 0$ en Γ_2 , entonces

$$\int_{\partial_N \Omega} w g_2^n d\underline{x} = \int_{\Gamma_1} w g_1 d\underline{x} + \int_{\Gamma_2} w g_2 d\underline{x} = \int_{\Gamma_1} w u_{in} d\underline{x} \quad (4.44)$$

Por otro lado, aplicando diferencias finitas hacia atrás para el término del tiempo en la ecuación (4.35), para un tiempo t_n dado

$$(\phi c_t)^n \frac{p^n - p^{n-1}}{\Delta t_n} + \nabla \cdot \underline{u}^n = (q/\rho)^n \quad (4.45)$$

donde $p^n \equiv p(\underline{x}, t_n)$ y el tamaño de paso del tiempo es $\Delta t_n \equiv t_n - t_{n-1}$ con $n = 0, 1, \dots, N$.

Multiplicando por una función de peso $w \in W = L^2(\Omega)$ e integrando sobre el dominio y reacomodando términos

$$\int_{\Omega} ((\phi c_t)^n p^n + \Delta t_n (\nabla \cdot \underline{u}^n)) w d\underline{x} = \int_{\Omega} ((\Delta t_n) (q/\rho)^n + (\phi c_t)^n p^{n-1}) w d\underline{x} \quad (4.46)$$

Para $0 < h < 1$, sea K_h una partición de Ω en elementos, con tamaño de malla máximo h . Para discretizar la formulación anterior, son necesarios dos espacios de funciones discretos $\underline{V}_h \subset \underline{V}$ y $W_h \subset W$ asociados con la partición K_h , para formar un espacio de funciones mixto $\underline{V}_h \times W_h \subset \underline{V} \times W$, entonces considerando las ecuaciones (4.42) y (4.46), se obtiene la forma bilineal y lineal de la formulación variacional mixta:

Para cualquier $0 \leq n \leq N$, encontrar $\underline{u}_h^n \in \underline{V}_h$ y $p_h^n \in W_h$ tal que

$$a[(\underline{u}_h^n, p_h^n), (\underline{v}, w)] = L[(\underline{v}, w)], \quad \forall (w, \underline{v}) \in W_h \times \underline{V}_h \text{ y } t_n \in (t_0, t_N]. \quad (4.47)$$

donde

$$a[(\underline{u}_h^n, p_h^n), (\underline{v}, w)] = \int_{\Omega} ((\underline{\mu} \underline{k}^{-1} \cdot \underline{u}_h^n) \cdot \underline{v} - p_h^n \nabla \cdot \underline{v} + (\phi c_t)^n p_h^n w + \Delta t_n (\nabla \cdot \underline{u}_h^n) w) d\underline{x} \quad (4.48)$$

$$L[(\underline{v}, w)] = \int_{\Omega} ((\phi c_t)^n p^{n-1} + \Delta t_n q^n / \rho^n) w d\underline{x} - \int_{\Gamma_3} p_{out} (\underline{v} \cdot \underline{n}) ds \quad (4.49)$$

Una elección estable de espacios de elementos finitos es para \underline{V}_h los elementos de *Brezzi-Douglas-Marini* de orden polinomial k y para W_h los elementos *Discontinuos-Galerkin* de orden polinomial $k - 1$ [18].

4.3.3. Modelo numérico con la formulación mixta-dual

Reescribiendo el modelo matemático (ver ecuaciones 4.17-4.20) como un sistema de dos ecuaciones diferenciales en términos de la presión p y de la velocidad \underline{u} , considerando que la densidad es constante y despreciando la fuerza de gravedad de la siguiente manera:

$$\phi c_t \frac{\partial p}{\partial t} + \nabla \cdot \underline{u} = q / \rho \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (4.50)$$

$$\underline{u} = -\frac{1}{\underline{\mu} \underline{k}} \cdot \nabla p \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (4.51)$$

$$p = p_0 \quad , \quad \forall \underline{x} \in \Omega \text{ y } t = t_0 \quad (4.52)$$

$$p = p_{\partial} \quad , \quad \forall \underline{x} \in \partial_D \Omega \text{ y } t > t_0 \quad (4.53)$$

$$-\underline{u} \cdot \underline{n} = g_{\partial} \quad , \quad \forall \underline{x} \in \partial_N \Omega \text{ y } t > t_0 \quad (4.54)$$

Por un lado, de la ecuación (4.51), se obtiene que

$$\underline{u} + \frac{1}{\underline{\mu} \underline{k}} \cdot \nabla p = 0 \quad (4.55)$$

haciendo un producto interior con \underline{v} e integrando sobre el dominio, se obtiene

$$\int_{\Omega} \left(\underline{u} \cdot \underline{v} + \left(\frac{1}{\underline{\mu} \underline{k}} \cdot \nabla p \right) \cdot \underline{v} \right) d\underline{x} = 0 \quad (4.56)$$

Por otro lado, aplicando diferencias finitas hacia atrás en el tiempo, se puede reescribir la ecuación (4.50), para un tiempo t_n dado, como sigue:

$$(\phi c_t)^n \frac{p^n - p^{n-1}}{\Delta t_n} + \nabla \cdot \underline{u}^n = (q / \rho)^n \quad (4.57)$$

donde $p^n \equiv p(\underline{x}, t_n)$ y el tamaño de paso del tiempo es $\Delta t_n \equiv t_n - t_{n-1}$ con $n = 0, 1, \dots, N$.

Multiplicando la ecuación (4.57) por una función de peso w e integrando sobre el dominio Ω y reacomodando términos, se obtiene lo siguiente

$$\int_{\Omega} (\phi_{c_t})^n p^n w d\underline{x} + \Delta t_n \int_{\Omega} (\nabla \cdot \underline{u}^n) w d\underline{x} = \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi_{c_t})^n p^{n-1}) w d\underline{x} \quad (4.58)$$

integrando por partes la segunda integral de la izquierda de la igualdad, después aplicando la condición de frontera de tipo Neumann (4.54) y reacomodando términos, se obtiene

$$\int_{\Omega} (\phi_{c_t})^n p^n w d\underline{x} - \Delta t_n \int_{\Omega} \underline{u}^n \cdot \nabla w d\underline{x} \quad (4.59)$$

$$= \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi_{c_t})^n p^{n-1}) w d\underline{x} + \Delta t_n \int_{\partial_N \Omega} w g_{\partial}^n ds \quad (4.60)$$

Considerando la indexación convenida de las fronteras (ver fig. 4.1), se establece una condición de frontera tipo Dirichlet en la frontera de salida Γ_3 , donde $p(\underline{x}, t) = p_{out}(\underline{x}, t)$. Por otro lado, sea $\partial_N \Omega = \Gamma_1 \cup \Gamma_2$, donde Γ_2 es la superficie lateral curva del cilindro, $g_1 = u_{in}$ en Γ_1 con $u_{in} = cte.$ y $g_2 = 0$ en Γ_2 , entonces

$$\int_{\partial_N \Omega} w g_{\partial}^n d\underline{x} = \int_{\Gamma_1} w g_1^n d\underline{x} + \int_{\Gamma_2} w g_2^n d\underline{x} = \int_{\Gamma_1} w u_{in}^n d\underline{x} \quad (4.61)$$

Comparado con la formulación mixta, aquí la condición de frontera de tipo Dirichlet es una condición esencial y las condiciones de frontera de tipo Neumann son condiciones naturales. Sustituyendo la ecuación (4.61) en la ecuación (4.60), se obtiene lo siguiente

$$\int_{\Omega} (\phi_{c_t})^n p^n w d\underline{x} - \Delta t_n \int_{\Omega} \underline{u}^n \cdot \nabla w d\underline{x} \quad (4.62)$$

$$= \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi_{c_t})^n p^{n-1}) w d\underline{x} + \Delta t_n \int_{\Gamma_1} w u_{in}^n d\underline{x} \quad (4.63)$$

Considerando las ecuaciones (4.56) y (4.63) se obtiene la formulación variacional mixta dual: Encontrar $\underline{u}^n \in \underline{V}$ y $p^n \in W$ tales que satisfacen

$$a[(\underline{u}^n, p^n), (\underline{v}, w)] = L_n[(\underline{v}, w)] \quad \forall (\underline{v}, w) \in \underline{V} \times W \quad (4.64)$$

donde

$$a[(\underline{u}^n, p^n), (\underline{v}, w)] = \int_{\Omega} \left(\underline{u}^n \cdot \underline{v} + \left(\frac{1}{\mu} \underline{k}^n \cdot \nabla p^n \right) \cdot \underline{v} + (\phi_{c_t})^n p^n w - (\Delta t_n) \underline{u}^n \cdot \nabla w \right) d\underline{x} \quad (4.65)$$

$$L_n[(\underline{v}, w)] = \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi_{c_t})^n p^{n-1}) w d\underline{x} + \Delta t_n \int_{\Gamma_1} w u_{in}^n d\underline{x} \quad (4.66)$$

Para $0 < h < 1$, sea K_h una partición de Ω en elementos, con tamaño de malla máximo h . Para discretizar la formulación anterior, se necesitan dos espacios funcionales discretos

$\underline{V}_h \subset \underline{V}$ y $W_h \subset W$ asociados a la partición K_h para formar un espacio de funciones mixto $\underline{V}_h \times W_h \subset \underline{V} \times W$.

Entonces, la formulación mixta dual se puede leer de la siguiente manera:

Para cualquier $0 \leq n \leq N$, encontrar $\underline{u}_h^n \in \underline{V}_h$ y $p_h^n \in W_h$ tales que

$$a[(\underline{u}_h^n, p_h^n), (\underline{v}, w)] = L_n[(\underline{v}, w)] \quad \forall (\underline{v}, w) \in \underline{V}_h \times W_h \quad y \quad t_n \in (t_0, t_N]. \quad (4.67)$$

$$a[(\underline{u}_h^n, p_h^n), (\underline{v}, w)] = \int_{\Omega} \left(\underline{u}_h^n \cdot \underline{v} + \left(\frac{1}{\mu} k^n \cdot \nabla p_h^n \right) \cdot \underline{v} + (\phi c_t)^n p_h^n w - (\Delta t_n) \underline{u}_h^n \cdot \nabla w \right) d\underline{x} \quad (4.68)$$

$$L_n[(\underline{v}, w)] = \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi c_t)^n p^{n-1}) w d\underline{x} + \Delta t_n \int_{\Gamma_1} w u_{in} ds \quad (4.69)$$

Una elección estable de espacios de elementos finitos es para \underline{V}_h los elementos discontinuos de *Raviart-Thomas* de orden polinomial k y para W_h los elementos de *Lagrange* de orden polinomial $k + 1$ [18].

4.4. Modelo computacional

La implementación computacional del modelo de flujo monofásico se realizó con el proyecto FEniCS 2017.1, el cual es una biblioteca de código abierto, bajo la plataforma de Python 2.7, en el sistema operativo Linux Ubuntu 16.04, en una computadora Lenovo Think Pad T420 de 64 bits con procesador quad-core de 2.5 GHz y 8 GB de memoria RAM.

Se va a mostrar el modelo computacional considerando las hipótesis generales, discretización de la malla y del dominio, condiciones de frontera y su implementación en un código computacional. Para observar la forma que toma la solución numérica, se considera un núcleo con las siguientes propiedades (ver cuadro 4.1):

símbolo	Descripción	Valor	Unidad
p_o	Presión inicial	80	psi
r_w	Radio del núcleo	0.02	m
r_e	Altura del núcleo	0.25	m
μ	Viscosidad del aceite	1.31E-02	$Pa \cdot s$
k	Permeabilidad absoluta	1.51E-13	m^2
c_o	Compresibilidad del aceite (c_f)	0.00001	$1/psi$
c_R	Compresibilidad de la roca	0.000004	$1/psi$
ϕ	Porosidad	0.1978	fracción
Q	Tasa de inyección	1.39E-09	m^3/s
ρ_o	Densidad del aceite	8.72E+02	Kg/m^3

Cuadro 4.1: Valores de las propiedades del núcleo, del caso de estudio del modelo de flujo monofásico ligeramente compresible.

Bibliotecas

Primero se importan las bibliotecas que se utilizarán en la parte computacional y que son necesarias para implementar la solución computacional.

```
from fenics import *
from mshr import *
import numpy as np
import matplotlib.pyplot as plt
```

Mshr es el componente de generación de malla de FEniCS. Genera mallas DOLFIN simpliciales en 2D y 3D a partir de geometrías descritas por CSG (Constructive Solid Geometry) o de archivos de superficie.

NumPy es el paquete fundamental para la computación científica con Python. Contiene entre otras cosas: Un potente objeto de matriz N-dimensional, sofisticadas funciones de radiodifusión, herramientas para integrar código C/C++ y Fortran, álgebra lineal útil, transformada de Fourier y capacidades de números aleatorios.

Matplotlib es una biblioteca para graficar de Python que produce figuras de calidad de publicación en una variedad de formatos impresos y entornos interactivos a través de plataformas. **Matplotlib.pyplot** es una colección de funciones de estilo de comando que hacen que matplotlib funcione como MATLAB. Cada función pyplot hace algún cambio a una figura: por ejemplo, crea una figura, crea un área de trazado en una figura, traza algunas líneas en un área de trazado, decora la trama con etiquetas, etc. En matplotlib.pyplot se conservan varios estados a través de la función `plt`, De modo que se mantenga al tanto de cosas como la figura actual y el área de trazado, y las funciones de trazado se dirigen a los ejes actuales.

Dominio

El dominio del modelo de flujo monofásico es un cilindro de longitud $l = 0.25[m]$ y diámetro $d = 0.02[m]$, como el que se muestra en la figura (4.1). Entonces el dominio computacional lo tomaremos de la siguiente manera:

```
l=0.25 # core length L=0.08 [m]
d=0.02 # core diameter d=0.038 [m]
r=d/2 # core radius
xmin=0.0
xmax=d
ymin=0.0
ymax=d
zmin=0.0
zmax=l
domain=Cylinder(dolfin.Point(xmin,ymin,zmin),dolfin.Point(xmin,ymin,zmax),r,r)
```

Mallado

```
mesh = generate_mesh(domain, 120)
```



Figura 4.2: Mallado del dominio del modelo de flujo monfásico generado con FEniCS.

Fronteras

La numeración de las fronteras se colocaron de acuerdo a la indexación convenida en la figura (4.1) de la siguiente manera:

```
boundary_parts = FacetFunction("size-t", mesh)

#Boundary condition at the bottom

class BottomBoundary(SubDomain):
    def inside(self, x, on_boundary):
        tol = 1E-14 # tolerance for coordinate comparisons
        return on_boundary and abs(x[2]) < tol

Gamma_1 = BottomBoundary()
Gamma_1.mark(boundary_parts, 1)

#Boundary condition at the top

class TopBoundary(SubDomain):
    def inside(self, x, on_boundary):
        tol = 1E-14 # tolerance for coordinate comparisons
        return on_boundary and abs(x[2] - zmax) < tol

Gamma_3 = TopBoundary()
Gamma_3.mark(boundary_parts, 3)
```

Factores de conversión de unidades

```
millidarcy=1e-15 # [m^2]
psi=6894.757 # [Pa]
```

Parámetros del problema

```

R = phi*c_T
gamma = 0 # Gravity
phi=0.1978 # porosity
c_R=0.000004/psi #Rock compressibility [1/Pa]
c_f=0.00001/psi #Fluid compressibility [1/Pa]
c_T=c_R + c_f #Total compressibility [1/Pa]
mu=0.0131 # Fluid viscosity [Pa*s]
rho=872 # Fluid density [Kg/m3]

q=0 # Source term
p0= 80*psi # Initial pressure [Pa]
u_in=1.39e-9 # Injection velocity [m/s]
p_out=p0 # Production pressure [Pa]

#Time
t_0=0.0 # [s] #Initial time
t_N=100 # [s] #End time
dt = 1.0 # Time step
NT=int((t_N-t_0)/dt) #Number of time steps

# Permeability
k=151*millidarcy

# Hydrodynamic dispersion tensor

K_xx= k # [mD]
K_xy=0.0 # [mD]
K_xz=0.0 # [mD]

K_yx=0.0 # [mD]
K_yy= k # [mD]
K_yz=0.0 # [mD]

K_zx=0.0 # [mD]
K_zy= 0.0 # [mD]
K_zz= k # [mD]

K = as_matrix(((K_xx,K_xy,K_xz),(K_yx,K_yy,K_yz),(K_zx,K_zy,K_zz)))

Kinv = as_matrix(((1.0/k,0.0,0.0),(0.0,1.0/k,0.0),(0.0,0.0,1.0/k)))

# boundary normal vector
n = FacetNormal(mesh)

```

Cuadratura

```

ffc_parameters = {"quadrature_degree": fs_order + 1, "representation": "
quadrature"}

```

4.4.1. Implementación con la formulación clásica

Espacios de funciones

```
fs_order = 2 # function space order
V = FunctionSpace(mesh, 'Lagrange', fs_order) # for the pressure
U = VectorFunctionSpace(mesh, 'Lagrange', fs_order-1) # for the velocity
```

Fronteras naturales y esenciales

Se definieron de acuerdo a la indexación convenida en la figura (4.1) como sigue:

```
p_B = Constant(u_in)
bc1=DirichletBC(V, p_B, boundary_parts, 1)
p_T = Constant(p_out)
bc3=DirichletBC(V, p_T, boundary_parts, 3)
bcs = [bc3]
```

Funciones de base y de peso

```
p = TrialFunction(V)
w = TestFunction(V)
```

Formulación variacional clásica

```
a = R*p*w*dx + dt*inner(nabla_grad(w), dot(K/mu, nabla_grad(p)))*dx
L = (R*p_1 + dt*q/rho)*w*dx + dt*u_in*w*ds(1)
```

Condición inicial

```
p_0 = Expression('p0', p0=p0, degree=fs_order + 1)
p_1 = project(p_0, V)
```

Solución en el tiempo

```
A = assemble(a)
b = None
p = Function(V) # the unknown at a new time level
t = t_0
while t < t_N:
    b = assemble(L, tensor=b)
    p_0.t = t
    for bc in bcs:
        bc.apply(A, b) #if Dirichlet conditions
    solve(A, p.vector(), b, "lu")
    t += dt
    p_1.assign(p)
    p_h = project(p)
```

```
u = project(-dot(K/mu, nabla_grad(p)), U) # Darcy Velocity
u_x, u_y, u_z = u.split(deepcopy=True) # extract components
```

4.4.2. Implementación con la formulación mixta

Espacios de funciones

```
BDM = FiniteElement("BDM", 'triangle', fs_order+1)
Q = FiniteElement("DG", 'triangle', fs_order)
mixed_element = MixedElement([BDM, Q])
W = FunctionSpace(mesh, mixed_element)
```

Fronteras naturales y esenciales

```
u_B = Constant((u_x_in, u_y_in, u_z_in)) #Constant flow
bc1=DirichletBC(W.sub(0), u_B, boundary_parts, 1)

u_R = Constant((0, 0, 0)) #No flow
bc2=DirichletBC(W.sub(0), u_R, boundary_parts, 2)

p_T = Constant(p_out)
bc3=DirichletBC(W.sub(1), p_T, boundary_parts, 3)

bcs = [bc1, bc2]
```

Funciones de base y de peso

```
u, p = TrialFunctions(W)
v, w = TestFunctions(W)
```

Formulación variacional

```
a = inner(mu*Kinv*u, v)*dx - div(v)*p*dx + R*p*w*dx + dt*div(u)*w*dx
L = (R*p1 + dt*q/rho)*w*dx - p_T*inner(v,n)*ds(3)
```

Condiciones iniciales

```
p_0= 80*psi # Initial pressure [Pa]
u_0= 0 # Initial velocity [m/s]
u_x_in=0 # Injection velocity [m/s]
u_y_in=0 # Injection velocity [m/s]
u_z_in=1.39e-9 # Injection velocity [m/s]

up0 = Expression(('0.0', '0.0',, '0.0', 'pp'), pp=p_0, degree=fs_order)
up1.interpolate(up0)
```


Solución en el tiempo

```

up = Function(W) # Define functions for solutions at previous time steps
up1 = Function(W) # Define functions for solutions at current time steps

u1, p1 = split(up1) # Split system functions to access components

t = t_0
i=0

while t < t_N:
    solve(a == L, up, bcs)
    (u, p) = up.split()
    t += dt
    i=i+1
    up1.assign(up)

```

4.4.3. Implementación con la formulación mixta-dual

Espacios de funciones

```

DRT = FiniteElement("DRT", 'triangle', fs_order)
CG = FiniteElement("CG", 'triangle', fs_order+1)
mixed_element = MixedElement([DRT, CG])
MW = FunctionSpace(mesh, mixed_element)

```

Fronteras naturales y esenciales

```

u_B = Constant([ux_in, uy_in, uz_in]) #Constant flow
bc1=DirichletBC(MW.sub(0), u_B, boundary_parts, 1)

u_R = Constant((0, 0, 0)) #No flow
bc2=DirichletBC(MW.sub(0), u_R, boundary_parts, 2)

p_T = Constant(p_out)
bc3=DirichletBC(MW.sub(1), p_T, boundary_parts, 3)

bcs = [bc3]

```

Funciones de base y de peso

```

(u, p) = TrialFunctions(MW)
(v,w) = TestFunctions(MW)
dup = TrialFunction(MW)

```

Formulación variacional

```

a1 = (dot(u, v) + dot(dot(K/mu, nabla_grad(p)), v))*dx
a2 = (R*p*w - d_t*dot(u, nabla_grad(w)))*dx
a = a1 + a2
L = (R*p_l + d_t*q/rho)*w*dx + d_t*uy_in*w*ds(1)
F = a-L

```

Condiciones iniciales

```

p_0= Constant(80.0*psi) # Initial pressure [Pa]
u_0= 0.0 # Initial velocity [m/s]
ux_in=Constant(0.0) # Injection velocity [m/s]
uy_in=Constant(0.0) # Injection velocity [m/s]
uz_in=Constant(1.39e-9) # Injection velocity [m/s]

up0 = Expression(('0.0', '0.0', '0.0', '80.0*pp'), pp=psi, degree=fs_order)
up1.interpolate(up0)

```

Solución en el tiempo

```

up = Function(MW) # current solution
up1 = Function(MW) # solution from previous converged step

# Split system functions to access components
u, p = split(up)
u_l, p_l = split(up1)
du, dp = split(dup)

dF = derivative(F, up, dup)
problem = NonlinearVariationalProblem(F, up, bcs, dF)
solver = NonlinearVariationalSolver(problem)
solver.parameters['newton_solver']['absolute_tolerance'] = 1E-16
solver.parameters['newton_solver']['relative_tolerance'] = 1E-16

t = t_0
i=0

while t < t_N:
    solver.solve()
    (_u, _p) = up.split()
    t += dt
    i=i+1
    up1.assign(up) # Update previous solution

```

4.5. Resultados

La principal motivación de la presente cap fue desarrollar un modelo matemático, numérico y computacional para flujo monofásico de fluidos ligeramente compresibles en medios porosos a escala de laboratorio.

La metodología que se utilizó para llevar a cabo dicho desarrollo consistió primero en plantear un modelo conceptual, en el cual se establecieron las hipótesis, los alcances y las limitaciones del modelo. Posteriormente, aplicando la formulación sistemática de la modelación de medios continuos, se obtuvo un modelo matemático, obteniendo un sistema de ecuaciones diferenciales parciales con condiciones iniciales y de valores en la frontera. Después, se desarrolló una versión discretizada del modelo matemático (Formulación Variacional), con el método de elementos finitos. Finalmente se realizó su implementación computacional mediante el software FEniCS project 2017.1 bajo la plataforma de Python 2.7. Las soluciones numéricas son comparadas con las obtenidas con el software comercial COMSOL Multiphysics. Se muestra un caso de estudio en un cilindro vertical donde se inyecta agua en la parte inferior y se produce en la parte superior.

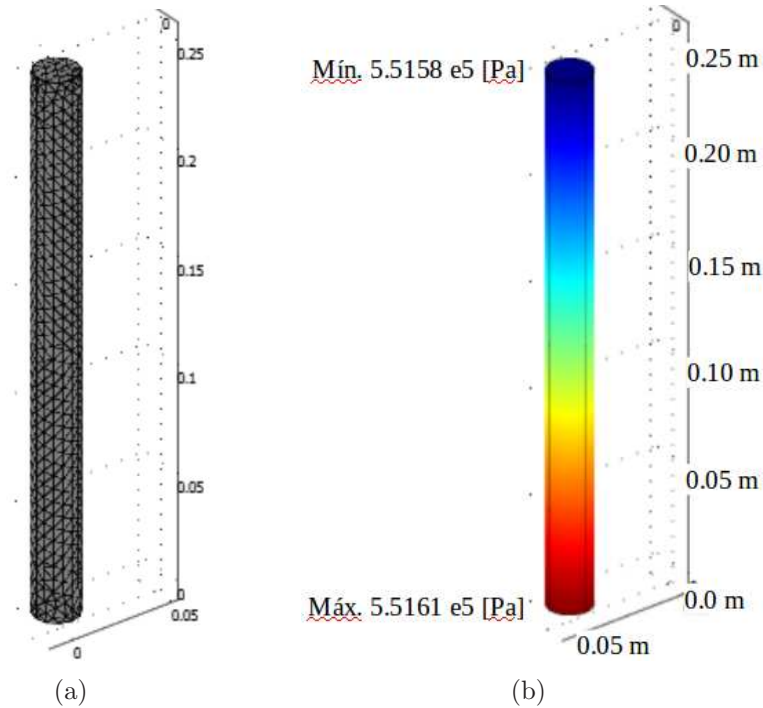


Figura 4.3: (a) Malla irregular del dominio generada con COMSOL Multiphysics con 5971 grados de libertad, 3304 elementos tetraédricos Lagrange cuadráticos, 969 puntos de malla. (b) Distribución final de la presión a lo largo del núcleo del modelo de flujo monfásico al cabo de 100 segundos, obtenida con COMSOL Multiphysics.

En la figura 4.3 se puede observar que la mayor presión se encuentra en la entrada del núcleo donde se está inyectando el fluido, mientras que en la salida del núcleo donde hay una producción constante, se aprecia una menor presión.

En la figura (4.4) se muestran los perfiles del cálculo numérico de la evolución de la presión en cada segundo a lo largo de la altura del núcleo, durante un periodo de tiempo

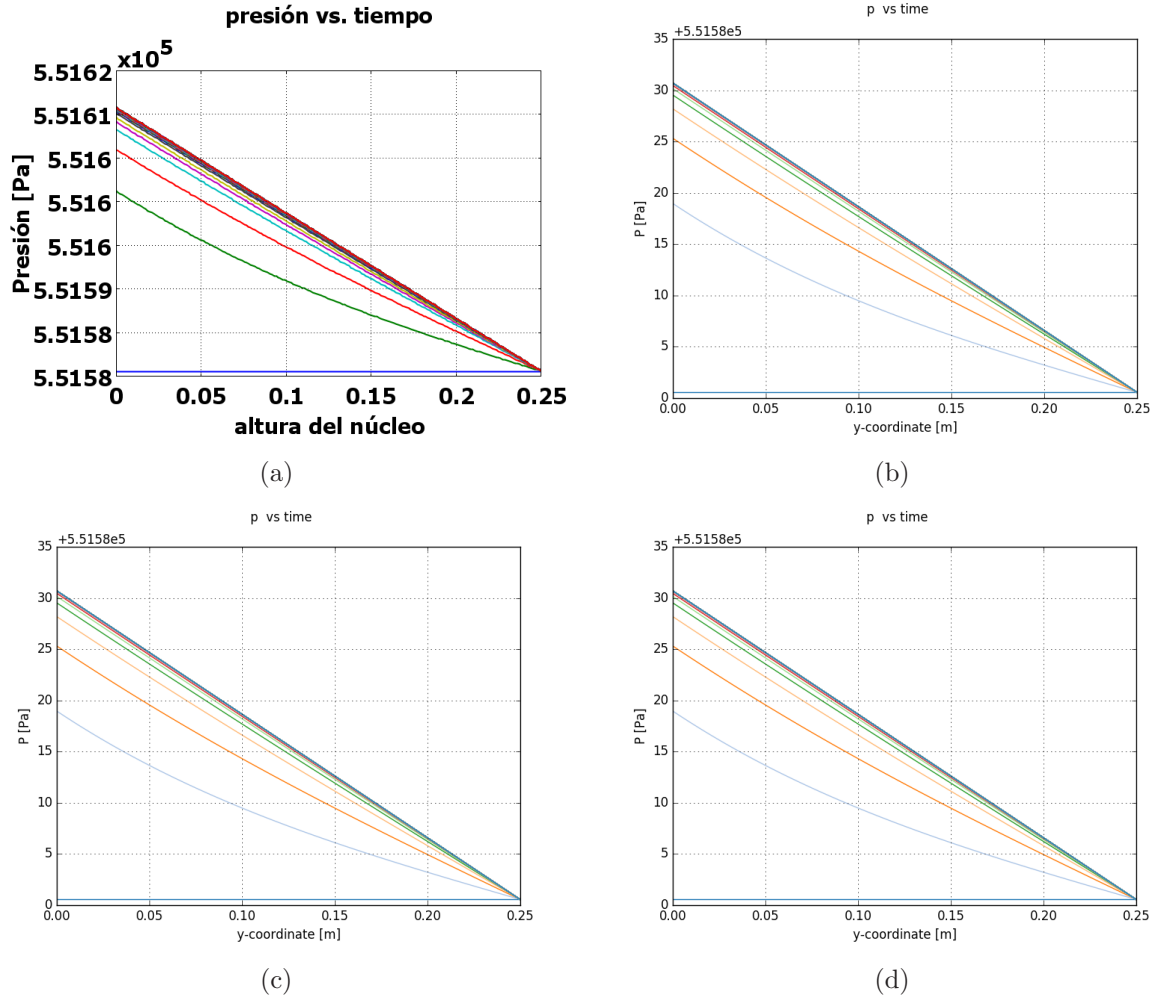


Figura 4.4: Perfiles de presión a lo largo de la altura del núcleo tomado cada segundo con: (a) COMSOL Multiphysics (b) FEniCS con MEF Clásico (c) FEniCS con MEF Mixto (d) FEniCS con MEF Mixto-Dual.

de 100 segundos. La forma de las curvas en dicha figura se deben a que la mayor presión se encuentra en la entrada del núcleo donde se está inyectando el fluido, mientras que en la salida del núcleo, donde hay una producción constante, se aprecia una menor presión, comenzando con una presión inicial constante de 80 [psi] equivalente a 551580.8 [Pa], que al evolucionar cada segundo, se puede apreciar su estabilización a los pocos segundos.

Se puede observar también que las gráficas de los perfiles de la presión son muy similares tanto en COMSOL Multiphysics como con FEniCS 2017 con los diferentes métodos de elementos finitos (clásico, mixto y mixto-dual) descritos anteriormente, sin embargo existen ciertas diferencias en los cálculos numéricos como se podrá apreciar en la sección 4.6.

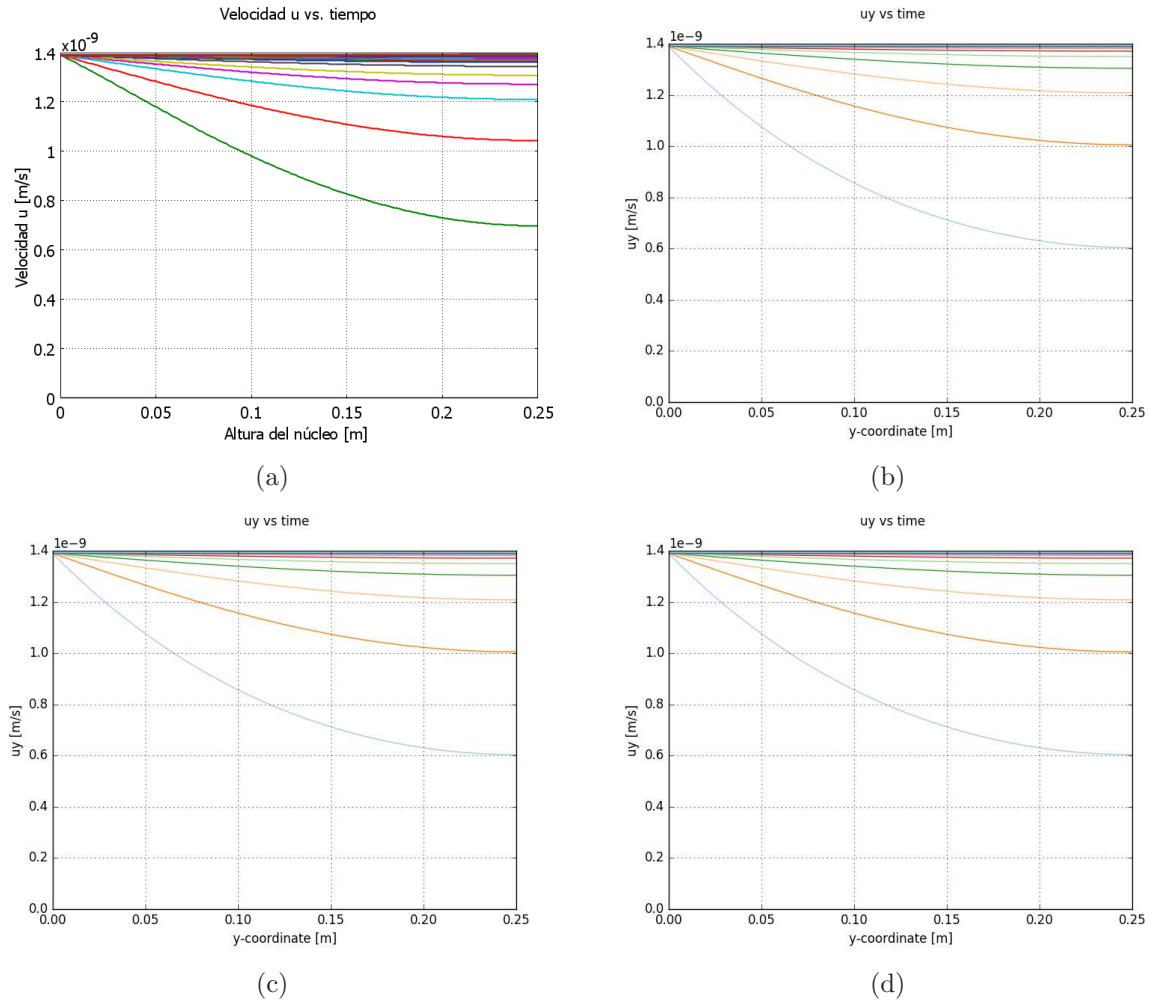


Figura 4.5: Perfiles de la velocidad a lo largo de la altura del núcleo cada segundo con: (a) COMSOL Multiphysics (b) FEniCS con MEF Clásico (c) FEniCS con MEF Mixto (d) FEniCS con MEF Mixto-Dual.

En la figura (4.5) se muestran los perfiles del cálculo numérico de la evolución de la velocidad del fluido obtenida con la ley de Darcy, en cada segundo a lo largo de la altura del núcleo, durante un periodo de tiempo de 100 segundos. La forma de las curvas en dicha figura se deben a que en la entrada del núcleo se está inyectando el fluido a una tasa constante,

mientras que en la salida del núcleo, donde hay una producción constante, se aprecian los cambios de la velocidad que decae linealmente.

Se puede observar también que las gráficas de los perfiles de la velocidad son muy similares tanto en COMSOL Multiphysics como con FEniCS 2017.1 con los tres diferentes métodos de elementos finitos (clásico, mixto y mixto-dual) descritos anteriormente, sin embargo existen ciertas diferencias en los cálculos numéricos como se podrá apreciar en la sección 4.6.

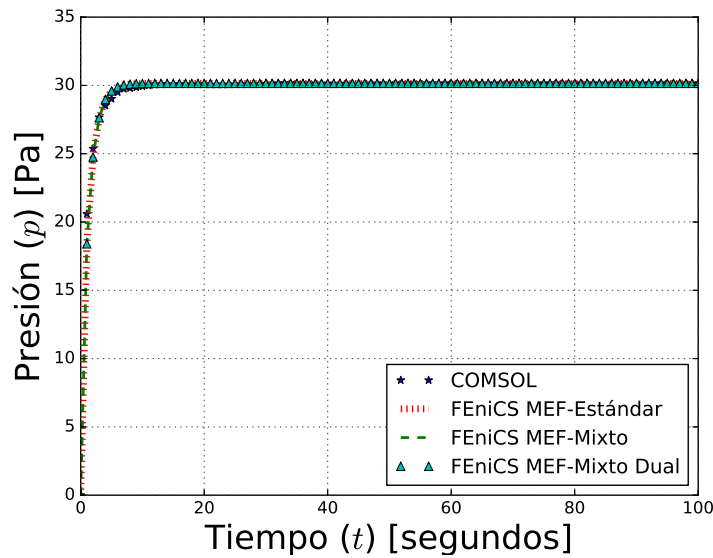


Figura 4.6: Cambio de la presión en la entrada y en la salida del núcleo con respecto al tiempo obtenido con COMSOL y los diferentes métodos implementados en FEniCS del modelo de flujo monofásico.

En la figura 4.6 se pueden apreciar las gráficas de la diferencia de presión en la entrada y en la salida del núcleo con respecto al tiempo, obtenidas tanto con FEniCS como con COMSOL Multiphysics, para visualizar su comparación entre ambas gráficas.

Las simulaciones realizadas tanto con FEniCS como con COMSOL muestran un comportamiento muy similar, además se comportan de una manera muy coherente con las condiciones impuestas en las fronteras. La ventaja de usar el software libre FEniCS es que podemos implementar los modelos matemáticos hechos a la medida de las condiciones específicas de nuestro problema físico particular con una gran aproximación numérica a través del método de elementos finitos.

Un método común para probar la implementación de un método numérico es comparar la solución numérica con una solución analítica exacta del problema de la prueba y concluir que el programa funciona si el error es "lo suficientemente pequeño". Esta es una técnica de verificación muy potente y se explica en detalle en la sección 4.6.

4.6. Análisis de las tasas de convergencia

Una cuestión central para cualquier método numérico es su tasa de convergencia: La rapidez con la que el error se aproxima a cero cuando se aumenta la resolución. Para los métodos de elementos finitos, esto corresponde a probar, que el error $E = p_e - p$ está limitado por el tamaño de malla h a alguna potencia m ; es decir, $\|E\| \leq Ch^m$ para alguna constante C . Al número m se le conoce como *la tasa de convergencia* del método.

Definiendo el tamaño del elemento como $h = 1/n$, donde n es el número de divisiones en la dirección de x o y . Todos los ejemplos se resolvieron para una partición uniforme del cuadrado unitario. El número de elementos n se tomó igual en ambas direcciones ($n = n_x = n_y$) y se incrementó sucesivamente $n = 4, 8, 16, 32, 64, 128, 256$. Se realizaron experimentos con $h_0 > h_1 > h_2 > \dots$ y se calcularon los errores correspondientes E_0, E_1, E_2, \dots

Suponiendo que $E_i = Ch_i^m$ para constantes desconocidas C y m , podemos comparar dos experimentos consecutivos $E_{i-1} = Ch_{i-1}^m$ y $E_i = Ch_i^m$, de donde se obtiene que

$$\frac{E_{i-1}}{h_{i-1}^m} = C = \frac{E_i}{h_i^m} \quad (4.70)$$

aplicando la función logaritmo de ambos lados de la igualdad

$$\ln\left(\frac{E_{i-1}}{h_{i-1}^m}\right) = \ln\left(\frac{E_i}{h_i^m}\right) \quad (4.71)$$

aplicando las propiedades de los logaritmos

$$\ln(E_{i-1}) - m\ln(h_{i-1}) = \ln(E_i) - m\ln(h_i) \quad (4.72)$$

reacomodando términos y factorizando la constante m

$$m(\ln(h_i) - \ln(h_{i-1})) = \ln(E_i) - \ln(E_{i-1}) \quad (4.73)$$

aplicando nuevamente las propiedades de los logaritmos

$$m\ln(h_i/h_{i-1}) = \ln(E_i/E_{i-1}) \quad (4.74)$$

finalmente se deduce m

$$m = \frac{\ln(E_i/E_{i-1})}{\ln(h_i/h_{i-1})} \quad (4.75)$$

Los valores de m se aproximan a la tasa de convergencia esperada conforme i aumenta.

Para ilustrar los cálculos de la tasa de convergencia del error de las soluciones en los tres tipos de métodos de elementos finitos, los experimentos numéricos consistieron en resolver la ecuación elíptica general de segundo orden (4.76) con condiciones de frontera de tipo Dirichlet impuestas por la solución analítica, usando el método de elementos finitos con polinomios lineales, cuadráticos y cúbicos. Será prudente construir un problema con una solución analítica conocida para que podamos comprobar fácilmente que la solución calculada

es correcta. Consideremos el siguiente problema con valores en la frontera, sobre el cuadrado unitario:

$$-\nabla \cdot (\underline{a} \cdot \nabla p) + \nabla \cdot (\underline{b}p) + cp = f, \quad \forall \underline{x} \in [0, 1] \times [0, 1] \quad (4.76)$$

$$p(\underline{x}) = 0, \quad \forall \underline{x} \in \partial[0, 1] \times [0, 1] \quad (4.77)$$

donde

$$\underline{a} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \quad \underline{b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}; \quad c = 0 \quad (4.78)$$

Para mostrar el cálculo de las tasas de convergencia, se elige una solución exacta $p_e \in C^\infty$:

$$p_e(x, y) = \sin(\pi x) \sin(\pi y) \quad (4.79)$$

Al sustituir la ecuación (4.79) en la ecuación (4.76), esta elección implica lo siguiente:

$$f(x, y) = 2\pi^2 \sin(\pi x) \sin(\pi y) \quad (4.80)$$

Este método sencillo pero muy poderoso para construir problemas de prueba, se llama *método de soluciones fabricadas*: se elige una expresión simple para la solución exacta, se sustituye en la ecuación para obtener el lado derecho (término fuente f), luego se resuelve la ecuación con este lado derecho y usando la condición de frontera, se trata de reproducir la solución exacta.

Para el modelo numérico de la ecuación elíptica, se desarrolló la formulación variacional y se realizó la implementación computacional mediante el lenguaje de Programación Python, utilizando la biblioteca **FEniCS** 2017.1.

Se realizó la comparación de la convergencia de los métodos de elementos finitos en términos del error medido con la norma L_2 , la cual se define como

$$Error = \left(\int_{\Omega} (p_e - \tilde{p})^2 dx \right)^{1/2}, \quad \forall \underline{x} \in \Omega \quad (4.81)$$

donde p_e es la solución exacta y \tilde{p} es la solución numérica aproximada.

Se lograron resultados interesantes, usando la norma L_2 de la diferencia de los grados de libertad, se obtuvo el siguiente cuadro:

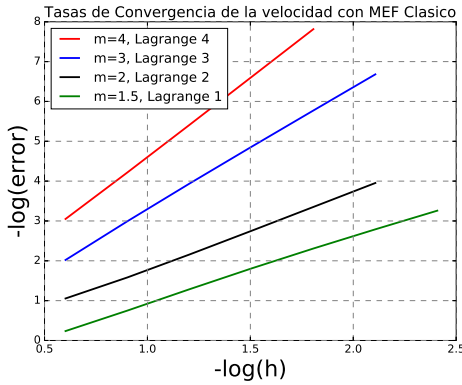
Elemento	$n = 8$	$n = 16$	$n = 32$	$n = 64$
P ₁	1.83	1.96	1.99	2.0
P ₂	3.08	3.03	3.01	3.0
P ₃	4.02	4.02	4.01	4.0
P ₄	4.98	4.99	5.0	5.0

Cuadro 4.2:

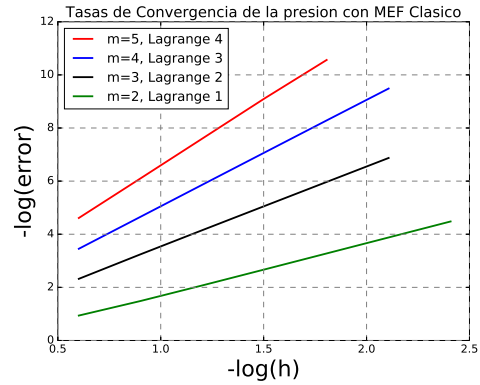
En el cuadro (4.2), una entrada como 1.99 para $n = 32$ y P₁ significa que se ha estimado la tasa de convergencia $m = 1.99$ comparando dos mallas, con resoluciones $n = 32$ y $n = 16$,

utilizando elementos P_1 (Lagrange de orden 1). Se puede observar la superconvergencia para P_2 en los nodos. Las mejores estimaciones de las tasas aparecen en la columna más a la derecha, ya que estas tasas se basan en las resoluciones más finas y por lo tanto más profundas en el régimen asintótico (hasta llegar a un nivel donde los errores de redondeo y la solución aproximada del sistema lineal comienza interpretar un papel).

El orden m del error con respecto a h , es decir $O(h^m)$ se obtuvo a partir de la estimación de la pendiente de la regresión lineal en la gráfica de $-\log(\text{error})$ contra $-\log(h)$, para diferentes valores sucesivamente menores de h .



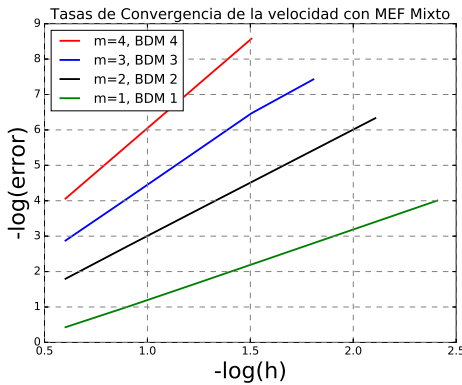
(a) Análisis de la tasa de la convergencia del MEF Clásico para la velocidad \underline{u} .



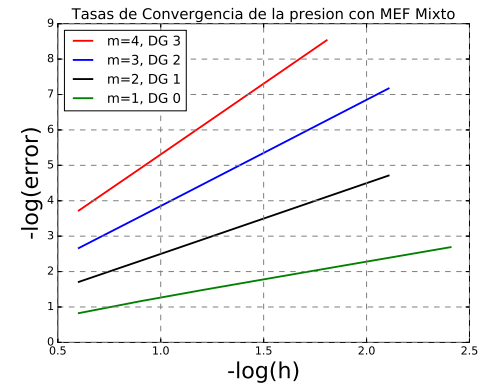
(b) Análisis de la tasa de la convergencia del MEF Clásico para la presión p .

Figura 4.7:

En la figura (4.7), se muestra el análisis de la tasa de la convergencia del Método de Elementos Finitos Clásico, para el error en la velocidad \underline{u} y el error en la presión p , calculados con elementos de Lagrange de orden d . Los errores de la norma L_2 muestran una tasa de convergencia $m = d$ esperada para la velocidad \underline{u} y una tasa de convergencia $m = d + 1$ esperada para la presión p .



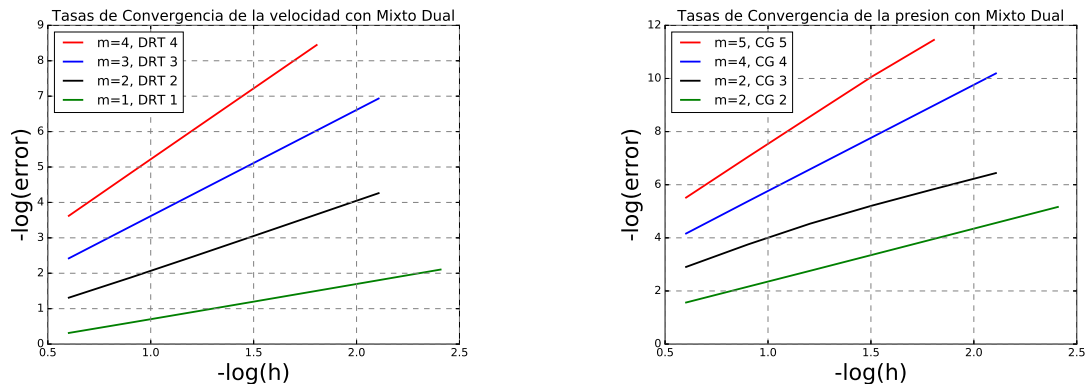
(a) Análisis de la tasa de la convergencia del MEF Mixto para la velocidad \underline{u} .



(b) Análisis de la tasa de la convergencia del MEF Mixto para la presión p .

Figura 4.8:

En la figura (4.8), se muestra el análisis de la tasa de la convergencia del Método de Elementos Finitos Mixto, para: **(a)** el orden del error en la solución numérica aproximada de la velocidad \underline{u} calculada con elementos Brezzi-Douglas-Marini (BDM) de orden d y **(b)** el orden del error en la solución numérica aproximada de la presión p calculada con elementos Galerkin Discontinuos (DG) de orden $d - 1$. Los errores de la norma L_2 muestran una tasa de convergencia $m = d$ esperada para la velocidad \underline{u} y la presión p .



(a) Análisis de la tasa de la convergencia del MEF Mixto-Dual para la velocidad \underline{u} .

(b) Análisis de la tasa de la convergencia del MEF Mixto-Dual para la presión p .

Figura 4.9: Análisis del error con el MEF Mixto-Dual

En la figura (4.9), se muestra el análisis de la tasa de la convergencia del Método de Elementos Finitos Mixto-Dual, para: **(a)** el orden del error en la solución numérica aproximada de la velocidad \underline{u} calculada con elementos Raviart-Thomas Discontinuos (DRT) de orden d y **(b)** el orden del error en la solución numérica aproximada de la presión p calculada con elementos de Galerkin Continuos (CG) de orden $d + 1$. Los errores de la norma L_2 muestran una tasa de convergencia $m = d + 1$ esperada para la velocidad \underline{u} y la presión p .

Capítulo 5

Modelo de transporte multicomponente

En el presente capítulo se desarrolla el modelo de transporte multicomponente en un medio poroso, el cual se puede expresar como un sistema no-lineal de ecuaciones diferenciales parciales parabólicas que se puede resolver numéricamente mediante el método de elementos finitos estándar para discretizar el espacio, mientras que para para discretizar el tiempo se aplica el método de Euler hacia atrás y se derivan las formulaciones variacionales de las ecuaciones de transporte de cada una de las componentes. Finalmente se resuelve el sistema de ecuaciones no-lineales con un método tipo Newton, el cual se implementó computacionalmente con elementos finitos mixtos.

5.1. Modelo conceptual

En el modelo de transporte multicomponente en un medio poroso se van a considerar las siguientes hipótesis generales:

1. Sólo hay una fase fluida (f) y una fase sólida (s). Ambas son ligeramente compresibles.
2. Hay m componentes en la fase fluida (f) y en la fase sólida (s).
3. El medio poroso se considera homogéneo, isótropo y completamente saturado.
4. Algunas componentes pueden pasar de la fase fluida a la fase sólida y viceversa mediante mecanismos de transporte como procesos de adsorción, desorción, advección, difusión, dispersión y reacción.
5. El estado inicial de la porosidad es constante, pero permite la variación dinámica de la porosidad debido a los procesos de taponamiento/destaponamiento.
6. La permeabilidad inicial es constante y es la misma en todas las direcciones, pero se puede modificar en función con la porosidad de los procesos.
7. Velocidad constante y condiciones de frontera generales.

5.2. Modelo matemático

Los modelos de transporte multicomponente en medios porosos desarrollados en el presente trabajo se pueden expresar como un sistema no-lineal de m ecuaciones diferenciales parciales parabólicas generales de segundo orden, donde para cada componente i se tiene

$$R_i \frac{\partial c_i}{\partial t} - \nabla \cdot (\underline{\underline{D}}_i \cdot \nabla c_i) + \nabla \cdot (\underline{u}_i c_i) + r_i c_i = f_i \quad , \quad \forall \underline{x} \in \Omega \text{ y } t \geq t_0 \quad (5.1)$$

$$c_i = c_i^0 \quad , \quad \forall \underline{x} \in \Omega \text{ y } t = t_0 \quad (5.2)$$

$$c_i = c_{i\partial} \quad , \quad \forall \underline{x} \in \partial_D \Omega \text{ y } t > t_0 \quad (5.3)$$

$$-\underline{n} \cdot (\underline{\underline{D}}_i \cdot \nabla c_i) = g_{i\partial} \quad , \quad \forall \underline{x} \in \partial_N \Omega \text{ y } t > t_0 \quad (5.4)$$

donde $\underline{\underline{D}}_i$ es un tensor simétrico, positivo definido, acotado y suave a tramos en Ω de la

forma $\underline{\underline{D}}_i = \begin{pmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{pmatrix}$, con $D_{ij} \equiv D_{ij}(\underline{x}, t)$, $\underline{u}_i \equiv \underline{u}_i(\underline{x}, t)$, mientras que $r_i \equiv r_i(\underline{x}, t)$

y $f_i \equiv f_i(\underline{x}, t)$. Ω es un dominio acotado con frontera $\partial\Omega$, a su vez la frontera está formada por dos partes, una con condiciones de tipo Dirichlet $\partial_D \Omega$ y otra con condiciones de tipo Neumann $\partial_N \Omega$, donde la frontera $\partial\Omega = \partial_D \Omega \cup \partial_N \Omega$, con $\partial_D \Omega \cap \partial_N \Omega = \emptyset$ y c_i es la concentración de la componente i , con $i = 1, 2, 3, \dots, m$.

5.3. Modelo numérico

Como el método numérico a aplicar en este caso es el de elementos finitos, usualmete se prefiere para los casos no estacionarios realizar una semidiscretización en el tiempo de la ecuación diferencial. Aquí se usará por simplicidad y por razones de estabilidad un esquema de diferencias finitas hacia atrás para el término del tiempo en cada componente i :

$$\frac{\partial c_i^n}{\partial t} = \frac{c_i^n - c_i^{n-1}}{\Delta t_n} + O(\Delta t_n) \quad (5.5)$$

donde $c_i^n \equiv c_i(\underline{x}, t_n)$ y el tamaño del paso del tiempo es $\Delta t_n \equiv t_n - t_{n-1}$, con $n = 0, 1, \dots, N$. Entonces para un tiempo t_n dado, se puede reescribir la ecuación (5.1) como sigue:

$$R_i^n \frac{c_i^n - c_i^{n-1}}{\Delta t_n} - \nabla \cdot (\underline{\underline{D}}_i^n \cdot \nabla c_i^n) + \nabla \cdot (\underline{u}_i^n c_i^n) + r_i^n c_i^n = f_i^n \quad (5.6)$$

con $R_i^n \equiv R_i(\underline{x}, t_n)$, $c_i^n \equiv c_i(\underline{x}, t_n)$, $\underline{\underline{D}}_i^n \equiv \underline{\underline{D}}_i(\underline{x}, t_n)$, $\underline{u}_i^n \equiv \underline{u}_i(\underline{x}, t_n)$, $r_i^n \equiv r_i(\underline{x}, t_n)$ y $f_i^n(\underline{x}, t_n)$.

Reacomodando los términos de la ecuación (5.6), se tiene que

$$\left(\frac{R_i^n}{\Delta t_n} + r_i^n \right) c_i^n - \nabla \cdot (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) = \frac{R_i^n}{\Delta t_n} c_i^{n-1} + f_i^n \quad (5.7)$$

Sea $W_i = \{c_i \in H^1(\Omega) | c_i = c_{i\partial}, \quad \forall \underline{x} \in \partial_D \Omega\}$ el espacio de las funciones de base de la componente i y $\hat{W}_i = \{w_i \in H^1(\Omega) | w_i = 0, \quad \forall \underline{x} \in \partial_D \Omega\}$ el espacio de las funciones de peso de la componente i , donde $H^1(\Omega) = \{w \in L^2(\Omega) : \nabla w \in L^2(\Omega)\}$ es el espacio de Sobolev y $L^2(\Omega) = \{w : \int_{\Omega} ||w||^2 d\underline{x} < \infty\}$ es un espacio vectorial normado.

Multiplicando la ec. (5.7) por una función de peso $w_i \in \hat{W}$ e integrando sobre Ω resulta:

$$\int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} + r_i^n \right) c_i^n w_i d\underline{x} - \int_{\Omega} \nabla \cdot (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) w_i d\underline{x} = \int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} c_i^{n-1} + f_i^n \right) w_i d\underline{x} \quad (5.8)$$

integrando por partes la segunda integral de la izquierda de la igualdad de la ecuación (5.8)

$$\begin{aligned} & \int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} + r_i^n \right) c_i^n w_i d\underline{x} + \int_{\Omega} \nabla w_i \cdot (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) d\underline{x} - \int_{\partial\Omega} w_i (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) \cdot \underline{n} d\underline{x} \\ &= \int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} c_i^{n-1} + f_i^n \right) w_i d\underline{x} \end{aligned} \quad (5.9)$$

Por otro lado

$$\int_{\partial\Omega} w_i (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) \cdot \underline{n} d\underline{x} = \int_{\partial_N \Omega} w_i (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) \cdot \underline{n} d\underline{x} + \int_{\partial_D \Omega} w_i (\underline{\underline{D}}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) \cdot \underline{n} d\underline{x} \quad (5.10)$$

como $w_i \in \hat{W}_i$ entonces $w_i = 0$ sobre la frontera de tipo Dirichlet, por lo cual la integral sobre la frontera de Dirichlet en la ecuación (5.10) es igual a cero, en consecuencia

$$\begin{aligned} & - \int_{\partial\Omega} w_i (\underline{D}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) \cdot \underline{n} d\underline{x} = - \int_{\partial_N\Omega} w_i (\underline{D}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) \cdot \underline{n} d\underline{x} \\ & = - \int_{\partial_N\Omega} w_i (\underline{D}_i^n \cdot \nabla c_i^n) \cdot \underline{n} d\underline{x} + \int_{\partial_N\Omega} c_i^n w_i \underline{u}_i^n \cdot \underline{n} d\underline{x} \end{aligned} \quad (5.11)$$

sustituyendo la ec. (5.11) en la ec. (5.9), después aplicando la condición de frontera (5.4) y reacomodando términos, se obtiene la formulación variacional de la componente i

$$\begin{aligned} & \int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} + r_i^n \right) c_i^n w_i d\underline{x} + \int_{\Omega} \nabla w_i \cdot (\underline{D}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) d\underline{x} + \int_{\partial_N\Omega} c_i^n w_i \underline{u}_i^n \cdot \underline{n} d\underline{x} \\ & = \int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} c_i^{n-1} + f_i^n \right) w_i d\underline{x} - \int_{\partial_N\Omega} w_i g_{i\partial} d\underline{x} \end{aligned} \quad (5.12)$$

Es conveniente pensar en el sistema de ecuaciones del modelo de transporte multicomponente como un vector de ecuaciones diferenciales parciales. Las funciones base y de peso, también se recogen en un vector, es decir, si $\underline{c}^n \equiv (c_1^n, \dots, c_m^n) \in \underline{W} \equiv W_1 \times \dots \times W_m$ es el vector de las funciones de base y $\underline{w} \equiv (w_1, \dots, w_m) \in \underline{\hat{W}} \equiv \hat{W}_1 \times \dots \times \hat{W}_m$ es el vector de las funciones de peso, entonces la formulación variacional del modelo de transporte multicomponente es el producto interior del vector de ecuaciones diferenciales parciales y el vector de las funciones de peso, lo cual se puede expresar como la suma de todas las formulaciones variacionales de las componentes, como se muestra a continuación:

$$\begin{aligned} & \sum_{i=1}^m \left(\int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} + r_i^n \right) c_i^n w_i d\underline{x} + \int_{\Omega} \nabla w_i \cdot (\underline{D}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) d\underline{x} + \int_{\partial_N\Omega} c_i^n w_i \underline{u}_i^n \cdot \underline{n} d\underline{x} \right) \\ & = \sum_{i=1}^m \left(\int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} c_i^{n-1} + f_i^n \right) w_i d\underline{x} - \int_{\partial_N\Omega} w_i g_{i\partial} d\underline{x} \right) \end{aligned} \quad (5.13)$$

Se puede reescribir la formulación variacional del modelo de transporte multicomponente (5.13) en su forma bilineal y lineal de la siguiente manera:

Encontrar $\underline{c}^n \in \underline{W}$ tal que $a(\underline{c}^n, \underline{w}) = L(\underline{w})$, $\forall \underline{w} \in \underline{\hat{W}}$ para cada $n = 1, 2, \dots, N$ donde

$$a(\underline{c}^n, \underline{w}) = \sum_{i=1}^m \left(\int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} + r_i^n \right) c_i^n w_i d\underline{x} + \int_{\Omega} \nabla w_i \cdot (\underline{D}_i^n \cdot \nabla c_i^n - \underline{u}_i^n c_i^n) d\underline{x} + \int_{\partial_N\Omega} c_i^n w_i \underline{u}_i^n \cdot \underline{n} d\underline{x} \right) \quad (5.14)$$

$$L(\underline{w}) = \sum_{i=1}^m \left(\int_{\Omega} \left(\frac{R_i^n}{\Delta t_n} c_i^{n-1} + f_i^n \right) w_i d\underline{x} - \int_{\partial_N\Omega} w_i g_{i\partial} d\underline{x} \right) \quad (5.15)$$

A partir de las ecuaciones (5.14) y (5.15), se puede expresar el problema variacional no-lineal del modelo de transporte multicomponente de la siguiente forma:

Encontrar $\underline{c}^n \in \underline{W}$ tal que $F(\underline{c}^n, \underline{w}) = 0$, $\forall \underline{w} \in \hat{\underline{W}}$ para cada $n = 1, 2, \dots, N$ donde

$$F(\underline{c}^n, \underline{w}) = a(\underline{c}^n, \underline{w}) - L(\underline{w}) \quad (5.16)$$

También se puede considerar el problema variacional no-lineal de cada componente i : Encontrar $c_i^n \in W_i$ tal que $F(c_i^n; w_i) = 0$, $\forall w_i \in \hat{W}_i$, para cada $n = 1, 2, \dots, N$, donde ahora $F : W_i \times \hat{W}_i \rightarrow \mathbb{R}$ es una forma semilineal, lineal en el(los) argumento(s) después del punto y coma (;). Para discretizar este problema variacional no-lineal, se puede restringir a un par de espacios discretos de base $W_{ih} \subset W_i$ y de peso $\hat{W}_{ih} \subset \hat{W}_i$:

Encontrar $c_{ih}^n \in W_{ih}$ tal que $F(c_{ih}^n; w_i) = 0$, $\forall w_i \in \hat{W}_{ih}$ para cada $n = 1, 2, \dots, N$.

La solución de elementos finitos $c_{ih}^n = \sum_{j=1}^k C_{ij}^n \phi_j$ puede calcularse resolviendo un sistema no-lineal de ecuaciones $b(C_i^n) = 0$, donde $b : \mathbb{R}^M \rightarrow \mathbb{R}^M$ y

$$b_k(C_i^n) = F(c_{ih}^n; \hat{\phi}_k), \quad k = 1, 2, \dots, M \quad (5.17)$$

Para resolver el sistema no-lineal (5.17) por el método de Newton o alguna variante del método de Newton, se calcula el Jacobiano $A = b'$. Si la forma semilineal F es diferenciable en c_i^n , entonces las entradas del Jacobiano A están dadas por

$$A_{kj}(c_{ih}^n) = \frac{\partial b_k(C_i^n)}{\partial C_{ij}^n} = \frac{\partial}{\partial C_{ij}^n} F(c_{ih}^n; \hat{\phi}_k) = F'(c_{ih}^n; \hat{\phi}_k) \frac{\partial c_{ih}^n}{\partial C_{ij}^n} = F'(c_{ih}^n; \hat{\phi}_k) \phi_j \equiv F'(c_{ih}^n; \phi_j, \hat{\phi}_k) \quad (5.18)$$

En cada iteración de Newton, se debe evaluar (ensamblar) la matriz A y el vector b , y actualizar el vector solución C_i^n por

$$(C_i^n)^{r+1} = (C_i^n)^r - \delta(C_i^n)^r, \quad r = 0, 1, \dots, \quad (5.19)$$

donde $\delta(C_i^n)^r$ resuelve el sistema lineal

$$A(c_{ih}^{nr}) \delta(C_i^n)^r = b(c_{ih}^{nr}) \quad (5.20)$$

Hay que notar que para cada c_{ih}^n fija, $a = F'(c_{ih}^n; \cdot, \cdot)$ es una forma bilineal y $L = F(c_{ih}^n; \cdot)$ es una forma lineal. En cada iteración de Newton, se resuelve un problema variacional lineal de la siguiente forma: Encontrar $\delta c_i^n \in W_{ih,0}$, para cada $n = 1, \dots, N$ tal que

$$F'(c_{ih}^n; \delta c_i^n, w_i) = F(c_{ih}^n; w_i), \quad \forall w_i \in \hat{W}_{ih} \quad (5.21)$$

donde $W_{ih,0} = \{c_i^n - c_j^n : c_i^n, c_j^n \in W_{ih}\}$. Discretizando la ecuación (5.21) como un problema lineal, se recupera el sistema lineal (5.20).

Al formular el método de Newton a un nivel de ecuaciones diferenciales parciales, se produce una linealización de éstas, antes de ser discretizadas. Este método es más sencillo, aunque no está tan comúnmente documentado en la literatura sobre métodos numéricos para ecuaciones diferenciales parciales. Aparentemente no hay necesidad de diferenciaciones para derivar una matriz Jacobiana, pero se hace una derivación matemáticamente equivalente cuando los términos no lineales se linealizan usando los primeros dos términos de la serie de Taylor y cuando se descuidan los productos en la perturbación $\delta(c_i^n)^r$.

5.4. Modelo computacional

La implementación computacional de la formulación variacional del sistema de m ecuaciones se realizó en el lenguaje de programación Python 2.7 utilizando FEniCS project 2017.1, el cual provee un solucionador tipo Newton en la forma de la clase `NewtonSolver` para resolver sistemas de ecuaciones no-lineales de la forma $F(\underline{x}) = 0$, donde $\underline{x} \in \mathbb{R}^n$ y $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Para resolver un problema de este tipo usando el solucionador tipo Newton de FEniCS, es necesario proporcionar una subclase de `NonlinearProblem`. El propósito de un objeto `NonlinearProblem` es evaluar F y el Jacobiano de F , el cual será denotado por $J : \mathbb{R}^n \rightarrow \mathbb{R}^n \times \mathbb{R}^n$. A continuación se muestra un esquema de una clase `MyNonlinearProblem` para resolver una ecuación diferencial no lineal.

```
class MyNonlinearProblem(NonlinearProblem):
    def __init__(self, L, a, bc):
        NonlinearProblem.__init__(self)
        self.L = L
        self.a = a
        self.bc = bc
    def F(self, b, x):
        assemble(self.L, tensor=b)
        self.bc.apply(b, x)
    def J(self, A, x):
        assemble(self.a, tensor=A)
        self.bc.apply(A)
```

Un objeto `MyNonlinearProblem` se construye usando una forma lineal L , que cuando se ensambla corresponde a F , y una forma bilineal a , que cuando se ensambla corresponde a J . Una vez que se ha definido una clase de problema no lineal, se puede crear un objeto `NewtonSolver` y se puede usar el solucionador tipo Newton para calcular el vector de solución \underline{x} del problema no lineal:

```
problem = MyNonlinearProblem(L, a, bc)
newton_solver = NewtonSolver()

newton_solver.solve(problem, u.vector())
```

Se pueden establecer varios parámetros para determinar el comportamiento del solucionador tipo Newton, por ejemplo:

```
newton_solver = NewtonSolver()
newton_solver.parameters["maximum_iterations"] = 20
newton_solver.parameters["relative_tolerance"] = 1.0e-6
newton_solver.parameters["absolute_tolerance"] = 1.0e-10
newton_solver.parameters["error_on_nonconvergence"] = False
```

Para un uso más avanzado, se puede construir un `NewtonSolver` con argumentos que especifiquen el solucionador lineal y el preconditionador que se utilizarán en el proceso de solución. Los métodos tipo Newton pueden ser automatizados con herramientas de FEniCS

```
problem = NonlinearVariationalProblem(F, u, bcs, J)
solver = NonlinearVariationalSolver(problem)
solver.solve()
```

donde F corresponde a la forma no-lineal $F(u; v)$, u es la incógnita que es un objeto `Function`, `bcs` representa las condiciones de frontera esenciales (en general una lista de objetos de la clase `DirichletBC`) y J es una forma variacional para el Jacobiano de F .

El lenguaje UFL, usado para especificar formas débiles, soporta la diferenciación de estas formas. Esta característica facilita el cálculo *simbólico* automático del Jacobiano J llamando a la función `derivative` con F , la solución calculada más recientemente u_- , la cual es una `Function` y la incógnita u que es una `TrialFunction`, como parámetros:

```
u = TrialFunction(V)
v = TestFunction(V)
u_ = Function(V)
F = ...
F = action(F, u_)

J = derivative(F, u_, u) # Gateaux derivative in dir. of u
```

La penúltima declaración es equivalente a $F(u = u_-; v)$, donde u_- es una función de elemento finito existente que representa la aproximación calculada más reciente a la solución. La derivada $J(J)$ de $F(F)$ es formalmente la derivada de Gateaux $DF(u^k; \delta u, v)$ de $F(u, v)$ en $u = u_-$ en la dirección de δu . Técnicamente, la derivada de Gateaux se calcula como:

$$\lim_{\epsilon \rightarrow 0} \frac{d}{d\epsilon} F_i(u_- + \epsilon \delta u; v) \quad (5.22)$$

La δu es ahora la función de base y u_- es la aproximación previa a la solución u .

El siguiente código define un problema variacional no-lineal y un solucionador asociado basado en el método de Newton. También muestra la manera de establecer parámetros clave en el método de Newton, así como la elección del solucionador, el preconditionador y los parámetros asociados, para el sistema lineal que ocurre en la iteración de Newton.

```
problem = NonlinearVariationalProblem(F, u_, bcs, J)
solver = NonlinearVariationalSolver(problem)

prm = solver.parameters
prm["newton_solver"]["absolute_tolerance"] = 1E-8
prm["newton_solver"]["relative_tolerance"] = 1E-7
prm["newton_solver"]["maximum_iterations"] = 25
prm["newton_solver"]["relaxation_parameter"] = 1.0
if iterative_solver:
    prm["linear_solver"] = "gmres"
    prm["preconditioner"] = "ilu"
    prm["krylov_solver"]["absolute_tolerance"] = 1E-9
    prm["krylov_solver"]["relative_tolerance"] = 1E-7
    prm["krylov_solver"]["maximum_iterations"] = 1000
    prm["krylov_solver"]["gmres"]["restart"] = 40
    prm["krylov_solver"]["preconditioner"]["ilu"]["fill_level"] = 0
set_log_level(PROGRESS)

solver.solve()
```

El objeto u_- con el que se retroalimenta el problema variacional no-lineal, se llena con la solución mediante la llamada `solver.solve()`.

Espacio de funciones

```
fs_order = 2 # function space order
P = FiniteElement('P', 'triangle', fs_order)
element = MixedElement([P,...,P]) # one P for each component, m-times
W = FunctionSpace(mesh, element) # function space for system equation
```

Funciones de base y de peso

```
c1,...,cm = TrialFunctions(W)
w1,...,wm = TestFunctions(W)
```

Forma variacional

```
a = R1*c1*w1*dx + dt*inner(nabla_grad(w1), dot(D1,nabla_grad(c1))-u*c1)*dx
  + dt*r1*c1*w1*dx + dt*dot(u,n)*c1*w1*ds(i)
  .
  .
  .
  + Rm*cm*wm*dx + dt*inner(nabla_grad(wm), dot(Dm,nabla_grad(cm))-u*cm)*dx
  + dt*rm*cm*wm*dx + dt*dot(u,n)*cm*wm*ds(i)

L = (R1*c1_n + dt*q1)*w1*dx - dt*g1_in*w1*ds(i)
  .
  .
  .
  + (Rm*cm_n + dt*qm)*wm*dx - dt*gm_in*wm*ds(i)

F = a - L
```

Condiciones iniciales

```
c_0 = Expression(('c10',..., 'cm0'), c10=c1_0,...,cm0=cm_0, degree=fs_order)
c_n.interpolate(c_0)
```

Vector de velocidad constante

```
u = Constant((u_1,...,u_n)) # with u_1,...,u_n constants
```

Lista de condiciones de frontera

```
bcs_t = [bc_1,...,bc_n] # List of boundary conditions
```

Solución en el tiempo

```

c = Function(W)
c_n = Function(W)

c1,...,cm = split(c)
c1_n,...,cm_n = split(c_n)

t_0 = ... # Initial time [s]
t_N = ... # Final time [s]
dt = ... # Time step [s]
t = t_0

while t < t_N:
    # Solve Transport problem for time step
    solve(F == 0, c, bcs_t)
    _c1,...,_cm = c.split()
    c_n.assign(c)      # Update previous solution
    t += dt

```


Capítulo 6

Modelo de flujo monofásico y transporte multicomponente

En el presente capítulo se desarrolla el modelo de flujo monofásico y transporte multicomponente acoplado y se resuelve numéricamente mediante el método de elementos finitos para discretizar el espacio, donde se derivan la formulación variacional mixta-dual para la ecuación de flujo y las formulaciones variacionales estándar de las ecuaciones de transporte, mientras que para para discretizar el tiempo se aplica el método de Euler hacia atrás. Finalmente se resuelven simultáneamente en un sistema de ecuaciones no-lineales.

6.1. Modelo conceptual

1. Sólo hay una fase fluida (f) y una fase sólida (s). Ambas son ligeramente compresibles.
2. El fluido es ligeramente compresible.
3. El flujo del fluido en el medio poroso es monofásico con un componente.
4. Consideraremos un fluido Newtoniano que ocupa el espacio intersticial del medio poroso, bajo condiciones isotérmicas.
5. Se considera la ley de Darcy.
6. El flujo de la masa debido a la dispersión y difusión son tan pequeños con relación al flujo de la masa que se pueden despreciar.
7. La interfase fluido-sólido es una superficie material con respecto a la masa del fluido tal que ninguna masa del fluido puede cruzarla.
8. Hay conservación en la masa del fluido.
9. Hay m componentes en la fase fluida (f) y en la fase sólida (s).
10. El medio poroso se considera homogéneo, isótropo y completamente saturado.
11. Algunas componentes pueden pasar de la fase fluida a la fase sólida y viceversa mediante mecanismos de transporte como procesos de adsorción, desorción, advección, difusión, dispersión y reacción.
12. El estado inicial de la porosidad es constante, pero permite la variación dinámica de la porosidad debido a los procesos de taponamiento/destaponamiento.
13. La permeabilidad inicial es constante y es la misma en todas las direcciones, pero se puede modificar en función con la porosidad de los procesos.
14. Velocidad variable y condiciones de frontera generales.

6.2. Modelo matemático

$$\phi c_t \frac{\partial p}{\partial t} + \nabla \cdot \underline{u} = q/\rho \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (6.1)$$

$$\underline{u} = -\frac{1}{\mu} \underline{k} \cdot \nabla p \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (6.2)$$

$$R_i \frac{\partial c_i}{\partial t} - \nabla \cdot (\underline{D}_i \cdot \nabla c_i) + \nabla \cdot (\underline{u}_i c_i) + r_i c_i = f_i \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (6.3)$$

$$p = p_0 \quad , \quad \forall \underline{x} \in \Omega \text{ y } t = t_0 \quad (6.4)$$

$$c_i = c_i^0 \quad , \quad \forall \underline{x} \in \Omega \text{ y } t = t_0 \quad (6.5)$$

$$p = p_\partial \quad , \quad \forall \underline{x} \in \partial_D \Omega \text{ y } t > t_0 \quad (6.6)$$

$$c_i = c_{i\partial} \quad , \quad \forall \underline{x} \in \partial_D \Omega \text{ y } t > t_0 \quad (6.7)$$

$$-\underline{n} \cdot \underline{u} = g_\partial \quad , \quad \forall \underline{x} \in \partial_N \Omega \text{ y } t > t_0 \quad (6.8)$$

$$-\underline{n} \cdot (\underline{D}_i \cdot \nabla c_i) = g_{i\partial} \quad , \quad \forall \underline{x} \in \partial_N \Omega \text{ y } t > t_0 \quad (6.9)$$

donde \underline{k} y \underline{D}_i son tensores simétricos, positivos definidos, acotados y suaves a tramos en

Ω de la forma $\underline{k} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix}$ y $\underline{D}_i = \begin{pmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{pmatrix}$, con $k_{ij} \equiv k_{ij}(\underline{x}, t)$ y

$D_{ij} \equiv D_{ij}(\underline{x}, t)$, $\underline{u}_i \equiv \underline{u}_i(\underline{x}, t)$, mientras que $\rho \equiv \rho(\underline{x}, t)$, $\mu \equiv \mu(\underline{x}, t)$, $\phi \equiv \phi(\underline{x}, t)$, $q \equiv q(\underline{x}, t)$, $R_i \equiv R_i(\underline{x}, t)$, $r_i \equiv r_i(\underline{x}, t)$ y $f_i \equiv f_i(\underline{x}, t)$. Ω es un dominio acotado con frontera $\partial\Omega$, a su vez la frontera está formada por dos partes, una con condiciones de tipo Dirichlet $\partial_D\Omega$ y otra con condiciones de tipo Neumann $\partial_N\Omega$, donde $\partial\Omega = \partial_D\Omega \cup \partial_N\Omega$, con $\partial_D\Omega \cap \partial_N\Omega = \emptyset$ y c_i es la concentración de la componente i , con $i = 1, 2, 3, \dots, m$.

6.3. Modelo numérico

Se observó que tanto en una formulación variacional estándar como en una formulación variacional mixta-dual, las condiciones de frontera naturales son las de tipo Neumann y las condiciones de frontera esenciales son las de tipo Dirichlet. Debido a esta observación es posible acoplar el modelo de flujo monofásico con el modelo de transporte multicomponente en FEniCS, combinando la formulación variacional mixta-dual del modelo de flujo monofásico (ver ecuaciones (4.69) y (4.68)) con la formulación variacional estándar del modelo de transporte multicomponente (ver ecuaciones (5.14) y (5.15)).

Encontrar $(\underline{c}_h^n, p_h^n, \underline{u}_h^n) \in \underline{W}_h \times W_h \times \underline{V}_h \subset \underline{W} \times W \times \underline{V}$ tal que para cada $n = 1, 2, 3, \dots, N$,

$$F[(\underline{c}_h^n, p_h^n, \underline{u}_h^n), (\underline{w}, w, \underline{v})] = 0, \quad \forall (\underline{w}, w, \underline{v}) \in \hat{\underline{W}} \times \hat{W} \times \hat{\underline{V}} \quad (6.10)$$

donde

$$F[(\underline{c}_h^n, p_h^n, \underline{u}_h^n), (\underline{w}, w, \underline{v})] = a[(\underline{c}_h^n, p_h^n, \underline{u}_h^n), (\underline{w}, w, \underline{v})] - L[(\underline{w}, w, \underline{v})] \quad (6.11)$$

$$\underline{W}_h \equiv W_{1h} \times \dots \times W_{mh} \quad , \quad \underline{W} \equiv W_1 \times \dots \times W_m \quad , \quad \hat{\underline{W}} \equiv \hat{W}_1 \times \dots \times \hat{W}_m$$

con

$$\begin{aligned}
 a[(\underline{c}_h^n, p_h^n, \underline{u}_h^n), (\underline{w}, w, \underline{v})] &= \int_{\Omega} \underline{u}_h^n \cdot \underline{v} + \left(\frac{1}{\mu} \underline{k}^n \cdot \nabla p_h^n \right) \cdot \underline{v} + (\phi c_t)^n p_h^n w - (\Delta t_n) \underline{u}_h^n \cdot \nabla w d\underline{x} \\
 &+ \sum_{i=1}^m \left(\int_{\Omega} ((R_i^n + \Delta t_n r_i^n) c_{ih}^n w_i + \Delta t_n \nabla w_i \cdot (\underline{D}_i^n \cdot \nabla c_{ih}^n - \underline{u}_i^n c_{ih}^n)) d\underline{x} + \Delta t_n \int_{\partial_N \Omega} c_{ih}^n w_i \underline{u}_i^n \cdot \underline{n} d\underline{x} \right)
 \end{aligned} \quad (6.12)$$

$$\begin{aligned}
 L[(\underline{w}, w, \underline{v})] &= \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi c_t)^n p^{n-1}) w d\underline{x} + \Delta t_n \int_{\partial_N \Omega} w u_{in} ds \\
 &+ \sum_{i=1}^m \left(\int_{\Omega} ((\Delta t_n) f_i^n + R_i^n c_i^{n-1}) w_i d\underline{x} - \Delta t_n \int_{\partial_N \Omega} w_i g_{i\partial} d\underline{x} \right)
 \end{aligned} \quad (6.13)$$

6.4. Modelo computacional

La implementación computacional de la formulación variacional del sistema de m ecuaciones se realizó en el lenguaje de programación Python 2.7 utilizando FEniCS project 2017.1.

Espacios de funciones

```

fs_order = 2 # function space order
DRT = FiniteElement("DRT", 'triangle', fs_order)
CG = FiniteElement("CG", 'triangle', fs_order+1)
P = FiniteElement('P', 'triangle', fs_order)
element = MixedElement([P,...,P,CG,DRT]) # one P for each component, m-times
W = FunctionSpace(mesh, element) # function space for system equation
V = VectorFunctionSpace(mesh, 'P', fs_order+1) # function space for velocity

```

Funciones de base y de peso

Considerando c_1, \dots, c_m las m componentes del modelo de transporte multicomponente, $cm1$ como la presión y $cm2$ como la velocidad del fluido, entonces hay $m + 2$ incógnitas:

```

c1,...,cm,cm1,cm2 = TrialFunctions(W)
w1,...,wm,wm1,wm2 = TestFunctions(W)

```

Condiciones iniciales

```

c_0 = Expression(( 'c10',..., 'cm0', 'cm1_0', '0.0', '0.0' ), c10=c1_0,..., cm0=cm_0,
                  cm10=cm1_0, degree=fs_order)
c_n.interpolate(c_0)

```

Forma variacional

```

a = dot(cm2,wm2)*dx + dot(dot((1/mu_w)*Dm2,nabla_grad(cm1)),wm2)*dx
  + R_cm1*cm1*wm1*dx - dt*inner(nabla_grad(wm1), cm2)*dx

  + R_c1*c1*w1*dx + dt*inner(nabla_grad(w1), dot(D1,nabla_grad(c1))-u*c1)*dx
  + dt*r_c1*c1*w1*dx + dt*dot(u,n)*c1*w1*ds(i)
      .
      .
      .
  + R_cm*cm*wm*dx + dt*inner(nabla_grad(wm), dot(Dm,nabla_grad(cm))-u*cm)*dx
  + dt*r_cm*cm*wm*dx + dt*dot(u,n)*cm*wm*ds(i)

L = (R_cm1*cm1_n + dt*q_cm1)*wm1*dx + dt*g_cm1_in*wm1*ds
  + (R_c1*c1_n + dt*q_c1)*w1*dx - dt*g_c1_in*w1*ds(i)
      .
      .
      .
  + (R_cm*cm_n + dt*q_cm)*wm*dx - dt*g_cm_in*wm*ds(i)

F = a - L

```

Solución en el tiempo

```

uf = Function(V)
c = Function(W)
c_n = Function(W)

c1,...,cm, cm1, cm2 = split(c)
c1_n,...,cm_n, cm1_n, cm2_n = split(c_n)

t_0 = ... # Initial time [s]
t_N = ... # Final time [s]
dt = ... # Time step [s]

t = t_0

while t < t_N:

    bcs_t = [bc_1,...,bc_n] # List of boundary conditions

    # Solve Flow and Transport problem for time step
    solve(F == 0, c, bcs_t)
    _c1,...,_cm,_cm1,_cm2 = c.split()
    c_n.assign(c) # Update previous solution

    u = project(cm2,V)
    u_x, u_y = u.split(deepcopy=True) # extract components

    t += dt

```


Capítulo 7

Caso de estudio: Simulación de un proceso de inyección de agua de baja salinidad a escala de laboratorio

En el presente capítulo se realiza la aplicación del modelo de flujo monofásico y transporte multicomponente a un caso de estudio para la simulación del proceso de desprendimiento, migración y bloqueo de gargantas de poros por finos a escala de laboratorio, mediante la técnica de inyección de agua de baja salinidad (LSWF por sus siglas en inglés). Se realizan las simulaciones numéricas usando datos tomados de la literatura y se comparan los resultados obtenidos con los publicados y obtenidos con el software comercial COMSOL Multiphysics®.

El proceso de inyección de agua de baja salinidad (LSWF Low Salinity Water Flooding, por sus siglas en inglés) se inició a mediados de la década de 1990 y es considerado un método de recuperación mejorada de aceite en yacimientos petroleros que consiste en inyectar agua con una salinidad menor al agua presente en el yacimiento, provocando una serie de efectos como desprendimiento, migración y bloqueo por finos, los cuales pueden modificar la permeabilidad de la roca y alterar el flujo en el yacimiento.

Si el agua que se inyecta es del mismo tipo de la que se saca, entonces todo se encuentra en un mismo equilibrio energético, pero si se inyecta de esa misma agua diluida varias veces, entonces disminuye drásticamente la concentración de las sales, las cuales están formadas por cationes, los cuales influyen sobre la adsorción de la roca y en consecuencia hay nuevas reacciones químicas buscando un nuevo equilibrio, entonces la roca experimenta un cambio en la mojabilidad, lo cual influye en la permeabilidad, finalmente favoreciendo el flujo; sin embargo, otro efecto es la precipitación de algunos minerales, dependiendo del tipo de roca, provocando la generación de algunos finos, los cuales al ser arrastrados, tapan las vías y disminuyen la eficiencia del flujo, pero el bloqueo de los principales caminos de flujo obliga al fluido de inyección a tomar otros caminos dentro del medio poroso, entonces los cambios locales de permeabilidad obligan al fluido de inyección a barrer zonas no contactadas previamente.

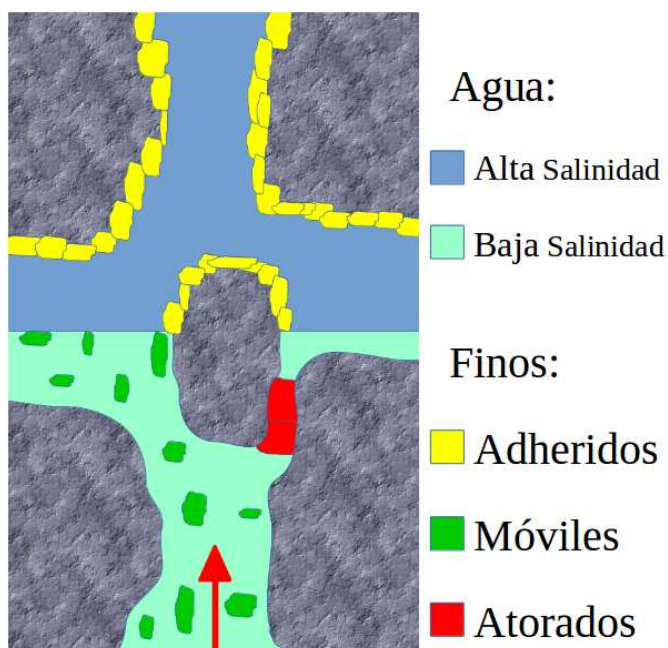


Figura 7.1: Ilustración del proceso de desprendimiento, migración de finos y taponamiento de las gargantas de los poros por inyección de agua de baja salinidad.

En el apéndice A se describen con más detalle las condiciones necesarias para propiciar el efecto de recuperación adicional por inyección de agua de baja salinidad, con el propósito de comprender mejor los fenómenos físicos y mecanismos involucrados en dicho proceso.

La aplicación del modelo de flujo monofásico y transporte multicomponente en medios porosos propuesto en la presente tesis, consistió en resolver numérica y computacionalmente, con el método de elementos finitos mixtos, las ecuaciones de un modelo matemático desarrollado por Coronado y Díaz-Viera [11], el cual describe el proceso de desprendimiento y migración de finos, el taponamiento de las gargantas de los poros y la pérdida de permeabilidad causada por la técnica de inyección de agua de baja salinidad en un núcleo de laboratorio, empleando las características y propiedades del núcleo y del fluido.

7.1. Modelo conceptual

1. Se considera un núcleo homogéneo e isotrópico, completamente saturado inicialmente con salmuera de alta salinidad, en forma de cilindro vertical de longitud L y radio r .
2. La salmuera y la roca son ligeramente compresibles.
3. El estado inicial de la porosidad ϕ_0 es constante, pero se permite la variación dinámica de la porosidad debido a los procesos de taponamiento y destaponamiento.
4. La permeabilidad inicial k_0 es constante y es la misma en todas las direcciones, pero la permeabilidad \underline{K} se puede modificar debido a los procesos de desprendimiento y bloqueo de poros por los finos.
5. La salmuera se inyecta a una velocidad constante en la entrada (fondo) del núcleo y se mantiene a presión constante en la salida (tapa) del núcleo.
6. La salinidad de la salmuera inyectada en la entrada del núcleo, la cual está en función del tiempo, se reduce en forma de escalera hacia menores salinidades, hasta llegar a una salinidad mínima y se registra la caída total de la presión en el núcleo.
7. El sistema de ecuaciones diferenciales parciales comprende ecuaciones para la presión del fluido p , la velocidad (Darcy) de la salmuera \underline{u} , la salinidad c_s , el total de finos c_f , los finos adheridos σ_a y los finos atorados σ_c .
8. El sistema de ecuaciones está acoplado dinámicamente, ya que el transporte de salinidad y los procesos de los finos, implican la velocidad advectiva \underline{u} , la concentración de finos atorados σ_c modifica la permeabilidad \underline{k} y esto a su vez modifica la presión p .
9. Al inicio, no hay finos móviles ni atorados, solamente hay algunos finos adheridos que se van desprendiendo, convirtiéndose en finos móviles que pueden llegar a atorarse más adelante pero no vuelven a adherirse, así como los finos atorados no se desatoran.
10. El volumen de finos efluentes es insignificante en comparación con el volumen de los poros del núcleo. Los finos dentro del núcleo cambian de posición desprendiéndose y atorándose, pero mantienen la porosidad dentro del núcleo esencialmente constante.

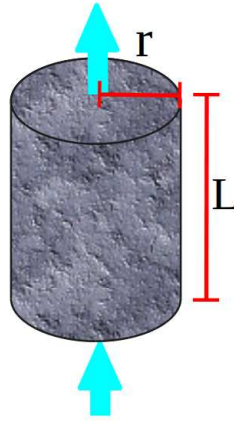


Figura 7.2: Núcleo vertical de longitud L y radio r .

7.2. Modelo matemático

Siguiendo a Coronado y Díaz-Viera [11], se presentan las ecuaciones que describen un proceso de desprendimiento y migración de finos, así como el taponamiento de las gargantas de poros, mediante la técnica de inyección de agua de baja salinidad en un núcleo de laboratorio.

7.2.1. Ecuación de transporte de la salinidad C_s

El transporte de la salinidad C_s se describe mediante la ecuación advectivo-dispersiva:

$$\phi \frac{\partial C_s}{\partial t} - \nabla \cdot (\underline{\underline{D}}_s \cdot \nabla C_s) + \nabla \cdot (\underline{u} C_s) = 0 \quad (7.1)$$

donde ϕ es la porosidad, \underline{u} es la velocidad de Darcy, $D_s = \alpha_s u$ es el coeficiente de dispersión longitudinal de la salinidad, donde la constante α_s es la dispersividad longitudinal de la salinidad y $u = -\frac{k}{\mu} \frac{\partial P}{\partial z}$.

Condición inicial:

$$C_s(\underline{x}, t_0) = C_{s0} \quad (7.2)$$

Condiciones de frontera:

- $C_s = C_{s,inj}(t)$. . . en la entrada del núcleo, la salinidad es variable en el tiempo.
- $(\partial C_s / \partial z) = 0$. . . en la salida del núcleo.
- No flujo en la dirección radial.

La salinidad inicial inyectada en la entrada del núcleo debe satisfacer $C_{s,inj}(t_0) = C_{s0}$. El avance del frente de baja salinidad en el núcleo se debe a que la salinidad de la salmuera inyectada en la entrada del núcleo, la cual está en función del tiempo, se reduce en forma de escalera hacia menores salinidades, hasta llegar a una salinidad mínima.

7.2.2. Ecuación de finos adheridos σ_a

Para el proceso de desprendimiento de finos ($\sigma_a \geq \sigma_{cr}$) la ecuación es

$$\frac{\partial \sigma_a}{\partial t} = -\lambda_d U (\sigma_a - \sigma_{cr}) \quad (7.3)$$

donde σ_a es la concentración de finos adheridos, λ_d es el coeficiente de desprendimiento, $U = Q_{iny}/(\pi r^2)$ donde Q_{iny} es el gasto de inyección de salmuera y σ_{cr} es la concentración crítica de finos adheridos a la superficie de la roca (ver ec. 7.12).

Condición inicial:

$$\sigma_a(\underline{x}, t_0) = \sigma_{a0} \quad (7.4)$$

Condición de frontera:

No-flujo en todas las fronteras.

$(\lambda_d U)^{-1}$ puede verse como el tiempo de retardo característico asociado al tiempo que el proceso de desprendimiento requiere para alcanzar el estado estacionario $\sigma_a = \sigma_{cr}$. Además, como σ_{cr} depende de la salinidad C_s , entonces $\sigma_{cr}(C_s)$ puede verse como una isoterma de adsorción. El valor máximo para σ_{a0} puede estimarse en términos de la fracción de masa de la roca que puede liberar finos χ_f como $\sigma_{a0,max} = \rho_R \chi_f$, donde ρ_R es la densidad de la roca.

7.2.3. Ecuación de transporte de finos móviles C_f

La ecuación que involucra la dispersión de los finos móviles C_f es la siguiente:

$$\phi \frac{\partial C_f}{\partial t} - \nabla \cdot (\underline{D}_f \cdot \nabla C_f) + \nabla \cdot (\underline{u} C_f) = -(1 - \phi) \left(\chi_f \frac{\partial \sigma_a}{\partial t} + \frac{\partial \sigma_c}{\partial t} \right) \quad (7.5)$$

donde ϕ es la porosidad, \underline{u} es la velocidad de Darcy, $D_f = \alpha_f u$ es el coeficiente de dispersión longitudinal de los finos móviles, donde la constante α_f es la dispersividad longitudinal de los finos móviles y $u = (-k/\mu)(\partial P/\partial z)$, χ_f es la fracción de masa de la roca que puede liberar finos, σ_a es la concentración de finos adheridos y σ_c es la concentración de finos atorados.

Condición inicial:

$$C_f(\underline{x}, t_0) = C_{f0} \quad (7.6)$$

donde $C_{f0} = 0$, dado que al tiempo inicial no hay finos móviles presentes.

Condiciones de frontera:

- $C_f = C_{f,inj} \dots$ en la entrada del núcleo, la concentración de finos móviles es $C_{f,inj} = 0$.
- $(\partial C_f / \partial z) = 0 \dots$ a la salida del núcleo.
- No flujo de finos en la dirección radial.

7.2.4. Ecuación de finos atorados σ_c

La dinámica de la concentración de los finos atorados σ_c está dada por la ecuación:

$$\frac{\partial \sigma_c}{\partial t} = \lambda_c U C_f \quad (7.7)$$

donde λ_c es la tasa de retención de finos, $U = Q_{iny}/(\pi r^2)$ donde Q_{iny} es el gasto de inyección de salmuera, y C_f es la concentración de los finos móviles.

Condición inicial:

$$\sigma_c(\underline{x}, t_0) = \sigma_{c0} \quad (7.8)$$

donde $\sigma_{c0} = 0$ dado que al tiempo inicial no hay finos atorados presentes.

Condición de frontera:

No flujo en todas las fronteras.

7.2.5. Ecuaciones de la presión p y la velocidad \underline{u}

Las ecuaciones que describen la presión p y la velocidad \underline{u} están dadas por el modelo de flujo monofásico ligeramente compresible en un medio poroso, descrito en el capítulo 4:

$$\phi c_t \frac{\partial p}{\partial t} + \nabla \cdot \underline{u} = q/\rho \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (7.9)$$

$$\underline{u} = -\frac{1}{\mu} \underline{k} \cdot \nabla p \quad , \quad \forall \underline{x} \in \Omega \text{ y } t > t_0 \quad (7.10)$$

donde ϕ es la porosidad del núcleo, $c_T = c_f + c_R$ es la compresibilidad total, igual a la suma de la compresibilidad del fluido c_f y la compresibilidad de la roca c_R , μ es la viscosidad de la salmuera y \underline{k} es la permeabilidad, la cual es una función que depende del espacio y el tiempo e involucra todo el proceso de desprendimiento y bloqueo de poros por los finos.

Condición inicial:

$$p(\underline{x}, t_0) = p_0 \quad (7.11)$$

donde $p_0 = p_{out}$, es decir, la presión inicial es igual a la presión en la salida del núcleo.

Condiciones de frontera:

- $-\underline{u} \cdot \underline{n} = Q_{iny}/(\pi r^2)$. . . en la entrada del núcleo, el gasto de inyección de la salmuera Q_{iny} es constante y r es el radio del núcleo.
- $p = p_{out}$. . . en la salida del núcleo, la presión es constante.
- No flujo en la dirección radial del núcleo.

7.2.6. Ecuaciones de Estado

Concentración crítica de finos adheridos σ_{cr}

La concentración crítica de los finos adheridos σ_{cr} , depende de la salinidad C_s y especifica la retención máxima de los finos adheridos a la superficie de la roca en los poros.

$$\sigma_{cr}(C_s) = \sigma_{cr,0} \exp\{-[C_{s1}/(C_s - C_{s2})]^{nc}\} \quad (7.12)$$

donde $\sigma_{cr,0}$ es la concentración inicial de los finos adheridos. Los parámetros $\sigma_{cr,0}$, C_{s1} , C_{s2} y nc determinan la curva de desprendimiento de los finos adheridos en función de la salinidad (ver figura 7.3). El parámetro $\sigma_{cr,0}$ determina la altura de la curva y los siguientes la forma. Para ajustar estos parámetros se requiere información de la concentración de los finos en el agua effluente.

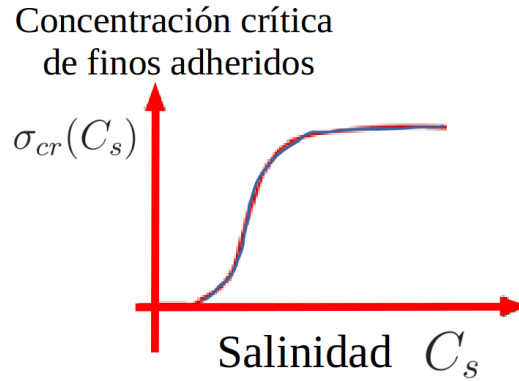


Figura 7.3: Concentración crítica de finos adheridos σ_{cr} como función de la salinidad.

Pérdida de permeabilidad

El efecto de los finos atorados σ_c sobre el deterioro de la permeabilidad absoluta k del medio poroso, se describe por la función de daño [22][16]:

$$k = \frac{k_0}{1 + \beta\sigma_c} \quad (7.13)$$

donde β es el coeficiente de daño a la formación y k_0 es la permeabilidad previa al bloqueo de los poros por los finos.

Permeabilidad efectiva k_{eff}

La permeabilidad efectiva k_{eff} se calcula de la siguiente manera:

$$k_{eff} = \frac{\mu Q_{iny} l}{\pi r^2 k_0 \Delta p} \quad (7.14)$$

donde μ es la viscosidad de la salmuera, Q_{iny} es el gasto de inyección de salmuera a la entrada del núcleo, l es la longitud del núcleo, r es el radio del núcleo, k_0 es la permeabilidad previa al bloqueo de los poros por los finos y Δp es la diferencia de presión entre la entrada y la salida del núcleo ($\Delta p > 0$).

7.2.7. Diagrama causal de la dinámica del sistema

El diagrama causal es un diagrama que recoge los elementos clave del sistema y las relaciones entre ellos [14]. En este diagrama, las relaciones están representadas por flechas entre las variables afectadas por ellas. Esas flechas van acompañadas de un signo (+ o -) que indica el tipo de influencia ejercida por una variable sobre la otra. Un signo (+) quiere decir que un cambio en la variable origen de la flecha producirá un cambio del mismo sentido en la variable destino. El signo (-) simboliza que el efecto producido será en sentido contrario. Así cuando un incremento de **A**, produce un incremento de **B**, o bien una disminución de **A** provoca una disminución de **B**, tendremos una relación positiva (+). Y cuando un incremento de **A**, produce una disminución de **B**, o bien una disminución de **A** provoca un aumento de **B**, tendremos una relación negativa (-). Una cadena cerrada de relaciones causales recibe el nombre de bucle, retroalimentación o feedback. Por ejemplo, cuando abrimos el grifo para llenar un vaso de agua aumentamos la cantidad de agua en el vaso, pero también la cantidad de agua que hay en el vaso modifica la velocidad en la que nosotros llenamos el vaso. Lo llenamos más despacio cuando está casi lleno; y por lo tanto existe un bucle. Los bucles se definen como *positivos* cuando el número de relaciones *negativas* es par, y *negativos* si es impar (igual que al multiplicar: $-a \times b = -c$). Los sistemas contienen ambos tipos de bucles y el comportamiento final dependerá de cual es el dominante en un momento determinado.

El sistema de ecuaciones está acoplado dinámicamente, ya que el transporte de salinidad y los procesos de los finos como la concentración de finos atorados σ_c modifican la permeabilidad \underline{K} que implica la velocidad advectiva \underline{u} y ésta a su vez modifica la presión p .

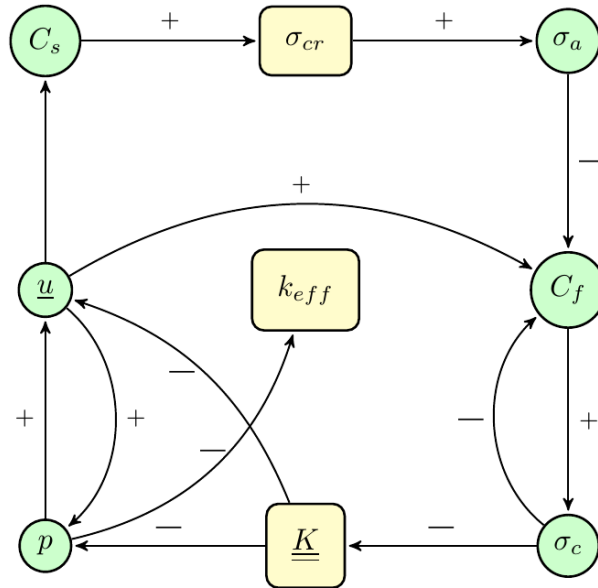


Figura 7.4: Diagrama causal de la dinámica entre las variables del sistema en un proceso de desprendimiento, migración y bloqueo de poros por finos con la técnica de inyección de agua de baja salinidad. Las variables principales o incógnitas del sistema están encerradas en círculos y las variables de las ecuaciones de estado están encerradas en rectángulos.

7.3. Modelo numérico

Considerando el modelo de flujo monofásico y transporte multicomponente (ecuaciones 6.1 a 6.9) visto en el capítulo 6, se observa que el modelo de flujo (ecuaciones 7.9 y 7.10) es el mismo, mientras que las ecuaciones que describen el transporte de la salinidad y la dinámica de los procesos de desprendimiento, migración y bloqueo de gargantas de poros por finos (ver ecuaciones 7.1, 7.3, 7.5 y 7.7), conforman un sistema de 4 ecuaciones parabólicas de segundo orden como la ecuación (7.15):

$$R_i \frac{\partial c_i}{\partial t} - \nabla \cdot (\underline{\underline{D}}_i \cdot \nabla c_i) + \nabla \cdot (\underline{u}_i c_i) + r_i c_i = f_i \quad (7.15)$$

donde:

R_i es el término de reacción.

c_i es la concentración de masa de la componente i , que se va a calcular.

$\underline{\underline{D}}_i$ es el tensor de dispersión hidrodinámica o término difusivo.

\underline{u}_i es la velocidad de cada fase: sólida o fluida.

$\nabla \cdot (\underline{u}_i c_i)$ es el término advectivo.

r_i es el término de reacción.

f_i es el término fuente.

Los coeficientes de la ecuación de cada componente, se plantean en el cuadro 7.1:

c_i	R_i	$\underline{\underline{D}}_i$	\underline{u}_i	r_i	f_i	Descripción
$c_1 = C_s$	ϕ	$\underline{\underline{D}}_s$	\underline{u}	0	0	Concentración de la sal.
$c_2 = \sigma_a$	1	$\underline{\underline{0}}$	$\underline{\underline{0}}$	0	$-\lambda_a U(\sigma_a - \sigma_{cr})$	Concentración de finos adheridos.
$c_3 = C_f$	ϕ	$\underline{\underline{D}}_f$	\underline{u}	0	$-(1 - \phi)(\chi_f \frac{\partial \sigma_a}{\partial t} + \frac{\partial \sigma_c}{\partial t})$	Concentración de finos móviles.
$c_4 = \sigma_c$	1	$\underline{\underline{0}}$	$\underline{\underline{0}}$	0	$\lambda_c U C_f$	Concentración de finos atorados.

Cuadro 7.1: Coeficientes de la ecuación de cada componente del sistema.

Sustituyendo los coeficientes de cada ecuación de cada componente como se muestra en el cuadro 7.1 en el modelo numérico del modelo de flujo monofásico y transporte multicomponente (ec. (6.10)), se obtiene el modelo numérico para este caso de estudio.

Encontrar:

$(c_{1h}^n, c_{2h}^n, c_{3h}^n, c_{4h}^n, p_h^n, \underline{u}_h^n) \in W_{1h} \times W_{2h} \times W_{3h} \times W_{4h} \times W_h \times \underline{V}_h \subset W_1 \times W_2 \times W_3 \times W_4 \times W \times \underline{V}$
tal que para cada $n = 1, 2, 3, \dots, N$

$$F((c_{1h}^n, c_{2h}^n, c_{3h}^n, c_{4h}^n, p_h^n, \underline{u}_h^n), (w_1, w_2, w_3, w_4, w, \underline{v})) = 0 \quad (7.16)$$

$$\forall (w_1, w_2, w_3, w_4, w, \underline{v}) \in \hat{W}_1 \times \hat{W}_2 \times \hat{W}_3 \times \hat{W}_4 \times \hat{W} \times \underline{\hat{V}}$$

donde

$$\begin{aligned} & F[(c_{1h}^n, c_{2h}^n, c_{3h}^n, c_{4h}^n, p_h^n, \underline{u}_h^n), (w_1, w_2, w_3, w_4, w, \underline{v})] \\ &= a[(c_{1h}^n, c_{2h}^n, c_{3h}^n, c_{4h}^n, p_h^n, \underline{u}_h^n), (w_1, w_2, w_3, w_4, w, \underline{v})] \\ &- L[(w_1, w_2, w_3, w_4, w, \underline{v})] \end{aligned} \quad (7.17)$$

con

$$\begin{aligned}
& a[(c_{1h}^n, c_{2h}^n, c_{3h}^n, c_{4h}^n, p_h^n, \underline{u}_h^n), (w_1, w_2, w_3, w_4, w, \underline{v})] \\
& = \int_{\Omega} \left(\underline{u}_h^n \cdot \underline{v} + \left(\frac{1}{\mu} \underline{k}^n \cdot \nabla p_h^n \right) \cdot \underline{v} + (\phi c_t)^n p_h^n w - (\Delta t_n) \underline{u}_h^n \cdot \nabla w \right) d\underline{x} \\
& + \sum_{i=1}^4 \left(\int_{\Omega} ((R_i^n + (\Delta t_n) r_i^n) c_{ih}^n w_i + \Delta t_n \nabla w_i \cdot (\underline{D}_i^n \cdot \nabla c_{ih}^n - \underline{u}_i^n c_{ih}^n)) d\underline{x} + \Delta t_n \int_{\partial_N \Omega} c_{ih}^n w_i \underline{u}_i^n \cdot \underline{n} d\underline{x} \right)
\end{aligned} \tag{7.18}$$

$$\begin{aligned}
L[(w_1, w_2, w_3, w_4, w, \underline{v})] & = \int_{\Omega} ((\Delta t_n)(q/\rho)^n + (\phi c_t)^n p^{n-1}) w d\underline{x} + \Delta t_n \int_{\partial_N \Omega} w u_{in} ds \\
& + \sum_{i=1}^4 \left(\int_{\Omega} ((\Delta t_n) f_i^n + R_i^n c_i^{n-1}) w_i d\underline{x} - \Delta t_n \int_{\partial_N \Omega} w_i g_{i\partial} ds \right)
\end{aligned} \tag{7.19}$$

Una combinación estable de espacios de elementos finitos mixtos es para $W_{1h}, W_{2h}, W_{3h}, W_{4h}$ los elementos de *Lagrange* (**P**) de orden polinomial $k+1$, mientras que para \underline{V}_h los elementos discontinuos de *Raviart-Thomas* (**DRT**) de orden polinomial k y para W_h los elementos continuos de *Galerkin* (**CG**) de orden polinomial $k+1$.

7.4. Modelo computacional

La implementación computacional del modelo numérico se realizó en el lenguaje de programación Python 2.7 utilizando FEniCS project 2017.1

7.4.1. Espacios de funciones

```

fs_order = 2 # function space order
DRT = FiniteElement("DRT", 'triangle', fs_order)
CG = FiniteElement("CG", 'triangle', fs_order+1)
P = FiniteElement('P', 'triangle', fs_order+1)
element = MixedElement([P, P, P, P, CG, DRT])
W = FunctionSpace(mesh, element) # function space for system equation
V = VectorFunctionSpace(mesh, 'P', fs_order+1) # function space for velocity

```

7.4.2. Funciones de base y de peso

```

c1, c2, c3, c4, c5, c6 = TrialFunctions(W)
w1, w2, w3, w4, w5, w6 = TestFunctions(W)

```

7.4.3. Fronteras naturales y esenciales

```

u.B = Constant((ux_in, uy_in)) #Constant flow
bc1_f=DirichletBC(W.sub(5), u.B, boundary_parts, 1)

c1.B = Constant(c1_in)
bc1_l=DirichletBC(W.sub(0), c1.B, boundary_parts, 1)

c3.B = Constant(c3_in)
bc3_l=DirichletBC(W.sub(2), c3.B, boundary_parts, 1)

p.T = Constant(p_out)
bc5_3=DirichletBC(W.sub(4), p.T, boundary_parts, 3)

bcs_t = [bc1_l, bc3_l, bc5_3]

```

7.4.4. Forma variacional

```

a = dot(c6, w6)*dx + dot(dot((1/mu_w)*D5, nabla_grad(c5)), w6)*dx
  + R_c5*c5*w5*dx - dt*inner(nabla_grad(w5), c6)*dx

  + R_c1*c1*w1*dx + dt*inner(nabla_grad(w1), dot(D1, nabla_grad(c1))-u*c1)*dx
  + dt*dot(u, n)*c1*w1*ds(3)

  + R_c2*c2*w2*dx

  + R_c3*c3*w3*dx + dt*inner(nabla_grad(w3), dot(D3, nabla_grad(c3))-u*c3)*dx
  + dt*dot(u, n)*c3*w3*ds(3)

  + R_c4*c4*w4*dx

L = (R_c5*c5_n + dt*q_c5)*w5*dx + dt*g_c5_in*w5*ds(1)

  + (R_c1*c1_n + dt*q_c1)*w1*dx

  + (R_c2*c2_n + dt*q_c2)*w2*dx

  + (R_c3*c3_n + dt*q_c3)*w3*dx

  + (R_c4*c4_n + dt*q_c4)*w4*dx

F = a - L

```

7.4.5. Condiciones iniciales

```

c_0 = Expression(('c10', 'c20', 'c30', 'c40', 'c50', '0.0', '0.0'), c10=c1_0, c20=c2_0,
  , c30=c3_0, c40=c4_0, c50=c5_0, degree=fs_order)
c_n.interpolate(c_0)

```

7.4.6. Solución en el tiempo

```

uf = Function(V)
c = Function(W)
c_n = Function(W)

c1, c2, c3, c4, c5, c6 = split(c)
c1_n, c2_n, c3_n, c4_n, c5_n, c6_n = split(c_n)

t_0 = ... # Initial time [s]
t_N = ... # Final time [s]
dt = ... # Time step [s]

t = t_0

while t < t_N:

    c1_B = Constant(c1_in)
    bc1_1=DirichletBC(W.sub(0), c1_B, boundary_parts, 1)
    bcs_t = [bc1_1, bc3_1, bc5_3] #BCs Transport

    # Solve Transport problem for time step
    solve(F == 0, c, bcs_t)
    _c1, _c2, _c3, _c4, _c5, _c6 = c.split()
    c_n.assign(c) # Update previous solution

    k=Expression("k_0/(1.0+betabeta*c4)", k_0=k_0, betabeta=betabeta, c4=_c4,
        degree=fs_order)
    u = project(c6, V)
    u_x, u_y = u.split(deepcopy=True) # extract components

    t += dt

```

7.5. Simulaciones

En esta sección se analizará un experimento de inyección de agua de baja salinidad. El experimento considera la inyección a tasa de flujo constante en un núcleo de laboratorio, mientras que la salinidad de la salmuera de inyección es reducida en escalera. Se trata de pruebas de laboratorio relativas al desplazamiento monofásico (salmuera) en núcleos [11].

Las características del núcleo, el fluido y del experimento se muestran en el cuadro (7.2).

Parámetro	Valor	Descripción
L	2.54 <i>cm</i>	Longitud del núcleo
D	2.54 <i>cm</i>	Diámetro del núcleo
ϕ	19 %	Porosidad
PV	2.44 <i>ml</i>	Volumen Poroso
k_0	100 <i>mD</i>	Permeabilidad inicial
μ	0.3 <i>cP</i>	Viscosidad del agua
ρ	1 <i>g/ml</i>	Densidad del agua
c_T	$6 * 10^{-10} \text{ Pa}^{-1}$	Compresibilidad total
α_s	$L/10$	Dispersividad longitudinal de la salinidad
α_f	$L/10$	Dispersividad longitudinal de los finos móviles
Q_{inj}	0.027 <i>ml/s</i>	Gasto de inyección
p_{out}	2900 <i>psi</i>	Presión a la salida
χ_f	8 % <i>w</i>	Fracción de la roca que puede liberar finos
λ_d	100 <i>m</i> ⁻¹	Tasa de desprendimiento de finos
λ_c	0.5 <i>m</i> ⁻¹	Tasa de taponamiento por finos
$\sigma_{cr,0}$	180 <i>g/l</i>	Altura de la curva de σ_{cr}
C_{s1}	18 <i>ppm</i>	Curvatura de la curva de σ_{cr}
C_{s2}	3905.5 <i>ppm</i>	Desplazamiento lateral σ_{cr}
nc	3.8 <i>ppm</i>	Parámetro en σ_{cr}
β	7000 [<i>g/l</i>] ⁻¹	Coeficiente de daño de la formación

Cuadro 7.2: Características del núcleo, fluido y experimento.

Los valores de la escalera de reducción de salinidad en la salmuera inyectada y los tiempos se listan en la tabla (7.3).

PVI	Tiempo de cambio	Salinidad de la salmuera
0 – 40	3,623.0 (s)	30, 000 (ppm)
40 – 80	7,245.6 (s)	10, 000 (ppm)
80 – 120	10,868.4 (s)	5, 000 (ppm)
120 – 160	14,491.1 (s)	4, 500 (ppm)
160 – 200	18,113.86 (s)	4, 250 (ppm)
200 – 240	21,600.0 (s)	4, 000 (ppm)

Cuadro 7.3: Datos de la escalera de reducción de salinidad de la salmuera de inyección.

Las mallas utilizadas para la simulación numérica generadas en COMSOL y FEniCS se muestran en las figuras 7.5 y 7.6 respectivamente.

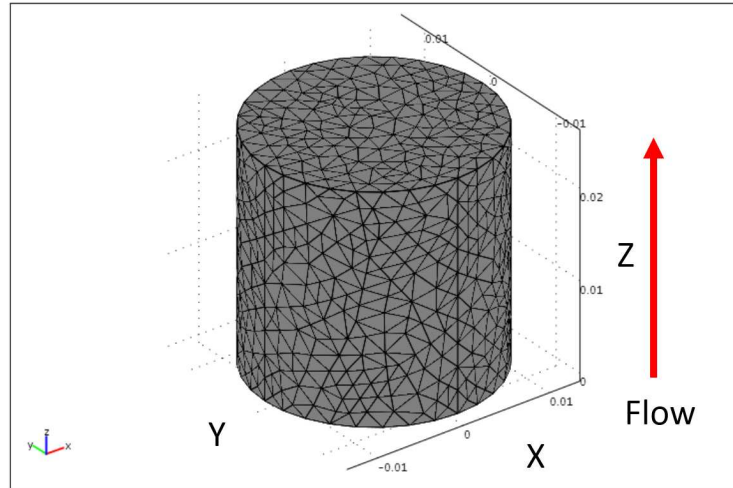


Figura 7.5: Malla generada con COMSOL en 3D con 13,412 elementos de tetraédricos y 2,714 vértices.

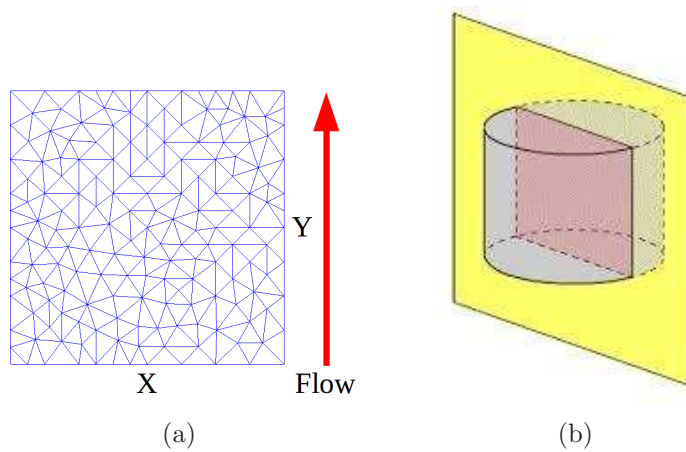


Figura 7.6: Malla generada con FEniCS en 2D con 294 elementos triangulares y 170 vértices. Esta malla representa una sección transversal (corte vertical) del núcleo de laboratorio.

7.5.1. Simulaciones de la salinidad del fluido

Las gráficas de las figuras (7.7) y (7.8) describen la evolución en el tiempo de la salinidad del fluido, en la entrada y en la salida del núcleo respectivamente y como aproximadamente cada hora se inyecta agua con menos sal, entonces la salinidad disminuye en forma de escalones conforme transcurre el tiempo. Se puede observar que las gráficas son muy similares tanto en FEniCS como en COMSOL.

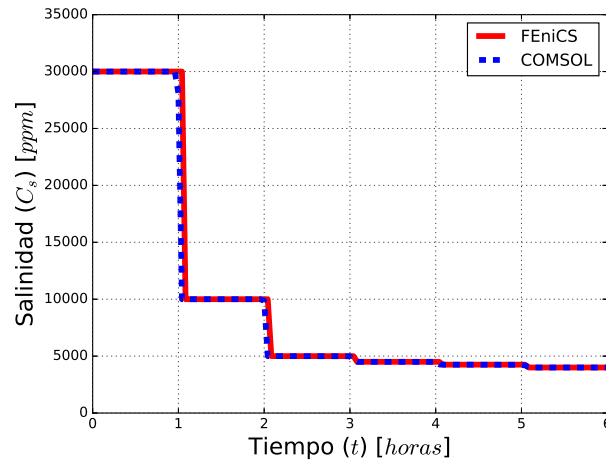


Figura 7.7: Evolución en el tiempo de la salinidad del fluido en la entrada del núcleo.

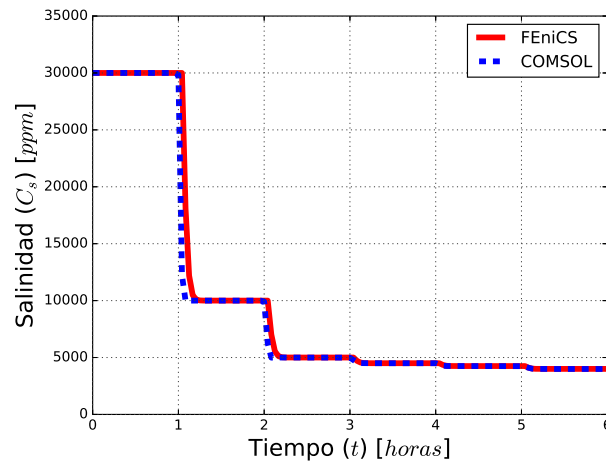


Figura 7.8: Evolución en el tiempo de la salinidad del fluido en la salida del núcleo.

Las gráficas de las figuras (7.9) y (7.10) representan a los perfiles de la salinidad del fluido a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. Se observa que durante la primera hora, el fluido a lo largo de la longitud del núcleo permanece con la misma salinidad alta y cada hora aproximadamente se va inyectando agua con menos salinidad. Las curvas intermedias denotan que la salinidad del fluido no disminuye completamente a lo largo del núcleo, aunque en la entrada ya disminuyó, después se estabiliza hasta cierto nivel y luego se vuelve a inyectar agua con menos salinidad.

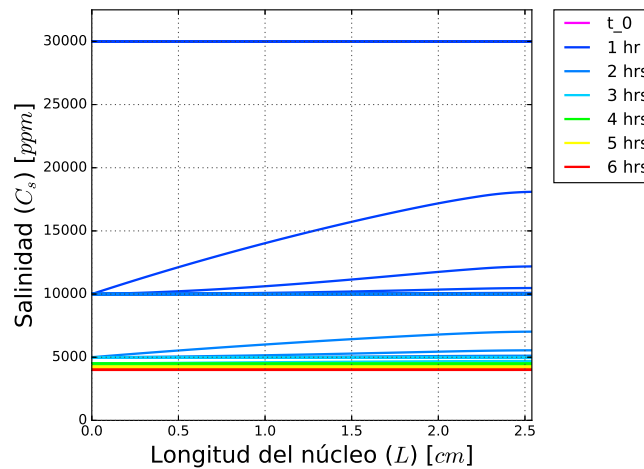


Figura 7.9: Simulación con FEniCS de los perfiles de la salinidad del fluido a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

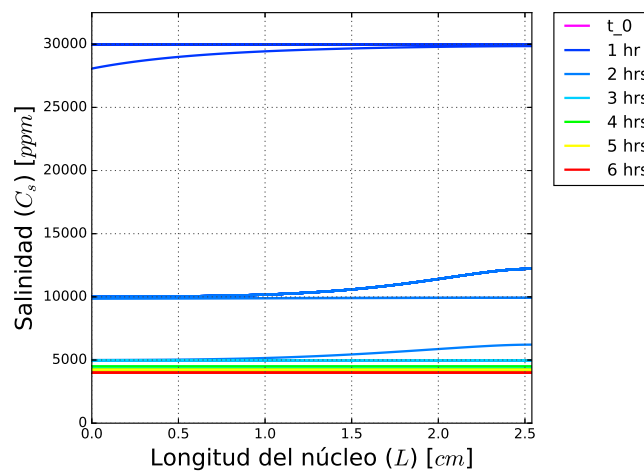


Figura 7.10: Simulación con COMSOL de los perfiles de la salinidad del fluido a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

7.5.2. Simulaciones de la concentración crítica de finos adheridos

La gráfica de la figura (7.11) describe el comportamiento de la concentración crítica de los finos adheridos σ_{cr} , en función de la salinidad. La gráfica de la figura (7.12) describe la evolución en el tiempo de la concentración crítica de los finos adheridos en la entrada del núcleo. La concentración crítica se mantiene hasta cierto tiempo y disminuye en forma abrupta a partir de que la salinidad se reduce lo suficiente (ver fig. 7.7), provocando que comiencen a desprenderse los finos que se encuentran adheridos a la roca.

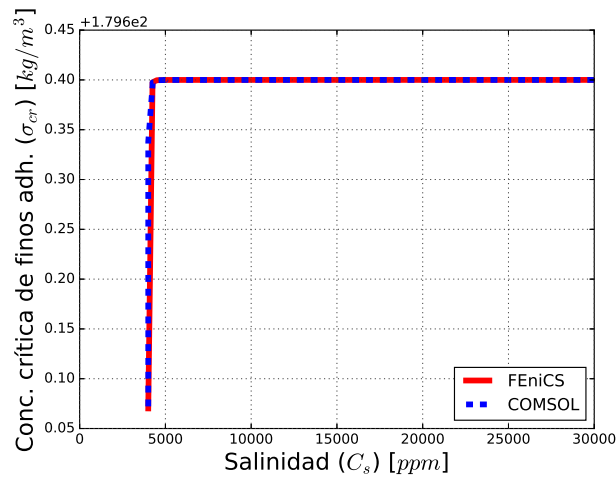


Figura 7.11: Concentración crítica de finos adheridos en función de la salinidad.

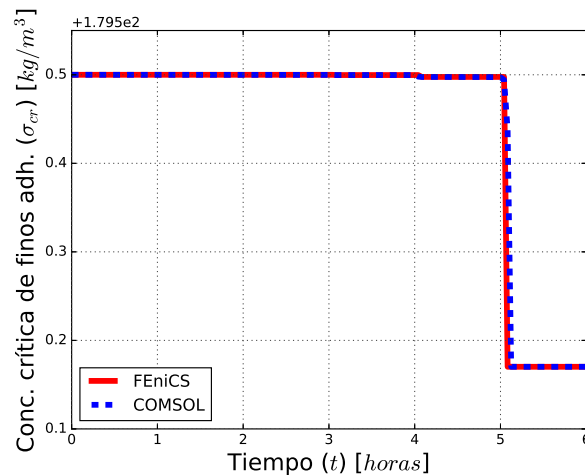


Figura 7.12: Evolución en el tiempo de la concentración crítica de finos adheridos.

7.5.3. Simulaciones de la concentración de finos adheridos

Las gráficas de las figuras (7.13) y (7.14) describen la evolución en el tiempo de la concentración de los finos adheridos a la roca σ_a en la entrada y en la salida del núcleo respectivamente. La concentración de finos adheridos a la roca se mantiene hasta cierto tiempo y disminuye a partir de que la salinidad del fluido se reduce lo suficiente, causando que comiencen a desprenderse algunos finos de la roca. Como el agua de baja salinidad se inyecta de abajo hacia arriba del núcleo, entonces debe haber una mayor concentración de finos adheridos en la salida del núcleo que en la entrada, por lo que en la figura (7.14), la gráfica obtenida con FEniCS es más coherente que la obtenida con COMSOL. La diferencia se debe a que en FEniCS, se implementó la formulación variacional con el método de elementos finitos mixtos dual, mientras que COMSOL solamente tiene implementado el método estándar.

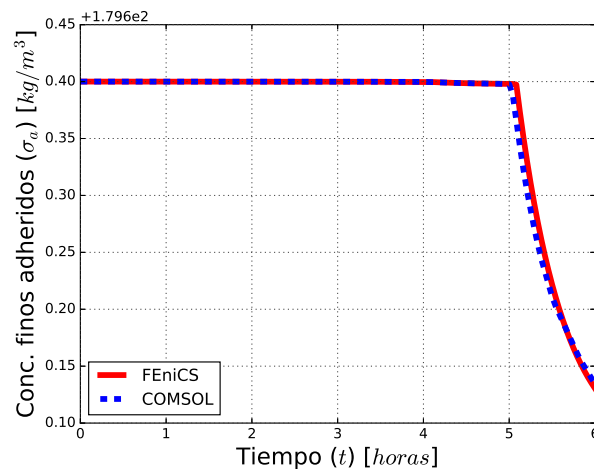


Figura 7.13: Evolución en el tiempo de la conc. de finos adheridos en la entrada del núcleo.

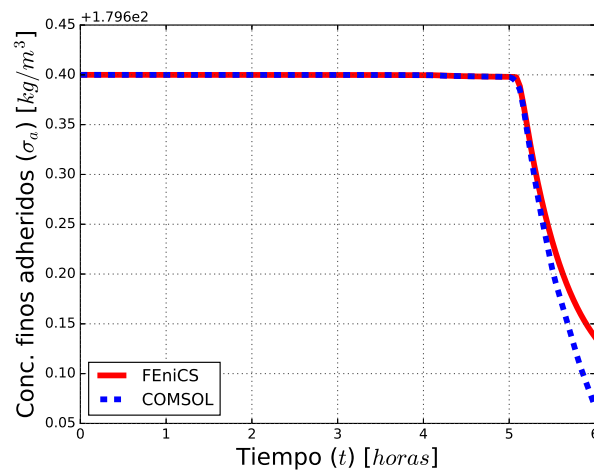


Figura 7.14: Evolución en el tiempo de la conc. de finos adheridos en la salida del núcleo.

Las gráficas de las figuras (7.15) y (7.16) representan a los perfiles de la concentración de finos adheridos a la roca a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. Como la concentración de finos adheridos debe ir disminuyendo de abajo hacia arriba del núcleo, conforme va entrando el agua de baja salinidad, entonces los perfiles obtenidos con FEniCS son congruentes con el proceso de desprendimiento de finos, mientras que los perfiles obtenidos con COMSOL son incongruentes, pues indican que cada vez hay menos finos adheridos de arriba hacia abajo del núcleo, como si el agua de baja salinidad entrara por arriba del núcleo. La diferencia de resultados se debe a que en FEniCS se utilizaron elementos finitos mixtos.

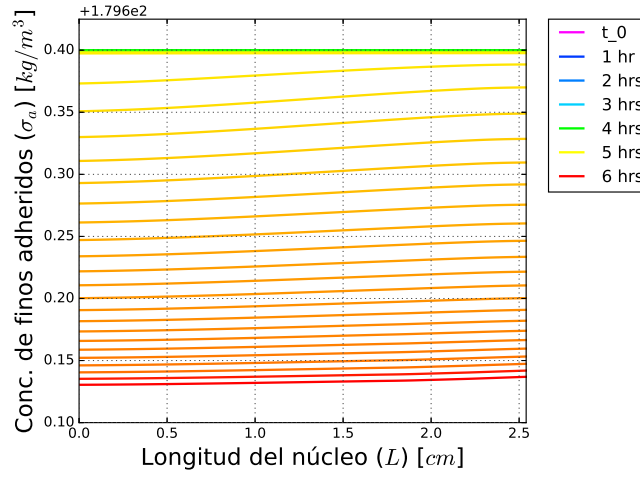


Figura 7.15: Simulación con FEniCS de los perfiles de la concentración de finos adheridos a la roca, a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas.

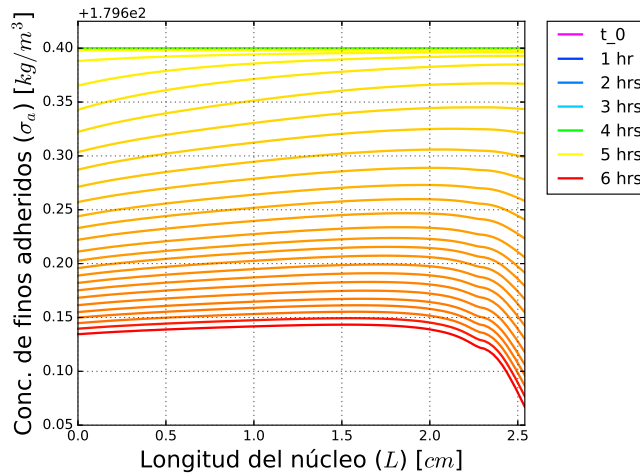


Figura 7.16: Simulación con COMSOL de los perfiles de la concentración de finos adheridos a la roca, a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas.

7.5.4. Simulaciones de la concentración de finos móviles

Las gráficas de las figuras (7.17) y (7.18) describen la evolución en el tiempo de la concentración de los finos móviles en el fluido en la entrada y en la salida del núcleo respectivamente. La gráfica de la figura (7.17) es congruente con la hipótesis donde se supuso que no se inyectan finos por la entrada del núcleo. La gráfica de la figura (7.18) muestra que la concentración de finos móviles en el fluido se mantiene hasta cierto tiempo, después aumenta cuando los finos adheridos a la roca empiezan a desprenderse, pero después disminuye cuando son más los finos que se atorán y tapan los poros, que los finos que se desprenden de la roca.

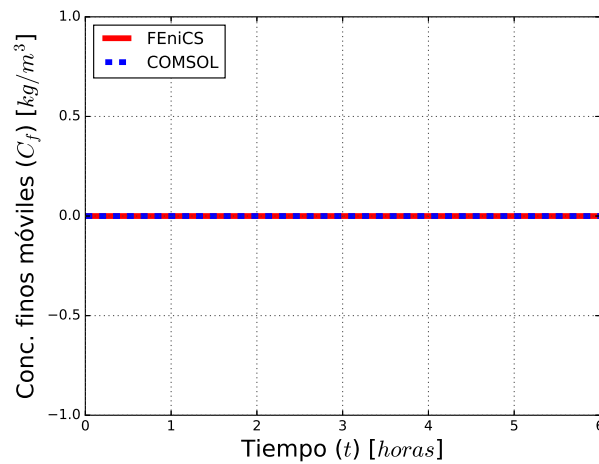


Figura 7.17: Evolución en el tiempo de la conc. de finos móviles en la entrada del núcleo.

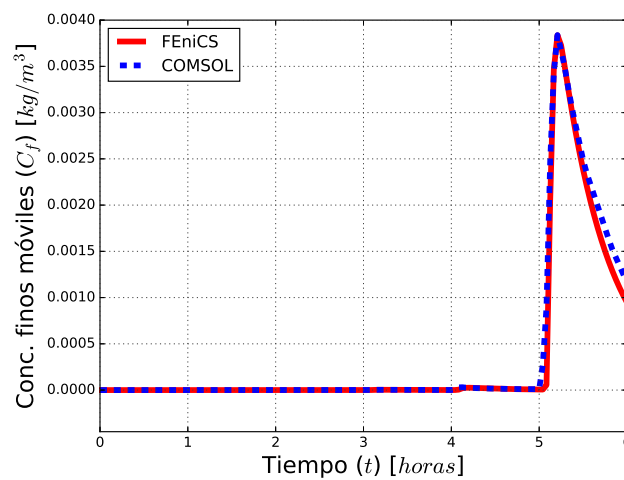


Figura 7.18: Evolución en el tiempo de la conc. de finos móviles en la salida del núcleo.

Las gráficas de las figuras (7.19) y (7.20) representan a los perfiles de la concentración de finos móviles en el fluido a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. La concentración de los finos móviles a lo largo de la longitud del núcleo, se mantiene hasta cierto tiempo, después aumenta cuando los finos comienzan a desprenderse de la roca y finalmente disminuye cuando hay más finos que se atorán tapando los poros, que los finos que se desprenden de la roca.

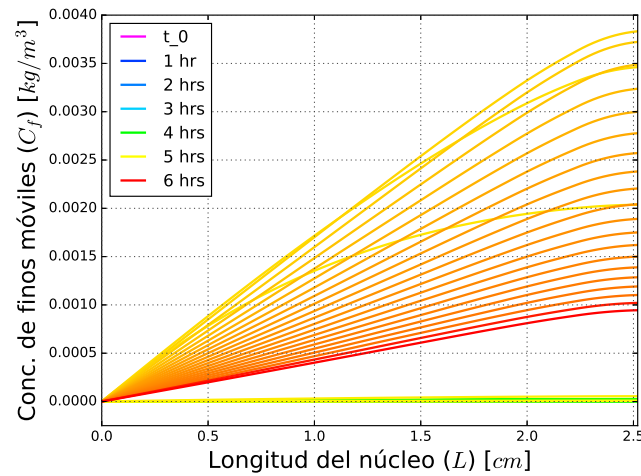


Figura 7.19: Simulación con FEniCS de los perfiles de la concentración de finos móviles en el fluido, a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas.

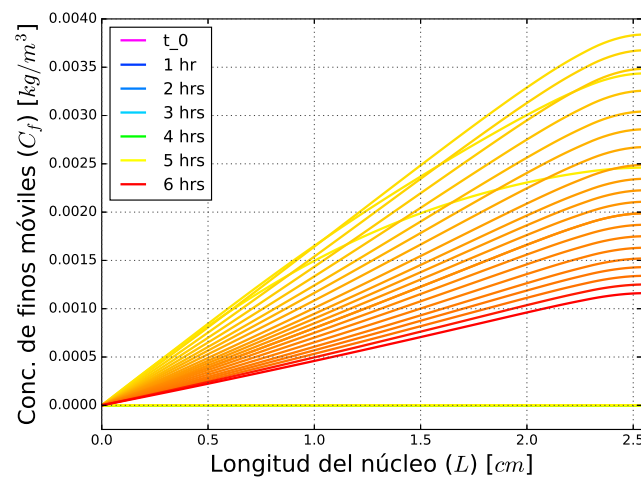


Figura 7.20: Simulación con COMSOL de los perfiles de la concentración de finos móviles en el fluido, a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas.

7.5.5. Simulaciones de la concentración de finos atorados

Las gráficas de las figuras (7.21) y (7.22) describen la evolución en el tiempo de la concentración de los finos atorados en las gargantas de los poros, en la entrada y en la salida del núcleo respectivamente. La gráfica de la figura (7.21) denota que no hay finos atorados en la entrada del núcleo, mientras que la gráfica de la figura (7.22) indica que la concentración de finos atorados es nula o casi nula hasta cierto tiempo, después aumenta a partir de que los finos desprendidos de la roca migran a otros sitios con el fluido y dependiendo de su tamaño, se van quedando atorados en el trayecto, causando el bloqueo de los poros.

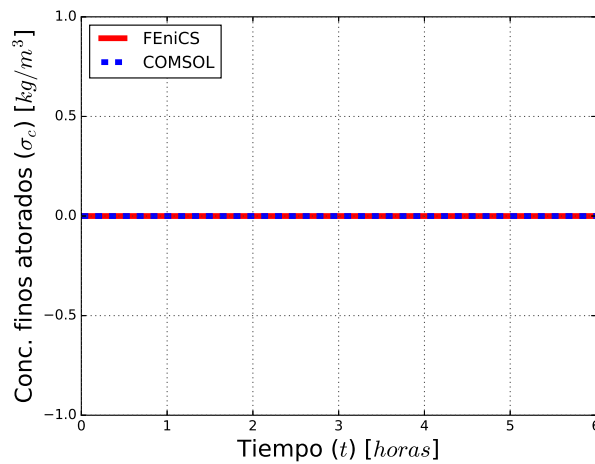


Figura 7.21: Evolución en el tiempo de la conc. de finos atorados en la entrada del núcleo.

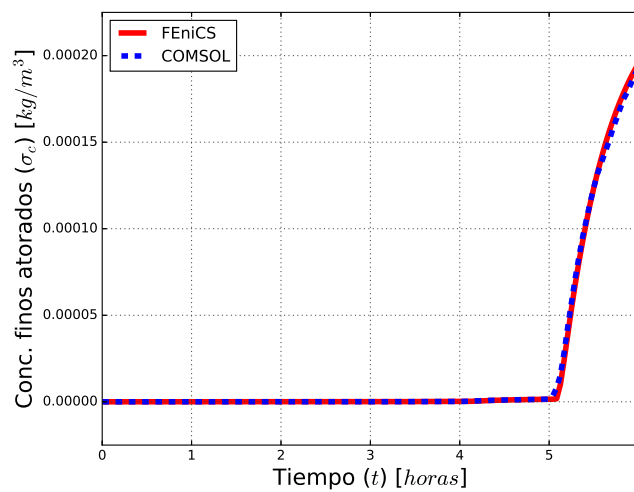


Figura 7.22: Evolución en el tiempo de la conc. de finos atorados en la salida del núcleo.

Las gráficas de las figuras (7.23) y (7.24) representan a los perfiles de la concentración de finos atorados en las gargantas de los poros a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. La concentración de los finos atorados se mantiene nula o casi nula hasta cierto tiempo, después aumenta a partir de que los finos desprendidos de la roca migran con el fluido y entonces dependiendo de su tamaño, se van atorando en las gargantas de los poros en su trayecto a lo largo de la longitud del núcleo.

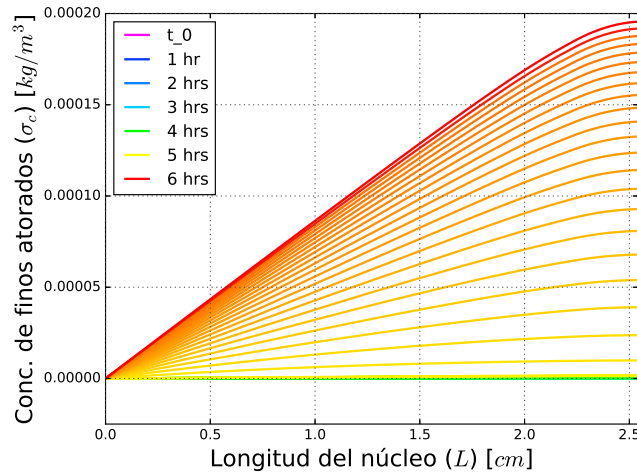


Figura 7.23: Simulación con FEniCS de los perfiles de la concentración de finos que taponan los poros, a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas.

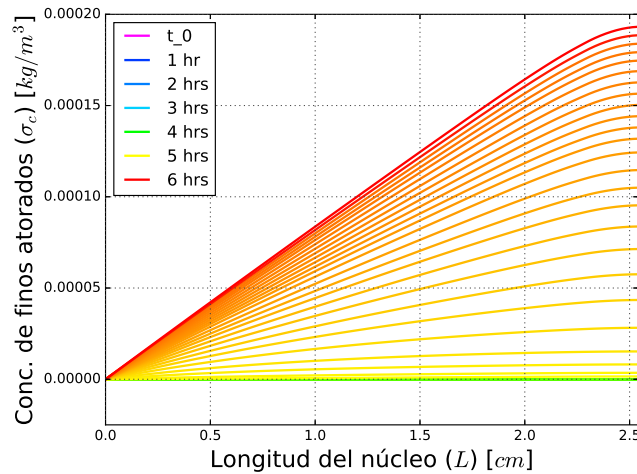


Figura 7.24: Simulación con COMSOL de los perfiles de la concentración de finos que taponan los poros, a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas.

7.5.6. Simulaciones de la permeabilidad

Las gráficas de las figuras (7.25) y (7.26) describen la evolución en el tiempo de la permeabilidad en la entrada y en la salida del núcleo respectivamente. La gráfica de la figura (7.25) es congruente con la hipótesis de que la permeabilidad permanece constante en la entrada del núcleo. La gráfica de la figura (7.26) indica que la permeabilidad se mantiene constante hasta cierto tiempo y comienza a disminuir cuando los finos que migran en el fluido comienzan a atorarse en las gargantas de los poros.

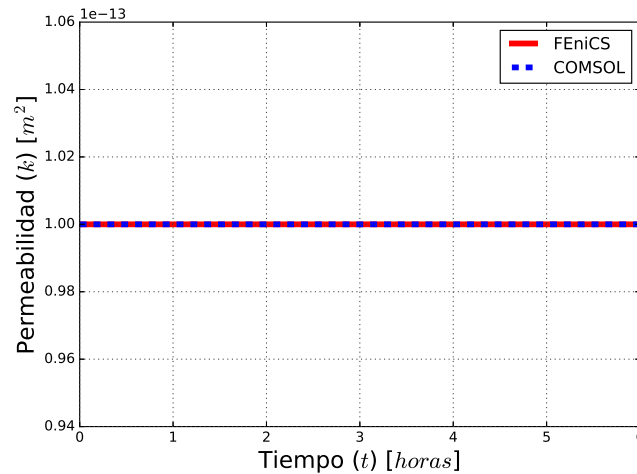


Figura 7.25: Evolución en el tiempo de la permeabilidad en la entrada del núcleo.

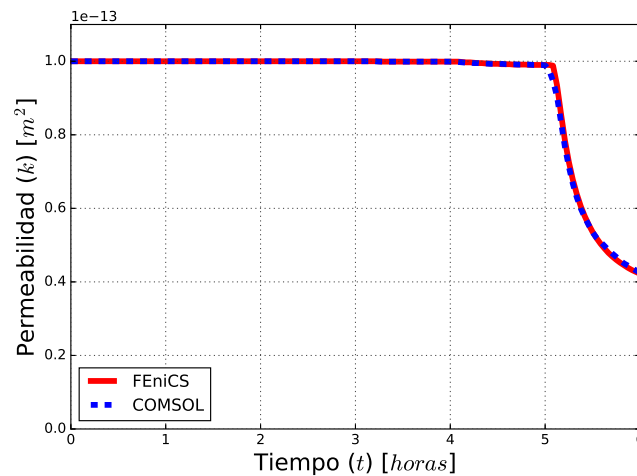


Figura 7.26: Evolución en el tiempo de la permeabilidad en la salida del núcleo.

Las gráficas de las figuras (7.27) y (7.28) representan a los perfiles de la permeabilidad a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. Dependiendo del tamaño de los finos y de la distribución de los poros, la migración de los finos puede provocar el taponamiento de los poros y por el hinchamiento de las arcillas ocurre una reducción de la permeabilidad a lo largo de la longitud del núcleo.

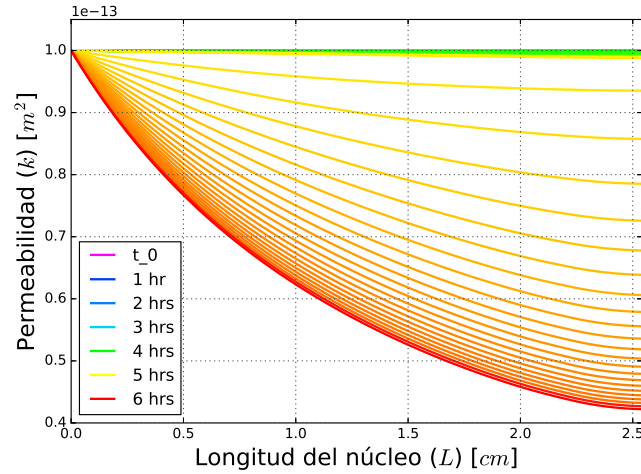


Figura 7.27: Simulación con FEniCS de los perfiles de la permeabilidad a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

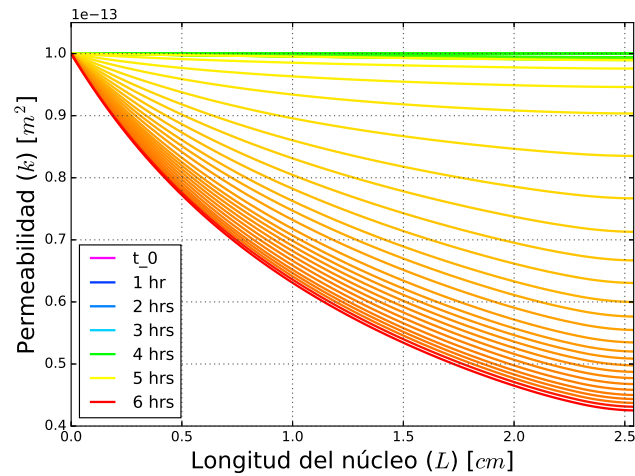


Figura 7.28: Simulación con COMSOL de los perfiles de la permeabilidad a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

7.5.7. Simulaciones de la presión

La gráfica de la figura (7.29) describe la evolución en el tiempo de la presión en la entrada del núcleo, mientras que la gráfica de la figura (7.30) describe la evolución en el tiempo de la caída de presión en el núcleo. La presión se mantiene hasta cierto tiempo, después aumenta a partir de que la permeabilidad disminuye debido al bloqueo de los poros por los finos y como se impuso un gasto constante de agua, entonces hay un incremento en la presión para mantener el gasto constante que se le impuso y en consecuencia hay un aumento en la diferencia de presión entre la entrada y la salida del núcleo.

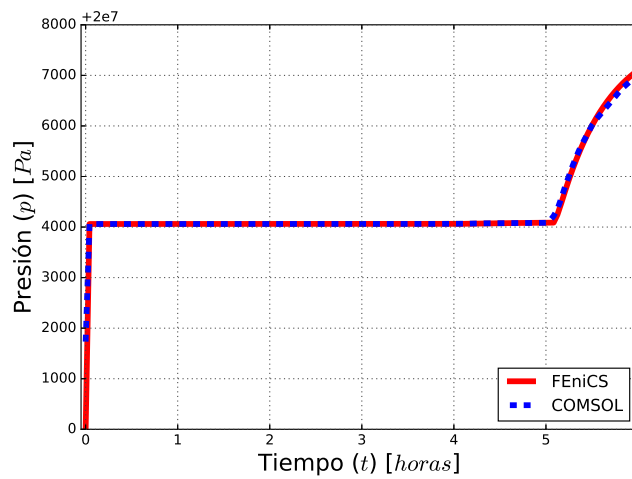


Figura 7.29: Evolución en el tiempo de la presión en la entrada del núcleo.

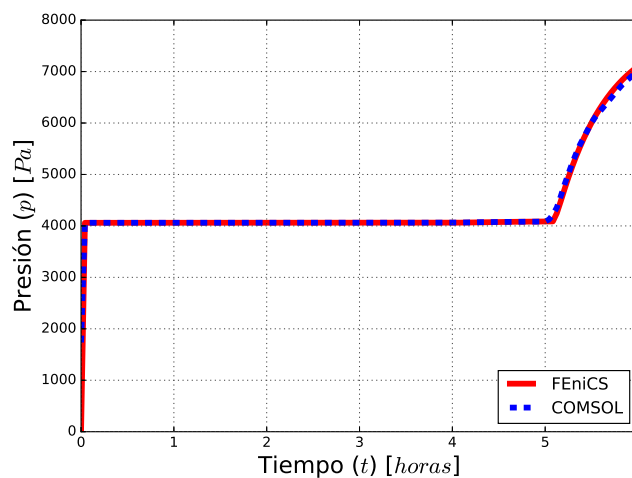


Figura 7.30: Evolución en el tiempo de la caída de presión en el núcleo.

Las gráficas de las figuras (7.31) y (7.32) representan a los perfiles de la presión del fluido a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. Las gráficas de los perfiles de la presión son coherentes con las condiciones de frontera impuestas: gasto constante en la entrada del núcleo y presión constante en la salida del núcleo, por eso los perfiles de las presiones cambian en la entrada, pero no en la salida. El perfil de la presión inicial en COMSOL es diferente al impuesto como condición inicial, porque COMSOL estimó una primera solución inicial para poder resolver el sistema no-lineal de ecuaciones diferenciales parciales, mientras que FEniCS respeta la condición inicial que se le impuso.

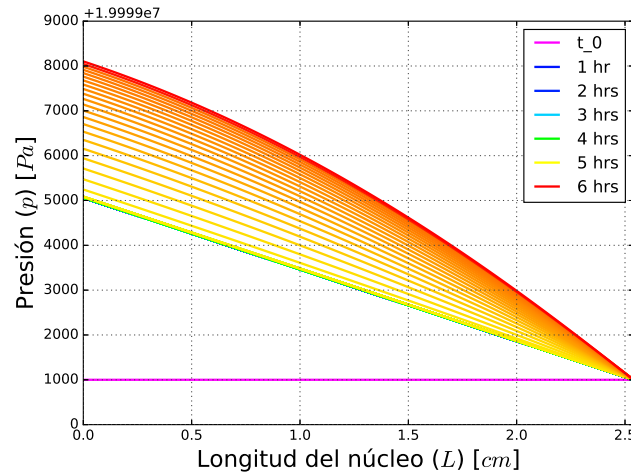


Figura 7.31: Simulación con FEniCS de los perfiles de la presión del fluido a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

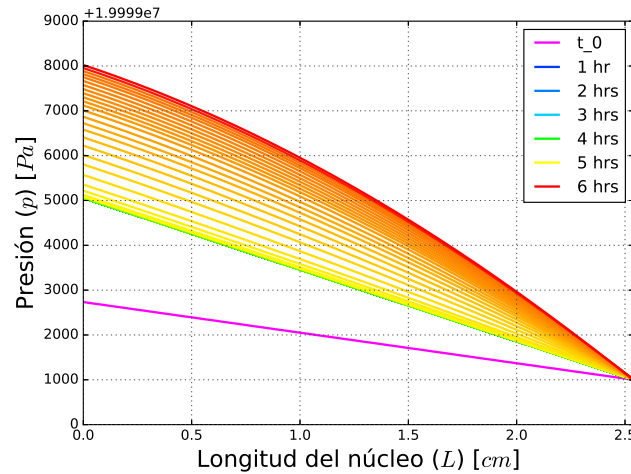


Figura 7.32: Simulación con COMSOL de los perfiles de la presión del fluido a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

7.5.8. Simulaciones de la velocidad

Las gráficas de las figuras (7.33) y (7.34) describen la evolución en el tiempo de la velocidad del fluido en la entrada y en la salida del núcleo respectivamente. La gráfica de la figura (7.33) indica que la velocidad del fluido es constante en la entrada del núcleo como se impuso en la condición de frontera. La gráfica de la figura (7.34) indica que la velocidad del fluido es relativamente constante en la salida del núcleo. Por lo tanto, las gráficas son coherentes con el planteamiento del problema.

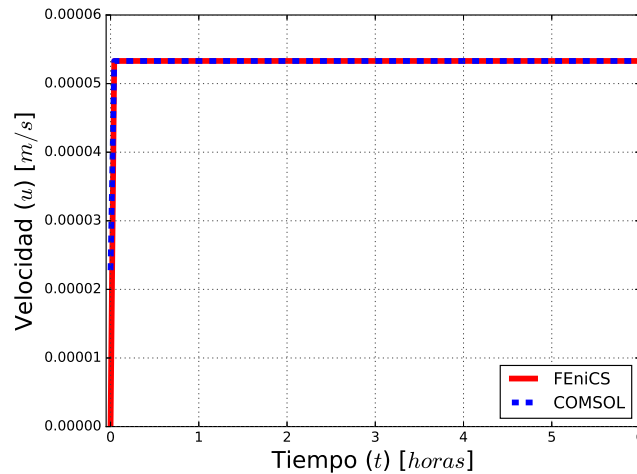


Figura 7.33: Evolución en el tiempo de la velocidad del fluido en la entrada del núcleo.

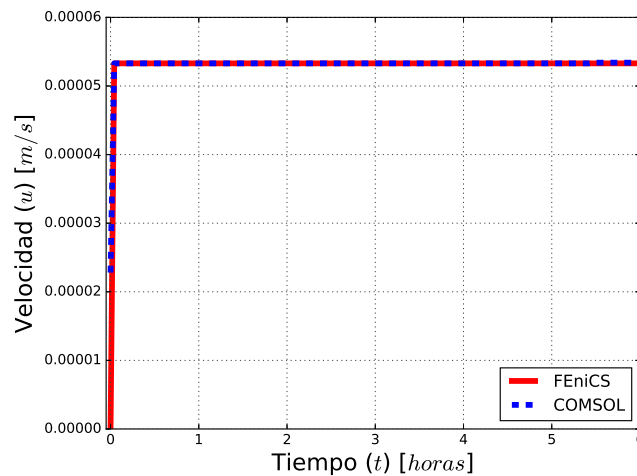


Figura 7.34: Evolución en el tiempo de la velocidad del fluido en la salida del núcleo.

Las gráficas de las figuras (7.35) y (7.36) representan los perfiles de la velocidad del fluido a lo largo de la longitud del núcleo, tomados a varios tiempos, obtenidos con FEniCS y COMSOL respectivamente. El perfil de la velocidad inicial en COMSOL es diferente al que se obtiene con FEniCS, el cual es coherente con el planteamiento del problema, porque COMSOL lo calcula a partir de la presión inicial que calculó anteriormente (ver figura 7.32).

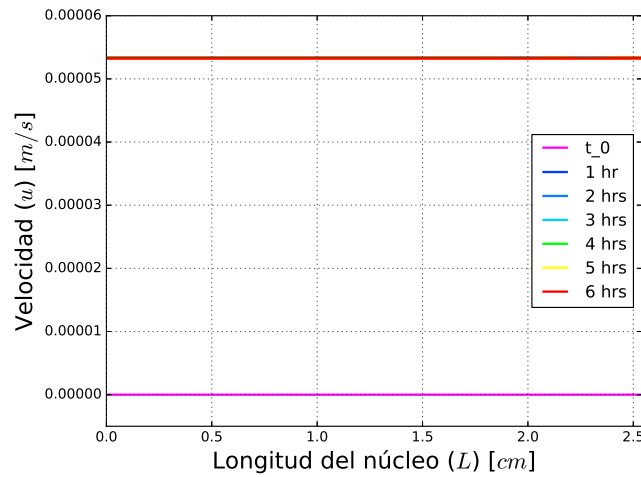


Figura 7.35: Simulación con FEniCS de los perfiles de la velocidad del fluido a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

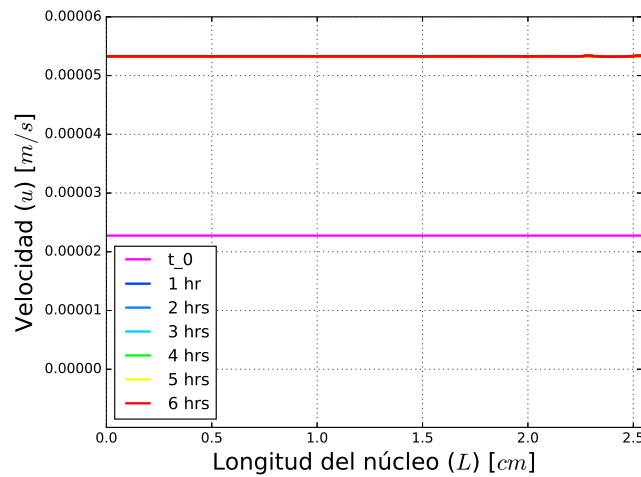


Figura 7.36: Simulación con COMSOL de los perfiles de la velocidad del fluido a lo largo de la longitud del núcleo, tomados cada 2.5 minutos durante 6 horas de inyección de agua.

7.5.9. Simulaciones de la permeabilidad efectiva

La gráfica de la figura (7.37) describe la evolución en el tiempo de la permeabilidad efectiva, la cual permanece constante hasta cierto tiempo, después disminuye cuando comienza a incrementarse la caída de la presión en el núcleo, debido al bloqueo de los poros por los finos y a que debe mantener el gasto constante de agua que se le impuso.

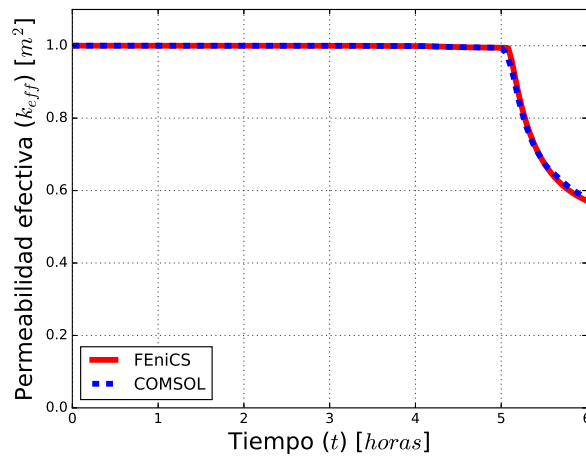


Figura 7.37: Evolución en el tiempo de la permeabilidad efectiva en la entrada del núcleo.

7.6. Análisis de resultados

La aplicación del modelo de flujo monofásico y transporte multicomponente en medios porosos propuesto en la presente tesis, consistió en resolver numérica y computacionalmente, con el método de elementos finitos mixtos, las ecuaciones de un modelo matemático que describe el proceso de la migración de finos y la pérdida de permeabilidad causada por el agua de baja salinidad tomado de la literatura, empleando las características y propiedades del núcleo y del fluido, comparando los resultados con los obtenidos con el software COMSOL.

El sistema de ecuaciones está acoplado dinámicamente, ya que al reducir lo suficiente la salinidad de la salmuera de inyección, provoca que los finos adheridos a la roca comiencen a desprenderse, generando finos móviles que migran con el fluido, los cuales dependiendo de su tamaño y de la distribución de los poros, se van quedando atorados en el trayecto, tapando los poros y bloqueando el flujo, por lo que se reduce la permeabilidad absoluta del medio y que a su vez se manifiesta en un aumento de la presión para mantener el gasto constante que se le impuso, lo que implica la velocidad advectiva, la cual es un ingrediente importante en la ecuación de advección-dispersión de la salinidad y de los finos móviles.

Los resultados obtenidos con FEniCS y COMSOL son semejantes, aunque los de FEniCS son más coherentes con el planteamiento y la dinámica del problema. Por lo tanto, el método de elementos finitos mixtos, además de obtener una mejor aproximación numérica de la presión y la velocidad, también mejoró la aproximación numérica de las demás variables.

Conclusiones

Un hecho que hay que destacar es que se logró desarrollar un modelo que tiene una serie de características notables entre las que se encuentran que es un sistema de ecuaciones diferenciales parciales parabólicas generales de segundo orden no-lineales con condiciones de frontera generales de tipo Dirichlet y Neumann. Además se obtuvo su formulación variacional aplicando el método de elementos finitos mixtos, para poderlo implementar computacionalmente en un script mediante el lenguaje de programación Python 2.7 usando FEniCS Project 2017.1 el cual es libre, de código abierto y multiplataforma (Linux, Windows, Mac) con el objetivo de modelar flujo y transporte en medios porosos para simular procesos de recuperación mejorada. Este modelo es fácilmente adaptable a cualquier problema a escala de laboratorio, donde el flujo es monofásico ligeramente compresible y el transporte es multicomponente, únicamente cambiando los valores de los parámetros.

La solución numérica del sistema de ecuaciones diferenciales, se puede obtener discretizando el espacio mediante el método de elementos finitos mixtos; mientras que para la discretización del tiempo, por simplicidad y estabilidad, se puede aplicar el método de diferencias finitas hacia atrás, resultando un esquema completamente implícito.

Se realizó una ardua labor de investigación para proponer la formulación variacional mixta del sistema de ecuaciones del modelo y posteriormente hacer su implementación computacional, porque en la literatura no se encuentra un sólo ejemplo de un modelo que reúna todas las características mencionadas anteriormente, por lo tanto la descripción de cada etapa del desarrollo del modelo puede servir como ejemplo de referencia para resolver modelos con características similares, por el método de elementos finitos mixtos e implementarlos computacionalmente en Python usando FEniCS, en dos o tres dimensiones.

El modelo se aplicó exitosamente para simular un proceso de desprendimiento, migración y bloqueo por finos, en un caso de estudio de inyección de agua de baja salinidad (LSWF por sus siglas en inglés), utilizando datos tomados de la literatura.

Como en algunas aplicaciones, una variable vectorial (por ejemplo, la velocidad del fluido) es la variable primaria en la que se está interesado entonces una ventaja que tienen los métodos de elementos finitos mixtos sobre el método estándar, es que pueden calcular simultáneamente esta variable vectorial y una variable escalar (por ejemplo, la presión), dando un orden de aproximación alto en ambas variables, empleando dos espacios diferentes de elementos finitos que deben satisfacer una condición inf-sup también conocida como condición de Ladyzhenskaya-Babushka-Brezzi, para que el método mixto sea estable, porque no todas las combinaciones de espacios son compatibles y estables, pero en particular,

las combinaciones de espacios que proponemos en el presente trabajo, si son compatibles y estables.

Se compararon tres diferentes formulaciones del método de elementos finitos: la clásica, la mixta y la mixta-dual. Una combinación estable de espacios de elementos finitos mixtos para la formulación variacional mixta son los elementos de Brezzi-Douglas-Marini (BDM) de orden polinomial k para aproximar la velocidad y los elementos discontinuos de Galerkin (DG) de orden polinomial $k - 1$ para aproximar la presión; mientras que una combinación estable de espacios de elementos finitos mixtos para la formulación variacional mixta-dual son, los elementos discontinuos de Raviart-Thomas (DRT) de orden polinomial k para aproximar la velocidad y los elementos continuos de Galerkin (CG) de orden polinomial $k + 1$ para aproximar la presión.

Las tres formulaciones fueron validadas para un caso de referencia y se realizó una comparación en términos del orden de convergencia y desempeño, resultando ser más óptimo el método mixto, mostrando una mejor aproximación tanto en la velocidad como en la presión; Sin embargo, se observó que tanto en una formulación variacional estándar como en una formulación variacional mixta-dual, las condiciones de frontera naturales son las de tipo Neumann y las condiciones de frontera esenciales son las de tipo Dirichlet, contrario a la formulación variacional mixta donde las condiciones de frontera naturales son las de tipo Dirichlet y las condiciones de frontera esenciales son las de tipo Neumann. Tal observación hizo posible acoplar el modelo de flujo monofásico con el modelo de transporte multicomponente, combinando la formulación variacional mixta-dual de la ecuación de flujo con la formulación variacional estándar del modelo de transporte multicomponente y resolverlas simultáneamente en FEniCS.

Se validó en una comparación con un caso de estudio para la simulación de migración de finos en un proceso de inyección de agua de baja salinidad a escala de laboratorio con el software comercial COMSOL Multiphysics®, dando resultados muy consistentes con un buen desempeño, una buena aproximación en todas las incógnitas y una mejor aproximación en la velocidad, debido a que se trabajó con los elementos mixtos, lo cual no se puede hacer con el software comercial mencionado.

Cada uno de los modelos obtenidos en el presente trabajo fue desarrollado siguiendo la metodología sistemática de la *modelación matemática numérica y computacional*, porque es una metodología conveniente, adecuada y apropiada para desarrollar modelos.

El trabajo presente puede servir como base para desarrollar modelos multifásicos y multicomponentes de flujo y transporte en medios porosos multiescala, con un gran potencial en su aplicación a la modelación de yacimientos heterogéneos o pruebas de recuperación mejorada a escala de laboratorio.

Como trabajo futuro, se desea obtener un modelo de flujo bifásico agua-aceite, un método de paso adaptativo en el tiempo, un módulo de optimización para realizar el ajuste de datos, un módulo de conversión de unidades y una interfaz gráfica para los usuarios.

Apéndices

Apéndice A

Proceso de inyección de agua de baja salinidad a escala de laboratorio

En el presente apéndice se describirán las condiciones necesarias para propiciar el efecto de recuperación adicional por inyección de agua de baja salinidad, conocida internacionalmente como LSWF por sus siglas en inglés (*Low Salinity Water Flooding*), con el propósito de comprender mejor los fenómenos físicos de de dicho proceso.

A.1. Mecanismos del proceso LSWF

En presencia de rocas con contenido de arcilla y con salmuera de formación de alta salinidad (con cationes divalentes Ca^{++} o Mg^{++}) la inyección del agua de baja salinidad a tasa constante provoca producción de aceite y cambios en la diferencia de presión entrada-salida, en el pH, en la composición iónica de la salmuera y en su contenido de finos. El fenómeno está frecuentemente asociado al cambio de mojabilidad de la roca, la cual cambia a más mojabable por agua. Los mecanismos propuestos en la literatura son los siguientes [17], [12], [26].

A.1.1. Desprendimiento y migración de finos

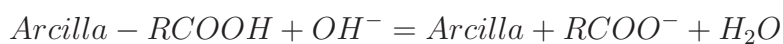
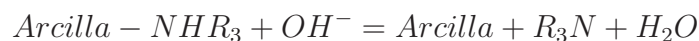
La baja salinidad provoca la hidratación e hinchamiento de las arcillas, cuando la fuerza iónica de la salmuera es menor que la concentración de floculación crítica se produce el desprendimiento de finos. Dependiendo de la distribución de tamaños de los finos y de los poros, la migración de los finos puede provocar el taponamiento de los poros y por el hinchamiento de las arcillas hay una reducción de la permeabilidad, incrementando la diferencia de presión entre la entrada y la salida del núcleo. El bloqueo de los principales caminos de flujo obliga al fluido de inyección a tomar otros caminos dentro del medio poroso, barriendo otras zonas del núcleo y con ello aumenta la producción adicional de aceite (ver fig. 7.1).

Existen experimentos que muestran correlación de la presencia de finos con la recuperación de aceite, pero hay otros que encuentran gran producción de finos sin el efecto LSWF [5], [31], [30] y otros que encuentran el efecto LSWF sin producción de finos [17].

A.1.2. Contribución del pH

Generación de tensoactivos in-situ: Para el aumento de pH es necesaria la presencia de carbonatos [19]. La disolución lenta de carbonatos crean exceso de OH^- y ocurre el intercambio catiónico rápido entre minerales de la arcilla y el agua con una reducción de H^+ provocando el aumento de pH. Si el $pH > 9$ la interacción con el aceite puede provocar la generación de tensoactivo in-situ, lo cual libera la tensión interfacial y con ello se libera aceite absorbido en la superficie arcillosa; sin embargo, experimentalmente no se observan aumentos de pH tan altos.

Desorción de aceite con aumento de pH:[3] [12] Inicialmente material orgánico y básico están adsorbidos en la arcilla conjuntamente con cationes del agua de formación (especialmente Ca^{++}). El agua de baja salinidad provoca la desorción de cationes y con ello incrementa el pH cerca de la superficie de arcilla al ser sustituido el catión por H^+ del agua, provocando el desprendimiento de aceite y el cambio de mojabilidad de la roca al perder orgánicos adsorbidos.



En este caso el aumento de pH es un efecto lateral, no el causante del cambio.

A.1.3. Efecto Salting-in

La solubilidad de compuestos orgánicos polares en el agua aumenta al reducir la salinidad. La nube de moléculas de hidrocarburos localizada alrededor de una gota de aceite crece al reducir la cantidad de iones en el agua (el agua rechaza” menos de su interior a las moléculas orgánicas). Por ello, la reducción de la salinidad baja la tensión interfacial agua-aceite y parcialmente desorbe moléculas orgánicas débilmente amarradas a la arcilla.

A.1.4. Intercambio iónico multicomponente

En rocas con arcilla y con agua congénita de alto contenido de cationes divalentes (Ca^{++} , Mg^{++}) el aceite está fuertemente adsorbido en la superficie arcillosa de la roca a través de sus partes polares (asfaltenos y resinas). Esto tiende a hacer a la roca mojabla por aceite. Con la inyección de agua de baja salinidad ocurre que al haber menos iones en el agua, las moléculas de agua compiten con superioridad respecto al aceite por la superficie de la arcilla, desplazan los cationes divalentes y desprenden el aceite adsorbido. La reacción se expresaría como [19]



A.1.5. Alteración de la mojabilidad

Los cambios de mojabilidad de la roca hacia mayormente mojada por agua parecen explicar de forma general el efecto LSWF. Varios mecanismos dan lugar a esta alteración debido a su capacidad de quitar compuestos orgánicos de la superficie de la arcilla, desprendimiento de finos y el intercambio iónico, pero adicionalmente, la expansión de la doble capa eléctrica por la reducción de la concentración de iones en el agua, lo cual provoca el desprendimiento de aceite de la superficie.

A.2. Condiciones necesarias para el efecto LSWF

Las condiciones necesarias para que tenga lugar el fenómeno de recuperación adicional por inyección de agua de baja salinidad son [3]:

1. En el **medio poroso**, las areniscas deben contener arcilla. El tipo de arcilla puede ser importante.
2. El **aceite** debe contener componentes polares (ácidos o bases). No se ha observado el efecto en aceites refinados sin componentes polares.
3. La **salmuera de formación** debe contener cationes divalentes, por ejemplo Ca^{++} , Mg^{++} , además se debe tener agua de formación inicialmente en el proceso. La eficiencia del proceso está ligada a la saturación inicial del agua, S_{wi} .
4. **Agua de inyección de baja salinidad**. La salinidad usualmente tiene entre 1000 y 2000 ppm, pero se ha observado el efecto hasta 5000 ppm. éste parece ser sensible a la composición iónica.

5. Para sistemas sin control de pH, el **agua producida o de salida** usualmente aumenta de 1 a 3 unidades de pH cuando se inyecta agua de baja salinidad; sin embargo se ha observado que el aumento de pH no está ligado a la recuperación de aceite. En algunos casos se ha detectado la producción de finos, pero también se ha observado el efecto LSWF sin producción visible de finos.
6. **Decremento en la permeabilidad:** Al cambiar a agua de baja salinidad, usualmente se ha observado un incremento en la diferencia de presión, lo cual puede estar relacionado con la migración de finos o la formación de emulsión aceite-agua. No hay suficiente evidencia para experimental para concluir que el efecto de recuperación está acompañado de reducción de permeabilidad. Se han reportado experimentos de invasión de agua donde no se encuentra variación en los puntos finales de la permeabilidad relativa entre invasiones de alta y baja salinidad en modo de inyección secundaria o terciaria.
7. No hay limitaciones por la **temperatura**, la mayoría de los reportes mencionan temperaturas menores a 100°C .

A.3. Mecanismos presentes en el efecto LSWF

A continuación se explican los mecanismos y procesos reportados en la literatura que explican el fenómeno de recuperación adicional de aceite por inyección de agua de baja salinidad en areniscas.

1. **Hinchamiento de arcillas:** Las arcillas se hidratan e hinchan en presencia de agua pero el fenómeno se reduce cuando el agua contiene sal.
2. **Desprendimiento y migración de finos:** El desprendimiento ocurre cuando la fuerza iónica de la salmuera es menor a la concentración crítica de floculación (CFC). Este valor crítico aumenta con la concentración relativa de cationes divalentes. Estos cationes tienen un efecto estabilizador de finos en la superficie de la arcilla ya que reducen la fuerza repulsiva, es decir, reducen el potencial Zeta.
3. **Reducción o taponamiento de caminos de flujo:** El hinchamiento de las arcillas reduce la sección transversal disponible para el flujo de fluidos, mientras que los finos desprendidos pueden ser retenidos en los poros y gargantas pequeñas, bloqueando el paso de fluidos, provocando que el fluido se mueva por otros caminos, barriendo el aceite que ahí se encuentra. Ni el hinchamiento ni el bloqueo de caminos cambian la porosidad, pues el volumen ocupado por la arcilla es el mismo. El efecto es sobre la permeabilidad total $K_T = kK_r$, pero se asume que no afecta a la permeabilidad relativa, K_r si no a la permeabilidad absoluta k .
4. **No-efecto de los finos en la recuperación de aceite:** Se ha reportado recuperación de aceite sin producción de finos ni reducción de permeabilidad y viceversa, producción de finos, sin recuperación de aceite [17] [26] [32] [28].
5. **Cambios por desprendimiento de finos:** El desprendimiento de finos provoca que la superficie de la roca quede expuesta y tenga otras propiedades de mojabilidad.

6. **Intercambio iónico entre la arcilla y la roca:** En rocas con agua congénita de alto contenido de cationes divalentes (Ca^{++} , Mg) el aceite está fuertemente adsorbido en la superficie arcillosa de la roca a través de sus partes polares (asfaltenos y resinas). Esto tiende a hacer a la roca mojabable por aceite. Con la inyección de agua con bajo contenido de iones ocurre que al haber menos iones en el agua, las moléculas del agua compiten con superioridad contra el aceite por la superficie de la arcilla, desplazan los cationes divalentes y desprenden el aceite adsorbido.
7. **Cambios por desprendimiento de aceite:** El desprendimiento de aceite deja expuesta la superficie que es más mojabable al agua que la previa superficie.
8. **Expansión de doble capa eléctrica:** La reducción de la salinidad incrementa el tamaño de la doble capa eléctrica en las interfaces roca-salmuera y aceite-salmuera, modificando la carga eléctrica sobre la superficie de la roca y su mojabilidad. Como el incremento de pH actúa en contra de este efecto, entonces un pH bajo combinado con una baja salinidad produce recuperación adicional de aceite [20].
9. **Incremento de solubilidad en orgánicos (salting-in):** La solubilidad de compuestos orgánicos polares aumenta al reducir la salinidad. La nube de moléculas de hidrocarburos flotando alrededor de una gota de aceite crece al reducir la cantidad de iones en el agua (el agua "expulsa" menos de su interior a las moléculas orgánicas). Esto parcialmente desorbe moléculas orgánicas débilmente amarradas a la arcilla.
10. **Aumento de pH:** Es necesaria la presencia de carbonatos. La disolución lenta de carbonatos crea exceso de OH^- e intercambio catiónico rápido entre minerales de la arcilla y el agua (los minerales intercambian cationes adsorbidos por H^+), provocando una reducción de H^+ y en consecuencia un aumento del pH.
11. **Generación de tensoactivos in-situ:** Los tensoactivos modifican la superficie de la roca haciéndola más mojabable al agua [19]. Se requieren números ácidos mayores a 0.2 KOH/g para generar tensoactivos in-situ (regularmente es menor a 0.05 KOH/g [12]). El mecanismo es similar a la recuperación por invasión alcalina, incremento de pH y reducción de tensión interfacial [19]. Con $pH > 9$ los componentes ácidos o polares del aceite se saponifican por reacciones químicas.
12. **Efectos del cambio de tensión interfacial agua-aceite:** La reducción de la tensión interfacial reduce la presión capilar y la permeabilidad relativa a través de modificaciones en la saturación residual de aceite. Asimismo, la baja en la tensión interfacial promueve la formación de emulsiones. La emulsión cambia la viscosidad de la fase continua. Si es emulsión de aceite agua, la viscosidad del agua aumenta.
13. **Adsorción/Desorción sin puentes de cationes divalentes:** Los componentes polares del aceite se pueden adsorber en los minerales de la arcilla sin puentes de cationes divalentes y la reducción del contenido de magnesio es causado por precipitación, especialmente a alto pH [3] [1].
14. **Aumento de pH desprende material orgánico:** Originalmente el material básico y orgánico está adsorbido en la arcilla por cationes, principalmente Ca^{++} del agua de

formación, con el agua de baja salinidad el H^+ es intercambiado por Ca^{++} y con ello aumenta localmente el pH. Esto provoca cerca de la superficie de la arcilla reacciones entre el material ácido y básico adsorbido (reacción de transferencia protónica clásica). Se produce una reacción rápida entre OH^- y el material ácido básico adsorbido y con ello el material orgánico se desorbe de la arcilla [3].

15. **No-efecto del pH en la recuperación de aceite** [17] [26]: Se necesitan valores no suficientemente altos del pH para generar tensoactivos. Hay recuperación de aceite tanto con incrementos como con decrementos de pH.
16. **Cambio en la permeabilidad relativa:** La baja salinidad cambia los puntos finales de las curvas de permeabilidad relativa, se tiene menor S_{wi} y mayor S_{or} [29] [24] [13].
17. **Cambio de viscosidad del agua:** La presencia de finos modifica la viscosidad del agua y puede llegar a formarse un gel.

Apéndice B

Modelación matemática de los sistemas continuos

B.1. Teoría de los sistemas continuos

Los **sistemas continuos** están constituidos por cuerpos. Un **cuerpo** es un conjunto infinito de puntos materiales que en cualquier instante dado ocupa una región o dominio, en el sentido matemático, del espacio tridimensional.

Un **punto espacial** es un punto fijo en el espacio y un **punto material** es un punto que puede ocupar distintos puntos espaciales en su movimiento a lo largo del tiempo.

Hipótesis básica de los sistemas continuos Consiste en considerar que un sistema continuo llena todo el espacio que ocupa, es decir, cada punto del sistema continuo está lleno de materia. En la teoría de los sistemas continuos se trabaja con los promedios de sus propiedades físicas y existe un volumen llamado representativo, para el cual se calculan y son válidos los promedios de dichas propiedades.

Sea \mathbf{B} un cuerpo y $B(t)$ el dominio ocupado por \mathbf{B} en el instante t . Debido a la hipótesis básica de los sistemas continuos, en cualquier instante de tiempo t fijo, y en cada punto $\underline{x} \in B(t)$ de la región ocupada por el cuerpo, hay solo un punto material del cuerpo \mathbf{B} (ver figura B.1).

La **configuración espacial** del cuerpo \mathbf{B} en el instante t , es el lugar geométrico de las posiciones que ocupan en el espacio los puntos materiales del cuerpo en dicho instante. Se le denomina **configuración de referencia** a la configuración espacial en el instante inicial t_0 del intervalo de interés.

Coordenadas materiales o lagrangianas: Son las coordenadas del vector de posición $\underline{X} = (X_1, X_2, X_3)$ que ocupa un punto material del cuerpo \mathbf{B} en el instante inicial t_0 en la configuración de referencia.

Coordenadas espaciales o eulerianas: Son las coordenadas del vector de posición $\underline{x} = (x_1, x_2, x_3) = \underline{p}(\underline{X}, t) = (p_1, p_2, p_3)$ que ocupa el punto material \underline{X} del cuerpo \mathbf{B} en el instante t en la configuración espacial.

Al haber una relación biunívoca, podemos definir la inversa de la función $\underline{p}(\underline{X}, t)$ a la cual denotaremos por $\underline{p}^{-1}(\underline{x}, t) \equiv \underline{X}$. La región ocupada por el cuerpo \mathbf{B} cambia con el tiempo debido al movimiento y la definiremos formalmente para cualquier instante t en el tiempo como

$$B(t) \equiv \{ \underline{x} \in \mathbb{R}^3 \mid \exists \underline{X} \in \mathbf{B} \text{ y } \underline{x} = \underline{p}(\underline{X}, t) \}.$$

Si fijamos el tiempo t entonces $B(t) = \underline{p}(\mathbf{B}, t)$. Para observar la trayectoria de un punto material \underline{X} , sólo debemos fijar dicho punto y variar t en la función $\underline{p}(\underline{X}, t)$, lo cual permite obtener la velocidad $\underline{V}(\underline{X}, t)$ de cualquier punto material \underline{X} , importante para describir su movimiento. La velocidad se define como la derivada con respecto al tiempo a la posición cuando el punto material se mantiene fijo, es decir,

$$\underline{V}(\underline{X}, t) = \frac{\partial}{\partial t} \underline{p}(\underline{X}, t)$$

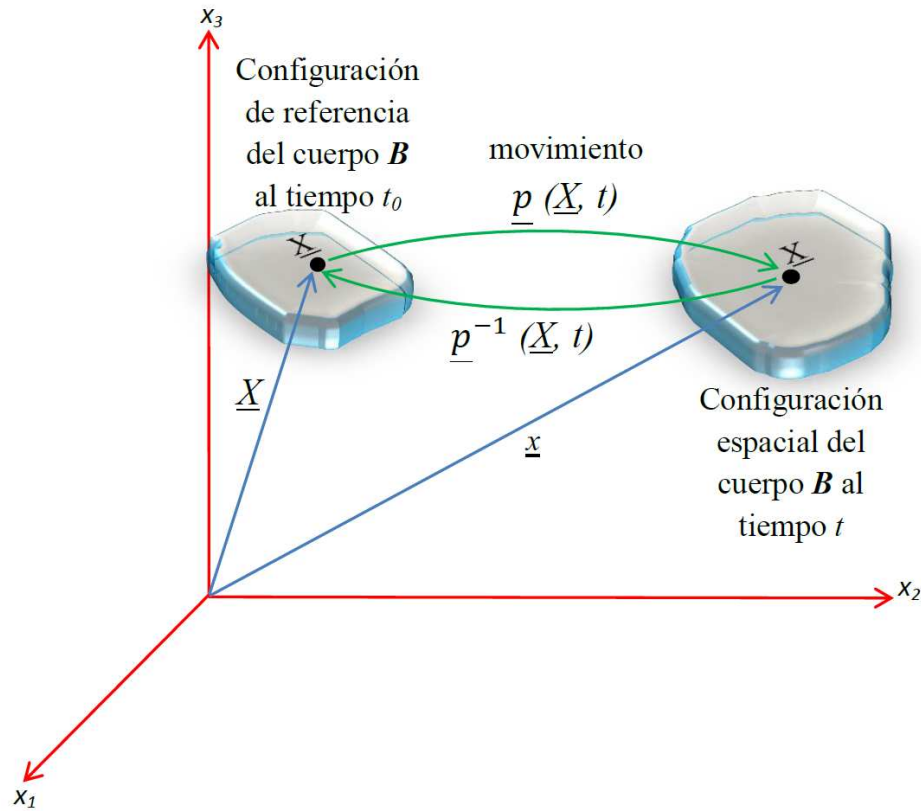


Figura B.1: Esquema de la relación entre la configuración de referencia del cuerpo B , las coordenadas materiales \underline{X} y las coordenadas espaciales \underline{x} de un punto material $\underline{X} \in B$.

B.2. Propiedades intensivas

Son propiedades muy específicas que no dependen de la cantidad de la materia, por ejemplo, la densidad de masa o la densidad de volumen y que pueden expresarse como funciones definidas para cada tiempo en cada uno de los puntos del sistema continuo, es decir, son funciones que están definidas en la posición \underline{x} del punto material \underline{X} al tiempo t . Pueden ser funciones escalares como la concentración de cierta sustancia al tiempo t o vectoriales como la velocidad, que depende del punto material \underline{X} y del tiempo t . Una propiedad intensiva con valores vectoriales es equivalente a tres escalares, correspondientes a cada una de sus tres componentes. En resumen, las propiedades intensivas son funciones definidas en los puntos materiales de un cuerpo, es decir, son funciones que hacen corresponder a cada punto material y cada tiempo en un número real o un vector del espacio Euclidiano tridimensional \mathbb{R}^3 . Hay dos formas de representar a las propiedades intensivas: la representación euleriana y la representación lagrangiana.

B.2.1. Representación Lagrangiana

Consideremos una propiedad intensiva escalar, tal que en el instante t toma en el punto material \underline{X} el valor

$$\phi(\underline{X}, t)$$

definiendo una función $\phi : \mathbf{B} \rightarrow \mathbb{R}^1$ para cada instante t , a la que se le llama representación lagrangiana de la propiedad intensiva considerada. Esta representación es más utilizada en el estudio de los sólidos.

B.2.2. Representación Euleriana

Consideremos una propiedad intensiva, tal que en el instante t toma en el punto material que ocupa la posición \underline{x} el valor

$$\psi(\underline{x}, t)$$

definiendo una función $\psi : B(t) \rightarrow \mathbb{R}^1$ para cada instante t , a la que se le llama representación euleriana de la función considerada. Esta representación es más utilizada en el estudio de los fluidos.

Ambas representaciones satisfacen las siguientes identidades

$$\phi(\underline{X}, t) \equiv \psi(\underline{p}(\underline{X}, t), t) \quad y \quad \psi(\underline{x}, t) \equiv \phi(\underline{p}^{-1}(\underline{X}, t), t)$$

Por ejemplo, la representación lagrangiana de la velocidad de un punto material

$$\frac{\partial \underline{p}}{\partial t}(\underline{X}, t) = \underline{V}(\underline{X}, t) \equiv \underline{v}(\underline{p}(\underline{X}, t), t)$$

donde $\underline{v}(\underline{x}, t)$ es la representación euleriana de la velocidad, entonces

$$\underline{v}(\underline{x}, t) \equiv \underline{V}(\underline{p}^{-1}(\underline{x}, t), t)$$

Es decir, que la velocidad en el punto \underline{x} del espacio físico, es igual a la velocidad del punto material que pasa por dicho punto en el instante t .

B.2.3. La derivada material

Si obtenemos la derivada parcial con respecto al tiempo de la representación lagrangiana, $\phi(\underline{X}, t)$, de una propiedad intensiva, de acuerdo a la definición de la derivada parcial de una función, es la tasa de cambio con respecto al tiempo que ocurre en un punto material fijo. Es decir, si nos montamos en un punto material y medimos a la propiedad intensiva y luego los valores así obtenidos los derivamos con respecto al tiempo, el resultado final es

$$\frac{\partial \phi}{\partial t}(\underline{X}, t) = \frac{\partial \psi}{\partial t}(\underline{p}(\underline{X}, t), t) + \sum_{i=1}^3 \frac{\partial \psi}{\partial x_i}(\underline{p}(\underline{X}, t), t) \frac{\partial p_i}{\partial t}(\underline{X}, t) = \frac{\partial \psi}{\partial t} + \underline{v} \cdot \nabla \psi$$

Tiene interés evaluar la tasa de cambio con respecto al tiempo que ocurre en un punto material fijo. La derivada material se puede denotar por el siguiente operador:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \underline{v} \cdot \nabla \quad (\text{B.1})$$

B.3. Propiedades extensivas

Son propiedades muy generales de cualquier sustancia que dependen de la cantidad de la materia, por ejemplo, el peso, la masa, el volumen, la longitud, etc. Se denotan por $E(t)$ o bien $E(B, t)$ y funciones que a cada cuerpo B de un sistema continuo y a cada tiempo t le asocia un número real o un vector en \mathbb{R}^3 y puede expresarse como sigue:

$$E(t) = \int_{B(t)} \psi(\underline{x}, t) d\underline{x} \quad (\text{B.2})$$

donde $\psi(\underline{x}, t)$ es la representación euleriana de una propiedad intensiva.

Existe una relación biunívoca entre las propiedades extensivas e intensivas, por que dada la representación euleriana $\psi(\underline{x}, t)$ de cualquier propiedad intensiva, su integral sobre el dominio ocupado por cualquier cuerpo define una propiedad extensiva e inversamente, dada una propiedad extensiva cuando el integrando se expresa en la forma de la ecuación de arriba define una propiedad intensiva, por ejemplo, la masa y la densidad de masa.

B.4. Ecuación de balance global

Los modelos matemáticos de los sistemas continuos están constituidos por balances de propiedades extensivas. En la mecánica de medios continuos se realiza un balance de las propiedades extensivas en las que se basa el modelo, en cada cuerpo del sistema continuo. Para realizar estos balances, debemos identificar las causas por las que las propiedades extensivas de cualquier cuerpo pueden cambiar, las cuales pueden ser:

- Por lo que se genera o se destruye en el interior del cuerpo.
- Por el flujo que se importa o exporta a través de la frontera.

En la teoría de sistemas continuos, la hipótesis básica desde el punto de vista físico para la formulación de las ecuaciones de balance de las propiedades extensivas es que *cualquier variación de la propiedad extensiva proviene de lo que se genera o se destruye dentro del cuerpo o de lo que entra o sale a través de su frontera*, pero también debemos tener en cuenta que puede haber discontinuidades (ver figura B.2).

Esto conduce a la siguiente ecuación de balance global:

$$\frac{d}{dt} E(t) = \int_{B(t)} g(\underline{x}, t) d\underline{x} + \int_{\partial B(t)} q(\underline{x}, t) d\underline{x} + \int_{\Sigma(t)} g_{\Sigma}(\underline{x}, t) d\underline{x} \quad (\text{B.3})$$

- $g(\underline{x}, t)$ es el término fuente en el interior del cuerpo $B(t)$, por unidad de volumen por unidad de tiempo.
- $q(\underline{x}, t)$ es lo que se importa o exporta a través de la frontera del cuerpo $\partial B(t)$, es decir, es el flujo de la propiedad extensiva a través de la frontera del cuerpo, por unidad de área, por unidad de tiempo.

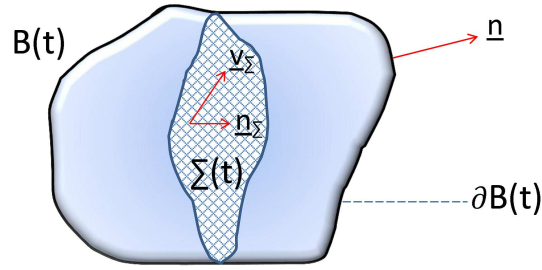


Figura B.2: Esquema de un cuerpo material $B(t)$ con frontera $\partial B(t)$, donde \underline{n} es su vector normal externo. La superficie de discontinuidad $\Sigma(t)$ tiene vector normal \underline{n}_Σ y se mueve con velocidad \underline{v}_Σ . $B(t)$ se mueve a velocidad \underline{v} .

- $g_\Sigma(\underline{x}, t)$ es término fuente por unidad de área en $\Sigma(t)$.

Como para cada tiempo t , existe un campo vectorial $\underline{\tau}(\underline{x}, t)$ tal que

$$q(\underline{x}, t) \equiv \underline{\tau}(\underline{x}, t) \cdot \underline{n}(\underline{x}, t)$$

donde \underline{n} es la normal exterior a la frontera $\partial B(t)$ (ver figura B.2), entonces

$$\frac{d}{dt}E(t) = \int_{B(t)} g(\underline{x}, t) d\underline{x} + \int_{\partial B(t)} \underline{\tau}(\underline{x}, t) \cdot \underline{n}(\underline{x}, t) d\underline{x} + \int_{\Sigma(t)} g_\Sigma(\underline{x}, t) d\underline{x} \quad (\text{B.4})$$

- $g(\underline{x}, t)$ es el término fuente en $B(t)$.
- $\underline{\tau}(\underline{x}, t)$ es el campo de flujo.
- $\underline{n}(\underline{x}, t)$ es la normal exterior a la frontera $\partial B(t)$.
- $g_\Sigma(\underline{x}, t)$ es término fuente en $\Sigma(t)$

B.5. Condiciones de balance local

Las ecuaciones de balance correspondientes a una colección de propiedades extensivas constituyen a los modelos de los sistemas continuos, de esta manera, a cada sistema continuo le corresponde una familia de propiedades extensivas, tal que, el modelo matemático del sistema está constituido por las condiciones de balance de cada una de las propiedades extensivas de dicha familia. Sin embargo, las propiedades extensivas mismas no se utilizan directamente en la formulación del modelo, en su lugar se usan las propiedades intensivas asociadas a cada una de ellas. Esto es posible porque las *ecuaciones de balance global* son equivalentes a las llamadas *condiciones de balance local*, las cuales se expresan en términos de las propiedades intensivas correspondientes. Las condiciones de balance local son de dos clases: las ecuaciones diferenciales de balance local y las condiciones de salto.

B.5.1. Ecuación diferencial de balance local

Son ecuaciones diferenciales parciales que se deben satisfacer en cada punto del espacio ocupado por el sistema continuo, y son de la forma:

$$\frac{\partial \psi}{\partial t} + \nabla \cdot (\psi \underline{v}) = g + \nabla \cdot \underline{\tau} \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.5})$$

desarrollando la divergencia en el lado izquierdo

$$\frac{\partial \psi}{\partial t} + \underline{v} \cdot \nabla \psi + \psi \nabla \cdot \underline{v} = g + \nabla \cdot \underline{\tau} \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.6})$$

por definición de derivada material de ψ , obtenemos

$$\frac{D\psi}{Dt} + \psi \nabla \cdot \underline{v} = g + \nabla \cdot \underline{\tau} \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.7})$$

Por lo tanto, la ecuación diferencial de balance local puede expresarse en términos de la derivada material, que usaremos para expresar la ecuación diferencial de balance local de las ecuaciones básicas de la mecánica de medios continuos.

B.5.2. Condición de salto de balance local

Las condiciones de salto son ecuaciones algebraicas que las discontinuidades deben satisfacer donde ocurren, es decir, en cada punto de la superficie de discontinuidad $\Sigma(t)$. Las propiedades intensivas pueden tener discontinuidades de salto exclusivamente a través de la superficie $\Sigma(t)$, donde los límites por ambos lados de $\Sigma(t)$ existen, pero son diferentes.

$$[\![\psi(\underline{v} - \underline{v}_\Sigma) - \underline{\tau}]\!] \cdot \underline{n}_\Sigma = g_\Sigma \quad \forall \underline{x} \in \Sigma(t). \quad (\text{B.8})$$

donde $[\![f]\!] = \lim_{\underline{x} \rightarrow \Sigma^+} f(\underline{x}) - \lim_{\underline{x} \rightarrow \Sigma^-} f(\underline{x})$ es el salto de la función f .

Las ecuaciones diferenciales de balance local son de uso mucho más amplio que las condiciones de salto, pues estas últimas únicamente se aplican en problemas de carácter especial donde las propiedades intensivas son discontinuas, mientras que las primeras en todo punto del espacio ocupado por el sistema continuo.

B.6. Teoremas

En esta sección, enunciamos los teoremas que se utilizarán en la derivación de las condiciones de balance local, a partir de la ecuación de balance global.

B.6.1. Teorema de Gauss extendido

Sea $B(t)$ una región conexa con frontera $\partial B(t)$ donde \underline{n} es un vector normal unitario y $\underline{\tau}(\underline{x}, t)$ una función vectorial continuamente diferenciable en $B(t)$ excepto en $\Sigma(t)$ con vector normal unitario \underline{n}_Σ , entonces

$$\int_{\partial B(t)} \underline{\tau} \cdot \underline{n} d\underline{x} = \int_{B(t)} \nabla \cdot \underline{\tau} d\underline{x} + \int_{\Sigma(t)} \llbracket \underline{\tau} \rrbracket \cdot \underline{n}_\Sigma d\underline{x} \quad (\text{B.9})$$

B.6.2. Teorema de transporte de Reynolds extendido

Sea $B(t)$ una región conexa (el dominio de un cuerpo) con velocidad \underline{v} y $\Sigma(t)$ una superficie que interseca a $B(t)$, con vector normal unitario \underline{n}_Σ y velocidad \underline{v}_Σ . Si $\psi(\underline{x}, t)$ es una función de valores reales definida en $B(t)$ continuamente diferenciable excepto en la superficie $\Sigma(t)$, entonces

$$\frac{d}{dt} \int_{B(t)} \psi(\underline{x}, t) d\underline{x} = \int_{B(t)} \left(\frac{\partial \psi}{\partial t} + \nabla \cdot (\psi \underline{v}) \right) d\underline{x} + \int_{\Sigma(t)} \llbracket \psi(\underline{v} - \underline{v}_\Sigma) \rrbracket \cdot \underline{n}_\Sigma d\underline{x} \quad (\text{B.10})$$

B.6.3. Lema de Dubois-Reymond extendido

Sea $f(\underline{x})$ continua excepto en $\Sigma(t)$ y $g(\underline{x})$ continua en $\Sigma(t)$. Si

$$\int_{B(t)} f(\underline{x}) d\underline{x} + \int_{\Sigma(t)} g(\underline{x}) d\underline{x} = 0 \quad (\text{B.11})$$

entonces $f(\underline{x}) = 0, \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad y \quad g(\underline{x}) = 0, \quad \forall \underline{x} \in \Sigma(t).$

B.7. De la ecuación de balance global a las ecuaciones de balance local

Afirmamos que en un sistema continuo, la *ecuación de balance global*, se satisface para todo cuerpo del sistema continuo, si y solamente si, se cumplen las *condiciones de balance local*. En efecto, sustituyendo en la ecuación de balance global (B.4) la definición de propiedad extensiva (B.2)

$$\frac{d}{dt} \int_{B(t)} \psi \, d\underline{x} = \int_{B(t)} g \, d\underline{x} + \int_{\partial B(t)} \underline{\tau} \cdot \underline{n} d\underline{x} + \int_{\Sigma(t)} g_{\Sigma} d\underline{x} \quad (\text{B.12})$$

aplicando al segundo sumando (B.9) el teorema de Gauss extendido

$$\begin{aligned} \frac{d}{dt} \int_{B(t)} \psi \, d\underline{x} &= \int_{B(t)} g \, d\underline{x} + \int_{B(t)} \nabla \cdot \underline{\tau} d\underline{x} + \int_{\Sigma(t)} [\![\underline{\tau}]\!] \cdot \underline{n}_{\Sigma} d\underline{x} + \int_{\Sigma(t)} g_{\Sigma} \, d\underline{x} \\ &= \int_{B(t)} (g + \nabla \cdot \underline{\tau}) \, d\underline{x} + \int_{\Sigma(t)} ([\![\underline{\tau}]\!] \cdot \underline{n}_{\Sigma} + g_{\Sigma}) \, d\underline{x} \end{aligned} \quad (\text{B.13})$$

aplicando en la izquierda (B.10) el teorema de transporte de Reynolds extendido

$$\begin{aligned} &\int_{B(t)} \left(\frac{\partial \psi}{\partial t} + \nabla \cdot (\psi \underline{v}) \right) d\underline{x} + \int_{\Sigma(t)} [\![\psi(\underline{v} - \underline{v}_{\Sigma})]\!] \cdot \underline{n}_{\Sigma} d\underline{x} \\ &= \int_{B(t)} (g(\underline{x}, t) + \nabla \cdot \underline{\tau}) d\underline{x} + \int_{\Sigma(t)} ([\![\underline{\tau}]\!] \cdot \underline{n}_{\Sigma} + g_{\Sigma}(\underline{x}, t)) d\underline{x} \end{aligned} \quad (\text{B.14})$$

agrupando términos del lado izquierdo de la igualdad

$$\begin{aligned} &\int_{B(t)} \left(\frac{\partial \psi}{\partial t} + \nabla \cdot (\psi \underline{v}) - (g(\underline{x}, t) + \nabla \cdot \underline{\tau}) \right) d\underline{x} \\ &+ \int_{\Sigma(t)} ([\![\psi(\underline{v} - \underline{v}_{\Sigma}) - \underline{\tau}]\!] \cdot \underline{n}_{\Sigma} - g_{\Sigma}(\underline{x}, t)) d\underline{x} = 0 \end{aligned} \quad (\text{B.15})$$

aplicando (B.11) el lema de Dubois-Reymond extendido, obtenemos la ecuación de balance local y la condición de salto de balance local respectivamente:

$$\frac{\partial \psi}{\partial t} + \nabla \cdot (\psi \underline{v}) = g(\underline{x}, t) + \nabla \cdot \underline{\tau}, \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.16})$$

$$[\![\psi(\underline{v} - \underline{v}_{\Sigma}) - \underline{\tau}]\!] \cdot \underline{n}_{\Sigma} = g_{\Sigma}(\underline{x}, t), \quad \forall \underline{x} \in \Sigma(t) \quad (\text{B.17})$$

B.8. Ejemplo

Consideremos el cuerpo de un fluido incompresible, el cual llena completamente el volumen en que se encuentra. Su movimiento ocurre en un espacio libre tomando en cuenta la siguiente hipótesis:

Los cuerpos conservan su volumen en el movimiento de sustancias incompresibles (sólidas o fluidas).

Sea $V_f(t)$ el volumen del fluido contenido en el cuerpo y $B(t)$ el dominio del espacio físico ocupado por el cuerpo, de esta manera, el volumen del fluido es igual al volumen de la región que ocupa y esto se expresa de la siguiente forma:

$$V_f(t) = \int_{B(t)} d\underline{x} \quad (\text{B.18})$$

El volumen del fluido es una propiedad extensiva $\implies E(t) \equiv V_f(t)$
y su propiedad intensiva correspondiente esta dada por $\psi(\underline{x}, t) \equiv 1$

Por hipótesis, el volumen de los cuerpos se conserva durante su movimiento:

$$\frac{d}{dt} V_f(t) = 0 \quad (\text{B.19})$$

Es decir, la **ecuación de balance global** esta dada por

$$\frac{d}{dt} V_f(t) = \int_{B(t)} 0 d\underline{x} + \int_{\partial B(t)} 0 d\underline{x} + \int_{\Sigma(t)} 0 d\underline{x} = 0 \quad (\text{B.20})$$

ecuación diferencial de balance local:

$$\nabla \cdot \underline{v} = 0 \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.21})$$

condición de salto de balance local:

$$[[\underline{v}]] \cdot \underline{n}_\Sigma = 0 \quad \forall \underline{x} \in \Sigma(t) \quad (\text{B.22})$$

B.9. Ecuaciones básicas de la MMC

B.9.1. Ecuaciones de balance de masa

$$M(t) = \int_{B(t)} \rho(\underline{x}, t) d\underline{x} \quad (\text{B.23})$$

Propiedad extensiva: masa $M(t)$

Propiedad intensiva: densidad $\psi \equiv \rho$

Si tenemos que: $g \equiv 0$, $g_\Sigma \equiv 0$ y $\underline{\tau} \equiv 0$, existe conservación de masa.

Ecuación de balance global

$$\frac{d}{dt} M(t) = 0 \quad (\text{B.24})$$

Ecuación de balance local:

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \underline{v} = 0 \quad ; \quad \forall \underline{x} \in B(t) \setminus \Sigma(t). \quad (\text{B.25})$$

también es conocida como *ecuación de continuidad en mecánica de fluidos*.

Condiciones de salto:

$$[[\rho(\underline{v} - \underline{v}_\Sigma)]] \cdot \underline{n}_\Sigma = 0 \quad ; \quad \forall \underline{x} \in \Sigma(t). \quad (\text{B.26})$$

B.9.2. Ecuaciones de balance de momento lineal

$$\underline{\mathfrak{M}}(t) = \int_{B(t)} \rho(\underline{x}, t) \underline{v}(\underline{x}, t) d\underline{x} \quad (\text{B.27})$$

Propiedad extensiva: momento lineal $\underline{\mathfrak{M}}(t)$.

Propiedad intensiva: densidad de momento lineal $\psi \equiv \rho \underline{v}$.

Si tenemos que $g \equiv \rho \underline{b}$, $g_\Sigma \equiv 0$ y $\underline{\tau} \equiv \underline{\underline{\sigma}}$

donde

- $\rho \underline{b}$ — fuerzas de cuerpo por unidad de volumen
- $\underline{\underline{\sigma}}$ — tensor de los esfuerzos
- $\underline{\underline{\sigma}} \cdot \underline{n} = \underline{T}$ — fuerzas de tracción.

Ecuación de balance global:

$$\frac{d}{dt}\mathfrak{M}(t) = \int_{B(t)} \rho \underline{b} d\underline{x} + \int_{\partial B(t)} \underline{\underline{\sigma}} \cdot \underline{n} d\underline{x} \quad (\text{B.28})$$

Ecuación de balance local:

$$\frac{D\rho \underline{v}}{Dt} + \rho \underline{v} \nabla \cdot \underline{v} = \nabla \cdot \underline{\underline{\sigma}} + \rho \underline{b} \quad ; \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.29})$$

Esta ecuación se puede interpretar como un sistema de ecuaciones si la expresamos como

$$\frac{D\rho v_i}{Dt} + \rho v_i \nabla \cdot \underline{v} = \nabla \cdot \sigma_i + \rho b_i \quad ; \quad i = 1, 2, 3; \quad (\text{B.30})$$

donde $\underline{\underline{\sigma}} = (\sigma_1, \sigma_2, \sigma_3)$; $\underline{b} = (b_1, b_2, b_3)$.

Desarrollando la derivada material en el término izquierdo

$$\left(\rho \frac{D\underline{v}}{Dt} + \underline{v} \frac{D\rho}{Dt} \right) + \rho \underline{v} \nabla \cdot \underline{v} = \rho \frac{D\underline{v}}{Dt} + \underline{v} \left(\frac{D\rho}{Dt} + \rho \nabla \cdot \underline{v} \right) = \rho \frac{D\underline{v}}{Dt} \quad (\text{B.31})$$

ya que $\frac{D\rho}{Dt} + \rho \nabla \cdot \underline{v} = 0$ por conservación de la masa. Entonces obtenemos la ecuación

$$\rho \frac{D\underline{v}}{Dt} = \nabla \cdot \underline{\underline{\sigma}} + \rho \underline{b} \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.32})$$

que constituye la 1ª ley de cauchy de la mecánica de medios continuos.

Condiciones de salto:

$$\llbracket \rho \underline{v}(\underline{v} - \underline{v}_\Sigma) - \underline{\underline{\sigma}} \rrbracket \cdot \underline{n}_\Sigma = 0; \quad \forall \underline{x} \in \Sigma(t). \quad (\text{B.33})$$

Ecuación general de flujo de Navier-Stokes

Sean

- $\rho \underline{b}$ - fuerzas de cuerpo por unidad de volumen.
- $\underline{\underline{D}}$ - la parte simétrica del tensor de las velocidades $\nabla \underline{v}$, donde $\underline{\underline{D}} = \frac{1}{2}(\nabla \underline{v} + (\nabla \underline{v})^T)$, es decir, $D_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$
- $\underline{\underline{\sigma}} = -p \underline{\underline{I}} + \lambda(\nabla \cdot \underline{v}) \underline{\underline{I}} + 2\mu \underline{\underline{D}}$ - es el tensor de los esfuerzos.

Usando la notación de Einstein para los índices repetidos

$$\nabla \cdot (-p \underline{\underline{I}}) = -\frac{\partial p}{\partial x_i} = -\nabla p$$

Análogamente

$$\nabla \cdot (\lambda(\nabla \cdot \underline{v}) \underline{\underline{I}}) = \lambda \nabla(\nabla \cdot \underline{v})$$

$$\begin{aligned}
\nabla \cdot (2\mu \underline{\underline{D}}) &= \mu \nabla \cdot (2\underline{\underline{D}}) = \mu \frac{\partial}{\partial x_i} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) = \mu \frac{\partial}{\partial x_i} \frac{\partial v_i}{\partial x_j} + \mu \frac{\partial}{\partial x_i} \frac{\partial v_j}{\partial x_i} \\
&= \mu \frac{\partial}{\partial x_j} \frac{\partial v_i}{\partial x_i} + \mu \frac{\partial}{\partial x_i} \frac{\partial v_j}{\partial x_i} = \mu \nabla (\nabla \cdot \underline{v}) + \mu \nabla^2 \underline{v}
\end{aligned}$$

Como la divergencia es un operador lineal

$$\begin{aligned}
\nabla \cdot \underline{\underline{\sigma}} &= \nabla \cdot (-p \underline{\underline{I}}) + \nabla \cdot (\lambda (\nabla \cdot \underline{v}) \underline{\underline{I}}) + \nabla \cdot (2\mu \underline{\underline{D}}) \\
&= -\nabla p + \mu \nabla^2 \underline{v} + (\lambda + \mu) \nabla (\nabla \cdot \underline{v})
\end{aligned}$$

al sustituir en la ecuación de balance de momento lineal, obtenemos

$$\rho \frac{D}{Dt} \underline{v} = \rho \underline{b} - \nabla p + \mu \nabla^2 \underline{v} + (\lambda + \mu) \nabla (\nabla \cdot \underline{v}) \quad (\text{B.34})$$

que es la ecuación general de flujo de Navier-Stokes.

Desarrollando en la izquierda la derivada material y dividiendo entre ρ de ambos lados, es la ecuación que utilizaremos para deducir la ley de Darcy.

$$\frac{\partial}{\partial t} \underline{v} + \underline{v} \cdot \nabla \underline{v} = \underline{b} - \frac{1}{\rho} \nabla p + \frac{\mu}{\rho} \nabla^2 \underline{v} + \frac{1}{\rho} (\lambda + \mu) \nabla (\nabla \cdot \underline{v}) \quad (\text{B.35})$$

B.9.3. Ecuaciones de balance de energía

$$\mathcal{E}(t) = \int_{B(t)} \rho \left(E + \frac{1}{2} |\underline{v}|^2 \right) d\underline{x} \quad (\text{B.36})$$

Propiedad extensiva: Energía $\mathcal{E}(t)$

Propiedad intensiva: $\psi \equiv \rho(E + \frac{1}{2} |\underline{v}|^2)$

donde E es la energía interna por unidad de masa.

Si tenemos que: $g \equiv \rho(h + \underline{b} \cdot \underline{v})$, $g_\Sigma \equiv 0$ y $\underline{\tau} \equiv \underline{q} + \underline{\underline{\sigma}} \cdot \underline{v}$ donde

- ρh - fuente de calor por unidad de volumen
- $\rho \underline{b} \cdot \underline{v}$ - trabajo por unidad de tiempo por unidad de volumen realizado por las fuerzas del cuerpo
- $\underline{q} \cdot \underline{n}$ - flujo de calor a través de la frontera por unidad de tiempo por unidad de área
- $(\underline{\underline{\sigma}} \cdot \underline{v}) \cdot \underline{n} = (\underline{\underline{\sigma}} \cdot \underline{v}) \cdot \underline{v} = \underline{\underline{T}} \cdot \underline{v}$ - trabajo por unidad de tiempo por unidad de área realizado por las fuerzas de tracción.

Ecuación de balance global:

$$\frac{d}{dt}\mathcal{E}(t) = \int_{B(t)} \rho(h + \underline{b} \cdot \underline{v}) d\underline{x} + \int_{\partial B(t)} (\underline{q} + \underline{\sigma} \cdot \underline{v}) \cdot \underline{n} d\underline{x} \quad (\text{B.37})$$

Ecuación de balance local:

$$\frac{D}{Dt} \left(\rho \left(E + \frac{1}{2} \underline{v} \cdot \underline{v} \right) \right) + \rho \left(E + \frac{1}{2} \underline{v} \cdot \underline{v} \right) \nabla \cdot \underline{v} = \nabla \cdot (\underline{q} + \underline{\sigma} \cdot \underline{v}) + \rho(h + \underline{b} \cdot \underline{v}) \quad (\text{B.38})$$

Desarrollando el término de la derivada material resulta:

$$\frac{D}{Dt} \left(\rho \left(E + \frac{1}{2} \underline{v} \cdot \underline{v} \right) \right) = \rho \frac{DE}{Dt} + \rho \frac{D(\frac{1}{2} \underline{v} \cdot \underline{v})}{Dt} + \frac{D\rho}{Dt} \left(E + \frac{1}{2} \underline{v} \cdot \underline{v} \right) \quad (\text{B.39})$$

Sustituyendo (2.30) en el miembro izquierdo de la ecuación (2.28) obtenemos:

$$\rho \frac{DE}{Dt} + \rho \frac{D(\frac{1}{2} \underline{v} \cdot \underline{v})}{Dt} + \left(E + \frac{1}{2} \underline{v} \cdot \underline{v} \right) \left(\frac{D\rho}{Dt} + \rho \nabla \cdot \underline{v} \right) = \nabla \cdot (\underline{q} + \underline{\sigma} \cdot \underline{v}) + \rho(h + \underline{b} \cdot \underline{v}) \quad (\text{B.40})$$

ya que $\frac{D\rho}{Dt} + \rho \nabla \cdot \underline{v} = 0$ por conservación de masa, resulta:

$$\rho \frac{DE}{Dt} + \rho \frac{D(\frac{1}{2} \underline{v} \cdot \underline{v})}{Dt} = \nabla \cdot (\underline{q} + \underline{\sigma} \cdot \underline{v}) + \rho(h + \underline{b} \cdot \underline{v}). \quad (\text{B.41})$$

Si tenemos en cuenta que:

$$\nabla \cdot (\underline{\sigma} \cdot \underline{v}) = \underline{v} \cdot (\nabla \cdot \underline{\sigma}) + \underline{\sigma} \cdot \nabla \underline{v} \quad y \quad \frac{D(\frac{1}{2} \underline{v} \cdot \underline{v})}{Dt} = \underline{v} \cdot \frac{D\underline{v}}{Dt}$$

entonces

$$\rho \frac{DE}{Dt} + \underline{v} \cdot \left(\rho \frac{D\underline{v}}{Dt} - \nabla \cdot \underline{\sigma} - \rho \underline{b} \right) = \nabla \cdot \underline{q} + \rho h + \underline{\sigma} \cdot \nabla \underline{v}. \quad (\text{B.42})$$

Si tenemos en cuenta que $\rho \frac{D\underline{v}}{Dt} - \nabla \cdot \underline{\sigma} - \rho \underline{b} = 0$ por la ecuación de balance del momento lineal, obtenemos la ecuación de balance de energía:

$$\rho \frac{DE}{Dt} = \nabla \cdot \underline{q} + \rho h + \underline{\sigma} \cdot \nabla \underline{v}, \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.43})$$

donde $\underline{\sigma} \cdot \nabla \underline{v} = \sum_{i=1}^3 \sum_{j=1}^3 \sigma_{ij} \frac{\partial v_i}{\partial x_j}$

Condiciones de salto:

$$\left[\left[\rho \left(E + \frac{1}{2} \underline{v} \cdot \underline{v} \right) (\underline{v} - \underline{v}_\Sigma) - (\underline{q} + \underline{\sigma} \cdot \underline{v}) \right] \cdot \underline{n}_\Sigma = 0, \quad \forall \underline{x} \in \Sigma(t) \right] \quad (\text{B.44})$$

B.10. Modelos de fases múltiples

Como introducción a los modelos de sistemas continuos de fases múltiples, comenzaremos con un ejemplo muy sencillo, para después dar un salto a las ecuaciones de los modelos de fases múltiples y varias componentes. Imaginemos un vaso de vidrio transparente con agua de sabor y algunos hielos. Si miramos el contenido dentro del vaso, observaremos que hay dos fases:

* **Fase 1:** La fase líquida (agua de sabor) * **Fase 2:** La fase sólida (hielos)

Desde este enfoque macroscópico suponemos que en cada punto de la región ocupada por el contenido del vaso en el espacio físico (\mathbb{R}^3), existe sólo un punto material de cada una de las fases y cada una se mueve con su propia velocidad. Ahora, si agitamos circularmente el vaso, observaremos que ambas fases se mueven con distintas velocidades.

El agua de sabor (la fase líquida) tiene dos componentes:

* **Componente 1:** agua pura líquida * **Componente 2:** saborizante

La fase sólida sólo tiene una componente: **agua pura sólida** En la teoría de los sistemas continuos, los modelos para procesos como el que se acaba de describir, se conceptualizan como se explica a continuación. Se considera que hay dos fases: la fase líquida y la fase sólida. En cada punto del espacio físico (\mathbb{R}^3) hay un punto material de cada una de las fases y cada una de ellas se mueve con su propia velocidad. Por lo mismo, cada dominio de \mathbb{R}^3 define dos cuerpos, que son los formados por los conjuntos de puntos materiales correspondientes a cada una de las fases. Además, en cada punto tenemos definidas dos velocidades, que en general pueden ser diferentes, por lo que los dos cuerpos que en un tiempo ocupan el mismo dominio de \mathbb{R}^3 , en general en otros tiempos ocupan dominios que no coinciden. En el caso que nos ocupa es posible hacer el balance de masa o de otras propiedades extensivas, pero en ellas debe utilizarse la velocidad asociada con la fase que corresponda.

Las **ecuaciones de balance global** del ejemplo son las siguientes:

$$\begin{aligned}\frac{d}{dt}E_1^1(t) &= \int_{B(t)} g_1^1(\underline{x}, t) d\underline{x} + \int_{\partial B(t)} \underline{\tau}_1^1(\underline{x}, t) \cdot \underline{n}(\underline{x}, t) d\underline{x} + \int_{\Sigma(t)} g_{\Sigma_1^1}^1(\underline{x}, t) d\underline{x} \\ \frac{d}{dt}E_2^1(t) &= \int_{B(t)} g_2^1(\underline{x}, t) d\underline{x} + \int_{\partial B(t)} \underline{\tau}_2^1(\underline{x}, t) \cdot \underline{n}(\underline{x}, t) d\underline{x} + \int_{\Sigma(t)} g_{\Sigma_2^1}^1(\underline{x}, t) d\underline{x} \\ \frac{d}{dt}E_1^2(t) &= \int_{B(t)} g_1^2(\underline{x}, t) d\underline{x} + \int_{\partial B(t)} \underline{\tau}_1^2(\underline{x}, t) \cdot \underline{n}(\underline{x}, t) d\underline{x} + \int_{\Sigma(t)} g_{\Sigma_1^2}^2(\underline{x}, t) d\underline{x}\end{aligned}$$

Ecuaciones diferenciales de balance local

$$\frac{\partial}{\partial t}\psi_1^1 + \nabla \cdot (\psi_1^1 \underline{v}^1) = g_1^1 + \nabla \cdot \underline{\tau}_1^1, \quad \forall \underline{x} \in B(t) \setminus \Sigma(t)$$

$$\begin{aligned}\frac{\partial}{\partial t}\psi_2^1 + \nabla \cdot (\psi_2^1 \underline{v}^1) &= g_2^1 + \nabla \cdot \underline{\mathcal{I}}_2^1, & \forall \underline{x} \in B(t) \setminus \Sigma(t) \\ \frac{\partial}{\partial t}\psi_1^2 + \nabla \cdot (\psi_1^2 \underline{v}^2) &= g_1^2 + \nabla \cdot \underline{\mathcal{I}}_1^2, & \forall \underline{x} \in B(t) \setminus \Sigma(t)\end{aligned}$$

Condiciones de salto de balance local

$$\begin{aligned} \llbracket \psi_1^1(\underline{v}^1 - \underline{v}_\Sigma^1) - \underline{\tau}_1^1 \rrbracket \cdot \underline{n}_\Sigma &= g_{\Sigma 1}^1, & \forall \underline{x} \in \Sigma(t) \\ \llbracket \psi_2^1(\underline{v}^1 - \underline{v}_\Sigma^1) - \underline{\tau}_2^1 \rrbracket \cdot \underline{n}_\Sigma &= g_{\Sigma 2}^1, & \forall \underline{x} \in \Sigma(t) \\ \llbracket \psi_1^2(\underline{v}^2 - \underline{v}_\Sigma^2) - \underline{\tau}_1^2 \rrbracket \cdot \underline{n}_\Sigma &= g_{\Sigma 1}^2, & \forall \underline{x} \in \Sigma(t) \end{aligned}$$

A continuación se explica como se construyen los modelos de los procesos, pues más que los sistemas mismos, se modelan sus procesos. Primero, se identifica a un conjunto finito, o familia, de propiedades extensivas, luego a cada una de las propiedades extensivas de esa familia se les impone la condición de satisfacer la ecuación de balance global en cada cuerpo del sistema continuo, lo cual es equivalente a la ecuación diferencial de balance local y a la condición de salto. Esta forma de proceder se aplica no solamente a sistemas continuos de una sola fase, sino también a sistemas de varias fases.

Las fases contienen varios componentes superpuestos que se mueven a la misma velocidad. Si consideramos un sistema con N fases donde cada fase tiene M_α componentes y a cada componente le corresponde una propiedad intensiva ψ_γ^α asociada a la propiedad extensiva E_γ^α , donde α denota el número de la fase ($\alpha = 1, \dots, N$) y γ denota el número de la componente dentro de dicha fase ($\gamma = 1, \dots, M_\alpha$), entonces las ecuaciones de balance global, al igual que las ecuaciones de balance local, se plantean por componentes.

La *propiedad intensiva* asociada a la propiedad extensiva E_γ^α es $\psi_\gamma^\alpha(\underline{x}, t)$:

$$E_\gamma^\alpha(t) = \int_{B(t)} \psi_\gamma^\alpha(\underline{x}, t) d\underline{x} \quad (\text{B.45})$$

Ecuaciones de balance global

$$\frac{d}{dt} E_\gamma^\alpha(t) = \int_{B(t)} g_\gamma^\alpha(\underline{x}, t) d\underline{x} + \int_{\partial B(t)} \underline{\tau}_\gamma^\alpha(\underline{x}, t) \cdot \underline{n}(\underline{x}, t) d\underline{x} + \int_{\Sigma(t)} g_{\Sigma \gamma}^\alpha(\underline{x}, t) d\underline{x} \quad (\text{B.46})$$

Ecuaciones diferenciales de balance local

$$\frac{\partial}{\partial t} \psi_\gamma^\alpha + \nabla \cdot (\psi_\gamma^\alpha \underline{v}^\alpha) = g_\gamma^\alpha + \nabla \cdot \underline{\tau}_\gamma^\alpha, \quad \forall \underline{x} \in B(t) \setminus \Sigma(t) \quad (\text{B.47})$$

Condiciones de salto de balance local

$$\llbracket [\psi_\gamma^\alpha(\underline{v}^\alpha - \underline{v}_\Sigma^\alpha) - \underline{\tau}_\gamma^\alpha] \rrbracket \cdot \underline{n}_\Sigma = g_{\Sigma \gamma}^\alpha, \quad \forall \underline{x} \in \Sigma(t) \quad (\text{B.48})$$

Estas son las ecuaciones básicas que gobiernan a una gran diversidad de sistemas continuos. Sin embargo, ellas no constituyen modelos completos, los cuales explicaremos con más detalle a continuación.

B.11. Modelos completos

En general las ecuaciones diferenciales tienen muchas soluciones, por lo que es necesario complementarlas con condiciones iniciales y de frontera. El modelo de un sistema continuo es completo si define un problema bien planteado. Un problema de valores iniciales y de frontera es bien planteado si se cumple que existe una única solución y ésta depende de las condiciones iniciales y de frontera de manera continua.

B.11.1. Condiciones iniciales

Cuando en la ecuación diferencial interviene el tiempo, se incluyen condiciones iniciales que expresan el valor de la función al tiempo inicial $t = 0$.

$$c(\underline{x}, 0) = c_0(\underline{x}) \quad (\text{B.49})$$

B.11.2. Condiciones de frontera

Se imponen en la frontera exterior del dominio

(a) Dirichlet

Especifica los valores que toma la función $c(\underline{x}, t)$ en la frontera $\partial B(t)$

$$c(\underline{x}, t) = f(\underline{x}) \quad (\text{B.50})$$

(b) Neumann

Se prescribe la derivada normal en la frontera, por lo tanto aquí se conoce el valor de la derivada de la función $c(\underline{x}, t)$ con respecto a la normal \underline{n} a lo largo de la frontera $\partial B(t)$

$$\nabla c(\underline{x}, t) \cdot \underline{n} = g(\underline{x}) \quad (\text{B.51})$$

(c) Robin

Esta condición es una combinación lineal de las dos anteriores.

$$a(\underline{x})c(\underline{x}, t) + b(\underline{x})\nabla c(\underline{x}, t) \cdot \underline{n} = \gamma(\underline{x}, t) \quad (\text{B.52})$$

$\gamma(\underline{x}, t)$ es la función prescrita en la frontera exterior.

Para obtener *modelos completos*, además de las condiciones de balance local, ecuación diferencial de balance local y condición de salto de balance local, es necesario evaluar la *generación interna*, determinada por $g(\underline{x}, t)$ y $g_\Sigma(\underline{x}, t)$, y el campo de flujo $\underline{\tau}(\underline{x}, t)$, en términos de las funciones conocidas de las propiedades intensivas asociadas. A través de estas ecuaciones constitutivas, se integra el conocimiento científico y tecnológico en los modelos matemáticos, todo depende de la clase de los procesos involucrados. En conclusión, los modelos de los sistemas continuos están constituidos por:

- Una colección de propiedades intensivas y extensivas.
- Un conjunto de condiciones de balance local correspondientes a cada propiedad intensiva, en cada una de las cuales, la velocidad de los puntos materiales es la de la fase correspondiente.
- Condiciones iniciales y de frontera que deben satisfacer las propiedades intensivas.
- Suficientes relaciones que ligan a las propiedades intensivas entre sí y que definen a g , $\underline{\tau}$ y \underline{v} en términos de éstas, conocidas como leyes constitutivas.

Bibliografía

- [1] ALADASANI, A., B., W., AND BAI, Y. S. Investigating low-salinity waterflooding recovery mechanisms in sandstone reservoirs. *18th SPE Improved Oil Recovery Symposium. Tulsa, Oklahoma 14-18 SPE 152997* (April 2012), 2521–2536. (Citado en la página 115.)
- [2] ALLEN, M. B., HERRERA, I., AND PINDER, G. F. *Numerical modeling in science and engineering*. John Wiley & Sons, 1988. (Citado en la página 35.)
- [3] AUSTAD, T., REZAEIDoust, A., AND PUNTERVOLD., T. Chemical mechanism of low salinity water flooding in sandstone reservoirs. *SPE Improved Oil Recovery Symposium. Tulsa Oklahoma, 24-28 April, SPE 129767* (2010), 2010. (Citado en las páginas 112, 113, 115 y 116.)
- [4] BABUSKA, I. The finite element method with lagrangian multipliers. (Citado en las páginas 17 y 19.)
- [5] BOUSSOUR, S., CISSOKO, M., CORDIER, P., BERTIN, H., AND HAMON., G. *Oil Recovery by Low Salinity Brine Injection: Laboratory Results on Outcrop and Reservoir Cores.*, vol. SPE 124277. SPE Annual Technical Conference and Exhibition. Louisiana, New Orleans, 4-7 octubre 2009. (Citado en la página 112.)
- [6] BRENNER, S. C., AND SCOTT, L. R. The mathematical theory of finite elements methods. (Citado en la página 6.)
- [7] BREZZI, F., DOUGLAS, J., AND MARINI, L. D. Two families of mixed finite elements for second order elliptic problems. (Citado en la página 8.)
- [8] CHEN, Z., AND HUAN. *Finite Element Methods and Their Applications*. Springer, 2005. (Citado en las páginas 14, 17, 35 y 40.)
- [9] CHEN, Z., HUAN, G., AND MA, Y. *Computational methods for multiphase flows in porous media*, vol. 2. Siam, 2006. (Citado en las páginas 6 y 37.)
- [10] CIARLET, P. G. Numerical analysis of the finite element method. *Les Presses de l'Université de Montreal* (1976). (Citado en la página 6.)
- [11] CORONADO-GALLARDO, M., AND DÍAZ-VIERA, M. A. Modeling fines migration and permeability loss caused by low salinity in porous media. *Journal of Petroleum Science and Engineering 150* (February 2017), 355–365. (Citado en las páginas 79, 80 y 89.)

- [12] DANG, C. T. Q., NGHIEM, L., CHEN, Z., Q.P., N., AND NGUYEN, T. State-of-the-art low salinity waterflooding for enhanced oil recovery. *SPE Asian Pacific Oil and Gas Conference and Exhibition, 22-24 October 2013. Jakarta, Indonesia SPE, 165903*. (Citado en las páginas 112 y 115.)
- [13] FJELDE, I., ASEN, S., AND OMEKEH., A. *Low Salinity Water Flooding Experiments and Interpretation by Simulations.*, vol. SPE, 154142. 18th SPE Improved Oil Recovery Symposium. Tulsa, OK 14-18 April 2012. (Citado en la página 116.)
- [14] GONZÁLEZ-BUSTO MÚGICA, B. N. *La dinámica de sistemas como metodología para la elaboración de modelos de simulación*. Universidad de Oviedo, 1998. (Citado en la página 84.)
- [15] HERRERA, I. *Modelación matemática de sistemas terrestres*. Instituto de Geofísica, UNAM, 2010. (Citado en la página 35.)
- [16] HUSSAIN, F., ZEINIJAHRMI, A., BEDRIKOVETSKY, P., BADALYAN, A., CARAGEORGOS, T., AND CINAR, Y. An experimental study of improved oil recovery through fines-assisted waterflooding. 187–197. (Citado en la página 83.)
- [17] LAGER, A., WEBB, K. J., BLACK, C. J., SINGLETON, M., AND SORBIE., K. S. Low salinity oil recovery - an experimental investigation. *International Symposium of the Society of Core Analysts SCA-2006. Trondheim, Norway (2006)*. (Citado en las páginas 112, 114 y 116.)
- [18] LOGG, A., KENT-ANDRE, M., AND WELLS, G. N. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2011. (Citado en las páginas XIV, 17, 19, 21, 22, 42 y 44.)
- [19] MCGUIRE, P., CHATHAM, J. R., PASKVAN, F. K., SOMER, D., AND CARINI., F. H. Low salinity oil recovery: An exciting new eor opportunity for alaskas north slope. *2005 SPE Western Regional Meeting. Irving, CA , 30 March-1 April 2005 SPE 93903 (2005)*. (Citado en las páginas 112, 113 y 115.)
- [20] NASRALLA, R. A., AND NASR-EL-DIN., H. Double-layer expansion: Is it a primary mechanism of improved oil recovery by low-salinity waterflooding. *SPE Reservoir Evaluation & Engineering SPE 154334 (February 2014)*, 49–59. (Citado en la página 115.)
- [21] NÉDÉLEC, J. C. Mixed finite elements in \mathbb{R}^3 . *Numer. Math.* (1980). (Citado en la página 11.)
- [22] PANG, S., AND SHARMA, M. A model for predicting injectivity decline in water-injection wells. *SPE Formation Evaluation (Septiembre 1997)*, 194–202. (Citado en la página 83.)
- [23] RAVIART, P.-A., AND THOMAS, J. M. A mixed finite element method for 2nd order elliptic problems. (Citado en la página 10.)

- [24] RIVET, S. M., LAKE, L. W., AND POPE., G. A. A coreflood investigation of low-salinity enhanced oil recovery. *PE Annual Technical Conference and Exhibition. Florence, Italy, 19-22 September 2010 SPE 134297*. (Citado en la página 116.)
- [25] SCOTT, L. R., AND VOGELIUS, M. Norm estimates for a maximal right inverse of the divergence operator in spaces of piecewise polynomials. (Citado en la página 8.)
- [26] SHENG, J. Critical review of low-salinity waterflooding. *J. Pet. Sc. Eng.* (2014), 216–224. (Citado en las páginas 112, 114 y 116.)
- [27] SKIBA, Y. *Métodos y esquemas numéricos. Un análisis computacional*. Dirección General de Publicaciones y Fomento Editorial UNAM, 2005. (Citado en la página 6.)
- [28] VALDYA, R., AND FOLGER., H. S. Fines migration and formation damage. *SPE Production Engineering SPE 19413* (November 1992), 325–330. (Citado en la página 114.)
- [29] WEBB, K. J., BLACK, C. J. J., AND AL-AJEEL., H. Low salinity oil recovery. *SPE/DOE 14th Symposium on Improved Oil Recovery. Tulsa, Oklahoma, 17-21 April 2001 SPE 89379* (2004). (Citado en la página 116.)
- [30] ZEINIJAHRMI, A., AND BEDRIKOVETSKY., P. Physics mechanisms of enhanced recovery by fines-migration-assisted waterflooding (laboratory study). *SPE European Formation Damage Conference and Exhibition. Noordwijk, The Netherlands, 5-7 June 2013 SPE 165190* (2013). (Citado en la página 112.)
- [31] ZEINIJAHRMI, A., NGUYEN, T., AND BEDRIKOVETSKY., P. Mathematical model for fines-migration-assisted waterflooding with induced formation damage. *SPE journal* (June 2013), 518–533. (Citado en la página 112.)
- [32] ZHANG, Y., AND MORROW., N. R. Comparison of secondary and tertiary recovery with change in injection brine composition for crude oil/sandstone combinations. *SPE/DOE Symposium on Improved Oil Recovery. Tulsa Oklahoma 22-26 April 2006 SPE 99757* (2006). (Citado en la página 114.)