

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Simulación Numérica de Flujo Bifásico Incompresible en Medios Porosos: Comparativa de Rendimiento del IMPES Clásico versus IMPES Mejorado

TESIS

Que para obtener el título de

INGENIERO PETROLERO

P R E S E N T A

LÓPEZ HERNÁNDEZ ABRAHAM

DIRECTOR DE TESIS

DR. VÍCTOR LEONARDO TEJA JUÁREZ



Ciudad Universitaria, Cd. Mx., 2022

AGRADECIMIENTOS

Con un eterno agradecimiento a la Universidad Nacional Autónoma de México por la mejor de las oportunidades en mi vida. Que desde mi adolescencia ha sido mi casa. A la Facultad de Ingeniería quien terminó de forjarme en muchos aspectos y que en ella conocí a muchas personas que forman parte de mi historia.

A mi director de tesis el Dr.V.L.Teja Juárez que con su gran calidad y paciencia me guio en la etapa final de mi recorrido por la Facultad de Ingeniería.

A mis sinodales de tesis M.I. José Ángel Gómez Cabrera, Dr. Rodolfo Gabriel Camacho Velázquez, Dr. Erick Luna Rojero, Dr. Víctor Hugo Arana Ortiz quienes me brindaron el ultimo impulso para continuar y terminar este recorrido.

Agradezco profundamente al proyecto PAPIIT:IA106820 *por facilitar los recursos computacionales para la realización de este trabajo de grado.*

Y finalmente a dios, quienquiera que este sea, por prestarnos vida a mí y mi núcleo que son los seres que más amo y valoro, y por darme la fuerza para poder concluir este largo trecho junto a ellos.

DEDICATORIA

Con especial cariño a mis padres Víctor Mario López García y Gregoria Hernández Rosas quienes con duro sacrificio lucharon para que pudiera llegar hasta estas instancias, que con su amor lograron que saliera adelante y que me apoyaron incondicionalmente en los buenos y malos momentos, pero sobre todo que nunca perdieron la fe en mí.

ÍNDICE

RESUMEN ..... **VIII**

1. SIMULACIÓN DE YACIMIENTOS **10**

1.1 Introducción **10**

1.2 Vista a la Simulación de Yacimientos **10**
1.2.1 Métodos Analógicos. 10
1.2.2 Métodos Matemáticos 11
1.2.2.1 Balance de Materia 11

1.3 Descripción de la Simulación de Yacimientos **11**
1.3.1 Etapas de la Simulación 12

1.4 Clasificación de Simuladores de Yacimientos **12**

1.5 Aplicación de la Simulación de Yacimientos **15**

1.6 Objetivos Generales y Particulares. **17**
1.6.1 Objetivo General 17
1.6.2 Objetivos Particulares 17

1.7 Estructura de la Tesis **17**

1.8 Justificación **18**

2. SOLUCIONADORES DE SISTEMAS LINEALES **20**

2.1 Introducción **20**

2.2 Método de Diferencias Finitas **20**
2.2.1 Diferencias Finitas de Primer Orden 23
2.2.1.1 Diferencias Finitas Progresivas o Adelantadas 23
2.2.1.2 Diferencias Finitas Regresivas 23
2.2.1.3 Diferencias Finitas Centrales 24
2.2.2 Diferencias Finitas de Segundo Orden 25
2.2.2.1 Diferencias Finitas Centrales 25
2.2.3 Diferencias Finitas para Bloques Centrados 25
2.2.4 Derivadas Temporales a partir del Esquema θ 28

2.3 Algoritmos para resolver Sistemas de Ecuaciones Lineales 29
2.3.1 Métodos Directos e Iterativos 29
2.3.2 Algoritmo de la Descomposición *LU* 30
2.3.3 Algoritmo de Thomas 33
2.3.4 Gradiente Conjugado (CG) 36
2.3.5 Algoritmo BICGSTAB 39
2.3.6 Algoritmo GMRES 41
2.3.7 Optimización de Algoritmos mediante la Implementación de Matrices Dispersas CSR 45

3. MODELADO NUMÉRICO DE FLUJO BIFÁSICO. 49

3.1 Flujo de Fluidos en Medios Porosos 49

3.2 Discretización del Yacimiento 51

3.3 Flujo Multidimensional en Coordenadas Cartesianas 53
3.3.1 Identificación de Bloques y Ordenamiento de Nodos 53

3.4 Formulación Presión - Saturación 55
3.4.1 Ecuación de presión 55
3.4.2 Ecuación de Saturación 57

3.5 Modelo Conceptual para Flujo Incompresible 58

3.6 Modelo Numérico de Flujo Bifásico Incompresible 60
3.6.1 Solución Numérica de Flujo Bifásico Mediante los Métodos IMPES Clásico e IMPES Mejorado 60

3.7 Enfoques de Solución en el Tiempo 62
3.7.1 IMPES Clásico 62
3.7.1 IMPES Mejorado 64

4. COMPARATIVA DE RENDIMIENTO DEL IMPES CLÁSICO VERSUS IMPES MEJORADO 68

4.1 Implementación Computacional 68
4.1.1 Algoritmo General 68
4.1.2 Implementación del formato de Matrices dispersas CSR 71

4.2 Verificación de los Resultados 72
4.2.1 Buckley-Leverett 72
4.2.2 Problema de los 5 Pozos 76

4.3 Resultados de la Comparación de la Simulación Numérica 88
4.3.1 Comparativa de Rendimiento 88
4.3.1.1 Modelo Unidimensional: Medio Isotrópico 88
4.3.1.1 Modelo Unidimensional: Medio Anisotrópico 91
4.3.1.2 Modelo Bidimensional: Medio Isotrópico 94
4.3.1.3 Modelo Bidimensional: Medio Anisotrópico 99
4.3.1.4 Comparativa utilizando una Matriz Dispersa 102
CONCLUSIONES 105
RECOMENDACIONES 106

5. REFERENCIAS 107

ANEXO A. Nomenclatura *110*
ANEXO B. Modelado de Pozos de Peaceman *112*
ANEXO C. Discretización del problema. *114*
ANEXO D. Implementación computacional *121*

LISTA DE TABLAS

Tabla 4.1-Parámetros para la Simulación del Problema de Buckley-Leverett *73*
Tabla 4.2. Permeabilidades Relativas krw *y* kro *74*
Tabla 4.3.Parámetros Para la Simulación del Problema de los 5 pozos *77*
Tabla 4.4 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Isotrópico. *89*
Tabla 4.5 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Isotrópico. *89*
Tabla 4.6 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Isotrópico. *89*
Tabla 4.7 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Isotrópico. *89*
Tabla 4.8. Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Isotrópico. *90*
Tabla 4.9 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Anisotrópico. *92*
Tabla 4.10 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Anisotrópico. *92*
Tabla 4.11 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Anisotrópico. *92*
Tabla 4.12 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Anisotrópico *92*
Tabla 4.13 Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Anisotrópico. *93*
Tabla 4.14 Tiempo de Ejecución Promedio de los Solvers mediante el IMPES Clásico para un Paso de Tiempo de 0.25 días en un Medio Isotrópico. *95*
Tabla 4.15 Tiempo de Ejecución Promedio de los Solvers mediante el IMPES Mejorado para un Paso de Tiempo de 0.25 días en un Medio Isotrópico. *95*
Tabla 4.16 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Isotrópico. *96*
Tabla 4.17 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 0.25 días en un Medio Isotrópico. *96*
Tabla 4.18 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Isotrópico. *97*
Tabla 4.19 Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Isotrópico *98*
Tabla 4.20 Tiempo de Ejecución Promedio de los Solvers mediante el IMPES Clásico para un Paso de Tiempo de 0.25 días en un Medio Anisotrópico. *99*

Tabla 4.21 Tiempo de Ejecución Promedio de los Solver mediante el IMPES Mejorado para un Paso de Tiempo de 1 día en un Medio Anisotrópico. 99
Tabla 4.22 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Anisotrópico. 100
Tabla 4.23 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1día en un Medio Anisotrópico. 100
Tabla 4.24 Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Anisotrópico. 101
Tabla 4.25 Tiempo Total de Ejecución del IMPES Mejorado con Matriz Dispersa. 102

LISTA DE FIGURAS

Figura 1.1.Etapas de la Simulación de Yacimientos. *13*
Figura 1.2 Clasificación General de los Simuladores de Yacimientos. *14*
Figura 1.3 Arreglo del Mallado Tridimensional para un Anticlinal *16*
Figura 1.4 Disposición de los bloques de un Mallado Tridimensional para una Capa de Arena con Espesor Variable. . *16*
Figura 2.1 Tipos de Mallado. *21*
Figura 2.2. Diferencias Finitas a partir del Cálculo Infinitesimal. *22*
Figura 2.3 Esquema de un esténcil de cinco puntos. *26*
Figura 2.4 Matriz Tridiagonal. *31*
Figura 2.5 Matriz Triangular Inferior ***L.*** *31*
Figura 2.6 Matriz Triangular Superior ***U.*** *32*
Figura 2.7 Algoritmo de Descomposición LU. *33*
Figura 2.8 Algoritmo de Thomas 1D. *35*
Figura 2.9 Algoritmo CG. *38*
Figura 2.10 Algoritmo BiCGSTAB. *40*
Figura 2.11 Algoritmo de Arnoldi. *42*
Figura 2.12 Algoritmo GMRES. *44*
Figura 2.13 Esquema de Matriz Dispersa. *46*
Figura 2.14 Almacenamiento de Matrices en formato CSR. *47*
Figura 3.1 Relación entre un bloque y sus bloques adyacentes. *52*
Figura 3.2 Relación de un bloque y sus bloques adyacentes para flujo en 1D, 2D y 3D utilizando notación de ingeniería. *52*
Figura 3.3 Notación ingenieril para identificación de bloques. *53*
Figura 3.4 Esquemas de Ordenamiento usados en la Simulación de Yacimientos. *54*
Figura 4.1.Algoritmo IMPES Clásico. *69*
Figura 4.2. Algoritmo IMPES Mejorado. *70*
Figura 4.3 Almacenamiento de Matrices en formato CSR para implementación con biblioteca SciPy. *71*
Figura 4.4. Medio Homogéneo de Longitud Lx, inicialmente Saturado con Aceite y Agua. *72*
Figura 4.5 Perfiles de SaturaciónSw. *75*
Figura 4.6 Perfiles de Presión po. *75*
Figura 4.7 Perfiles de Saturación Sw. *75*
Figura 4.8 Perfiles de Presión po. *75*
Figura 4.9. Dominio físico del modelo "five spot". *76*
Figura 4.10. Perfiles de Presión de la fase aceite po (psi) para un tamaño de malla de 90x90 celdas. *79*
Figura 4.11. Perfiles de Saturación Sw para un tamaño de malla de 90x90 celdas. *80*
Figura 4.12 Perfiles de Presión po y Saturación Sw en un medio Anisotrópico para una malla de 90 x 90 celdas. *81*
Figura 4.13 Perfiles de Velocidad de las Fases Agua y Aceite en un Medio Isotrópico. *83*

Figura 4.14 Perfiles de Velocidad de las Fases Agua y Aceite en un Medio Anisotrópico. 84
Figura 4.15 Perfiles de Líneas de Corriente de las Fases Agua y Aceite en un Medio Isotrópico. 85
Figura 4.16 Perfiles de Líneas de Corriente de las Fases Agua y Aceite en un Medio Anisotrópico. 86
Figura 4.17 Producción de Fluidos. 87
Figura 4.18 Flujo Fraccional. 87
Figura 4.19 Gráfica del Rendimiento en el Tiempo de Ejecución del Solver entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico. 91
Figura 4.20 Gráfica del Rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico. 91
Figura 4.21 Gráfica de Rendimiento en el Tiempo de Ejecución del Solver entre los Algoritmos IMPES Clásico e IMPES Mejorado un Medio Anisotrópico. 93
Figura 4.22 Gráfica del Rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Anisotrópico. 93
Figura 4.23 Gráfica del Rendimiento en el Tiempo de Ejecución de los Solvers entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico. 97
Figura 4.24 Gráfica del Rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico. 97
Figura 4.25 Gráfica del Rendimiento en el Tiempo de Ejecución entre los algoritmos IMPES Clásico e IMPES Mejorado un Medio Isotrópico con un Paso de Tiempo de 1 día. 98
Figura 4.26 Gráfica del Rendimiento en el Tiempo de ejecución de los Solvers entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Anisotrópico con un Paso de Tiempo de 0.25 y [1 días]. 101
Figura 4.27 Gráfica del rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado un Medio Anisotrópico con un Paso de Tiempo de 0.25 y 1 [días]. 101
Figura 4.28 Comportamiento del Tiempo de Ejecución. 104
Figura 5.1 Representación de un pozo en una celda y a escala de yacimiento. 113
Figura 5.2 Ordenamiento de bloques para una malla de n nodos. 114
Figura 5.3 Matriz resultante de la Discretización del Yacimiento. 115
Figura 5.4 Algoritmo del Simulador basado en el IMPES Clásico. 122
Figura 5.5 Algoritmo del Simulador basado en el IMPES Mejorado. 122
Figura 5.6 Algoritmo de la implementación CSR. 122

RESUMEN

La presente tesis utiliza los modelos matemáticos de flujo bifásico incompresible en medios porosos en dos dimensiones con el fin de desarrollar una herramienta práctica para simular numéricamente la inyección de agua en yacimientos de hidrocarburos basada en los algoritmos IMPES Clásico e IMPES Mejorado. El modelo matemático se obtiene a partir de la formulación axiomática. La discretización del sistema de ecuaciones resultantes se llevó a cabo mediante el método de diferencias finitas para bloques centrados, que se deriva a partir de la forma conservativa de las ecuaciones de balance. El sistema discreto, cumple con el principio de conservación de las propiedades extensivas del modelo para cada uno de los bloques en la malla. La relación que existe entre el algoritmo numérico y el principio de conservación es una de las cualidades más útiles de este método.

Posteriormente se discuten y comparan los resultados que se obtienen tras ejecutar el algoritmo que resuelve el modelo numérico para flujo bifásico incompresible, bajo dos algoritmos numéricos diferentes: Se trata del IMPES [1] (*Implicit Pressure - Explicit Saturation*) Clásico e IMPES Mejorado [2]. En el IMPES Clásico se resuelve la ecuación de presión con un esquema de discretización en el tiempo implícito y la saturación explícitamente considerando un mismo paso de tiempo para ambas ecuaciones. Es importante señalar que, este paso de tiempo debe ser lo suficientemente pequeño para que este método sea estable. Por otra parte, en el IMPES Mejorado se tiene un paso de tiempo diferente para la solución de la ecuación de la saturación, con el objetivo de controlar la variación máxima de la misma y con ello estabilizar el método. Cabe señalar que este dato es proporcionado por el usuario. [3]. Se lleva a cabo una comparativa de rendimiento para los casos unidimensional y bidimensional; mediante variaciones del solver de ecuaciones lineales, tamaño de malla, propiedades de flujo como lo son permeabilidad relativa y presión capilar, también se considera la anisotropía para el caso bidimensional. Esto con el objetivo de conocer cuantitativa y cualitativamente los parámetros numéricos para aplicar uno o el otro.

CAPÍTULO I

SIMULACIÓN DE YACIMIENTOS

1. Simulación de Yacimientos

1.1 Introducción

Un yacimiento petrolero es un medio poroso que contiene hidrocarburos [4]. El objetivo principal de la simulación de yacimientos es predecir el desempeño futuro del yacimiento y encontrar formas y medios de optimizar la recuperación de algunos de los hidrocarburos bajo diversas condiciones de operación [5]. El procedimiento general para predecir el comportamiento de un sistema es el modelado. En este proceso, se construyen modelos que se utilizan para predecir el comportamiento de un sistema. Un modelo de un sistema es un sustituto cuyo comportamiento imita el del sistema. La simulación implica cuatro etapas principales de modelado interrelacionadas [6]: establecimiento de modelos físicos, desarrollo de modelos matemáticos, discretización de estos modelos y diseño de algoritmos informáticos, y requiere una combinación de habilidades de físicos, matemáticos, ingenieros de yacimientos e informáticos.

Los modelos más utilizados en la actualidad son los modelos matemáticos, ya que son los más versátiles y económicos. En aplicaciones específicas, los modelos matemáticos consisten en códigos de computadora que se adaptan fácilmente para ajustarse a los cambios en las propiedades de los sistemas. Además, los fundamentos y métodos de aplicación de los modelos matemáticos poseen una notable unanimidad y generalidad conceptual. Estas características tienen implicaciones prácticas muy importantes, ya que conducen a grandes inversiones de esfuerzo y recursos.

Los modelos matemáticos integran conocimientos científicos y tecnológicos con el propósito de predecir el comportamiento del sistema. Por lo tanto, dicho conocimiento, se incorpora a los códigos computacionales que las computadoras ejecutan en la utilización del modelo. En la actualidad, los modelos matemáticos basados en la simulación numérica permiten el estudio de complejos sistemas y fenómenos naturales que de otro modo serían muy costosos, peligrosos o incluso imposibles de estudiar mediante experimentación directa.

1.2 Vista a la Simulación de Yacimientos.

1.2.1 Métodos Analógicos.

Antes de perforar, cuando los datos son limitados o no están disponibles, el único método que los ingenieros de yacimientos pueden usar para realizar un análisis económico es el de la analogía. En este método, se utilizan yacimientos en la misma cuenca geológica o provincia o yacimientos con propiedades petrofísicas similares para predecir el desempeño del yacimiento objetivo. Este método se puede utilizar para estimar los factores de recuperación, las tasas de producción inicial, las tasas de declinación, el espaciamiento entre pozos y los mecanismos de recuperación. El método analógico puede arrojar resultados confiables cuando se comparan dos yacimientos similares y se utilizan estrategias de desarrollo similares. Sin embargo, el método sufre si se consideran diferentes estrategias de desarrollo. Además, las sensibilidades de "*qué pasaría si*" no se pueden investigar.

Una forma de analogía es la prueba de campo por etapas que proporciona las predicciones más confiables para las operaciones de recuperación secundaria y mejorada. En este método, los patrones de pozos representativos en un campo que es candidato para la recuperación secundaria o mejorada se convierten en el nuevo proceso y se monitorea el rendimiento de la producción. Los resultados de la prueba de campo, que pueden tardar 1 o 2 años en obtenerse, se aplican a los patrones de pozo restantes y se puede predecir el rendimiento del campo. Las gerencias generalmente confían en las decisiones tomadas sobre la base de los resultados de una prueba de campo por etapas.

1.2.2 Métodos Matemáticos

Los métodos matemáticos son los métodos clásicos de ingeniería de yacimientos más utilizados en la industria petrolera para predecir el comportamiento de los yacimientos, reproduciendo la física de los fluidos involucrada en ello. Estos métodos incluyen balance de materia, curvas de declinación, métodos estadísticos y analíticos. En general, los procedimientos gráficos o manuales son suficientes cuando se aplican estos métodos. Sin embargo, con la mejora de las computadoras, muchos paquetes de software para estos métodos ya están disponibles. Aunque todas las bases teóricas tienen sus limitaciones, existe una fuerte relación entre estos modelos y la simulación numérica

1.2.2.1 Balance de Materia

Particularmente los métodos Clásicos de balance de materia utilizan una representación matemática de un yacimiento o volumen del drene. El principio básico de este modelo se basa en la conservación de la masa; es decir, la cantidad de masa de agua, petróleo o gas que queda en el yacimiento después de un intervalo de producción es igual a la diferencia de la cantidad de masa originalmente en el lugar y que se extrajo del yacimiento debido a la producción, más la cantidad de masa añadida debido a la inyección y la invasión. El método de balance de materia es simplemente un inventario de todos los fluidos que entran, salen y permanecen en el yacimiento.

Las ecuaciones de balance de materia contienen gran parte de la física que gobierna la producción de los yacimientos de hidrocarburos. Sin embargo, la confiabilidad del análisis de balance de materia depende de la precisión de los datos disponibles y del grado en que se cumplan los supuestos subyacentes.

1.3 Descripción de la Simulación de Yacimientos

La simulación de yacimientos se ha convertido en una herramienta predictiva estándar en la industria petrolera. Se puede utilizar para obtener predicciones de rendimiento precisas para un yacimiento de hidrocarburos en diferentes condiciones de operación. Un proyecto de recuperación de hidrocarburos generalmente involucra una inversión de capital de cientos de millones de dólares, y el riesgo asociado con las estrategias de desarrollo y producción seleccionadas debe evaluarse y minimizarse.

1.3.1 Etapas de la Simulación

La simulación de yacimientos involucra cuatro etapas principales relacionadas entre sí, la Figura 1.1 ilustra las etapas principales involucradas en el desarrollo de un simulador para ver su formulación, discretización, representación de pozos, linealización, solución y validación [7]. En primer lugar, se desarrolla un modelo físico de procesos relevantes que incorpora la física clásica para describir las características esenciales de los fenómenos físicos que están detrás del comportamiento de un yacimiento. En segundo lugar, se desarrolla un modelo matemático mediante un sistema de ecuaciones diferenciales, parciales, no lineales, dependientes del tiempo y acoplado; las cuales describen los fenómenos del modelo físico conceptual. En tercer lugar, se deriva y analiza un modelo numérico con las propiedades básicas de los modelos físico y matemático. En cuarto lugar, se desarrollan algoritmos informáticos (y códigos) para resolver de manera eficiente los sistemas de ecuaciones algebraicas lineales y no lineales que surgen de la discretización numérica.

Cada una de estas etapas es esencial para la simulación de yacimientos, y algunas veces es necesario un número de iteraciones entre estas etapas para ajustar los modelos físicos, matemáticos y numéricos y los algoritmos informáticos para que se pueda obtener un pronóstico preciso del rendimiento del yacimiento.

1.4 Clasificación de Simuladores de Yacimientos

Los simuladores de yacimientos se pueden clasificar desde diferentes perspectivas. Los criterios más comunes se basan en el tipo de fluidos de yacimiento que se estudian y los procesos de recuperación que ocurren en el yacimiento en cuestión. Otros enfoques incluyen el número de dimensiones (1D, 2D y 3D), el número de fases (monofásica, bifásica y trifásica) y el sistema de coordenadas utilizado en el modelo (rectangular, cilíndrico y esférico) [8] . El tipo de simuladores de yacimientos también se puede determinar por la estructura de la roca (ordinaria, doble porosidad/permeabilidad y fracturamiento y flujo hidráulico/térmico acoplado). Los simuladores de yacimientos basados en la clasificación del tipo de fluidos de yacimiento incluyen simuladores de gas, aceite negro y composicional. Los simuladores de aceite negro, son simuladores de recuperación convencionales y se utilizan en los casos en que los procesos de recuperación no son sensibles a los cambios de composición en los fluidos del yacimiento. Los simuladores composicionales se utilizan cuando los procesos de recuperación son sensibles a los cambios de composición e incluyen el agotamiento primario de yacimientos de gas condensado y aceite volátil, operaciones de mantenimiento de presión en estos yacimientos y procesos miscibles de contacto múltiple. Los simuladores de yacimientos clasificados mediante procesos de recuperación incluyen la recuperación convencional (aceite negro), el desplazamiento miscible, la recuperación térmica y los simuladores de inundación química. Los procesos primarios de recuperación de aceite, como el empuje hidráulico, el empuje por gas disuelto, la expansión del casquete de gas, el empuje por segregación gravitacional y la imbibición capilar, se pueden modelar con los simuladores de aceite negro.

Además, las etapas de recuperación secundaria, como la inyección de agua o gas, también se pueden modelar con estos simuladores. Los mecanismos de recuperación térmica implican inundaciones de vapor o combustión in situ y utilizan una ecuación de conservación de energía, además de la ley de conservación de masa. Los simuladores de inundación química incluyen inundación alcalina, surfactante, polímero y/o espuma y se pueden usar para cambiar la relación de movilidad dinámicamente en el desplazamiento (polímero) o movilizar el aceite residual (surfactante). Se deben considerar efectos adicionales como la adsorción en la roca, la reducción de la permeabilidad y el fluido no Newtoniano. La Figura 1.2 intenta resumir una clasificación general de los simuladores de yacimientos bajo posibles marcos de trabajo con la finalidad de explicar los atributos de los modelos existentes y los tipos de simulación que podrían realizarse con ellos, la Figura 1.2 busca definir seis parámetros de selección: Tipo de yacimiento, Nivel de simulación, Tipo de simulación, Tipo de fluido en el yacimiento, Número de dimensiones y la geometría que se supone para el yacimiento.

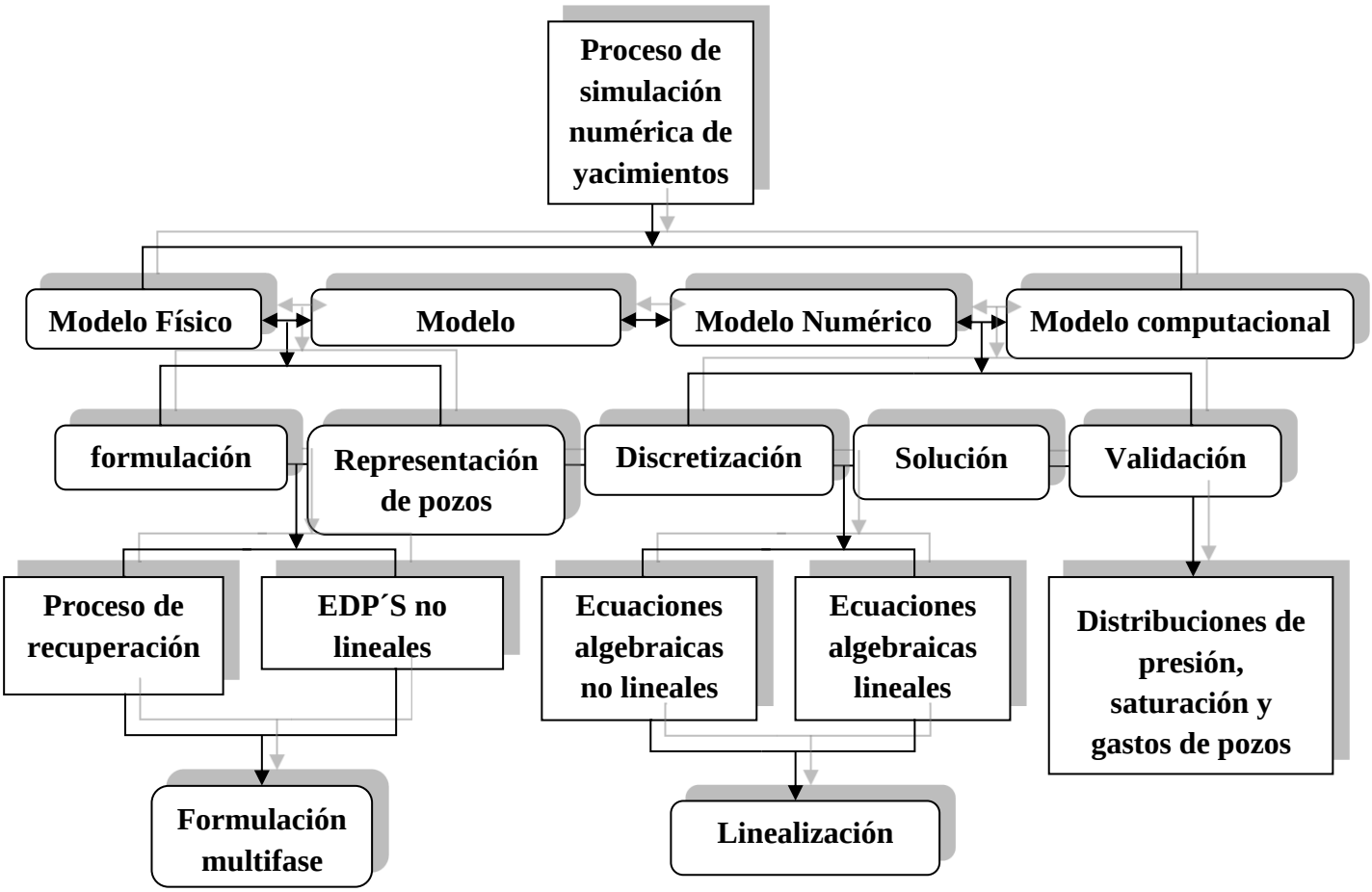


Figura 1.1.Etapas de la Simulación de Yacimientos.

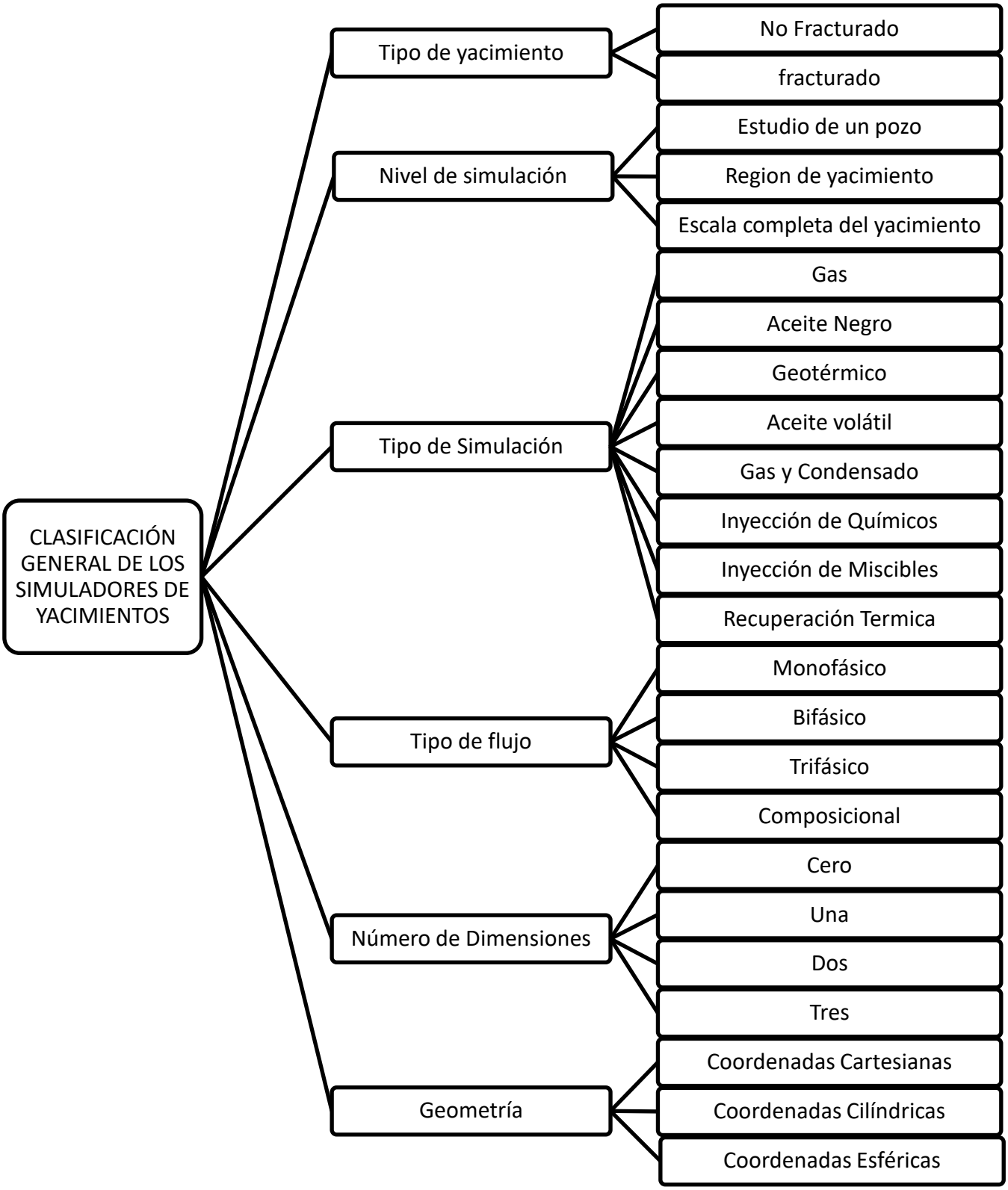


Figura 1.2 Clasificación General de los Simuladores de Yacimientos.

1.5 Aplicación de la Simulación de Yacimientos

La simulación de yacimientos generalmente se aplica en los siguientes pasos [1]:

Establecer objetivos de estudio. El primer paso en cualquier estudio de simulación de yacimientos es establecer objetivos claros y alcanzables. Estos objetivos compatibles con los datos de yacimientos y producción disponibles. Los objetivos se utilizan para establecer metas, definir la estrategia básica, identificar los recursos disponibles y determinar qué se aprenderá del estudio.

Recopilar y validar datos de yacimientos. Una vez que se han establecido los objetivos de la simulación, se recopilan los datos del yacimiento y la producción. Solo los datos requeridos para cumplir con los objetivos del estudio deben incorporarse al modelo de simulación.

Diseñar el simulador de yacimientos. Una vez recopilados y validados los datos, se diseña el simulador. Este paso involucra las cuatro etapas principales interrelacionadas descritas anteriormente: construcción de un modelo físico conceptual, desarrollo de modelos matemáticos y numéricos y diseño de códigos de computadora. En este paso, el yacimiento se divide en mallas, como en la Figura 1.3 y la Figura 1.4. Las propiedades de la formación, como la porosidad, las permeabilidades direccionales y el espesor de la zona productiva neta, se asignan a estas celdas de la malla. Las diferentes celdas de la malla pueden tener diferentes propiedades del yacimiento; sin embargo, se supone que las propiedades del yacimiento son homogéneas dentro de cada una celda del mallado.

Coincidencia histórica con el simulador de yacimientos. Una vez construido el simulador de yacimientos, debe ajustarse, o compararse con el historial, con los datos disponibles de yacimientos y producción, ya que es necesario verificar gran parte de los datos de un simulador típico.

Hacer predicciones. En el paso final de aplicación, se evalúan varios planes de desarrollo y producción, y se lleva a cabo un análisis de sensibilidad de varios parámetros de yacimiento y producción. Si bien la simulación de yacimientos es el método más completo, los métodos clásicos de ingeniería de yacimientos aún se utilizan para predecir el rendimiento de yacimientos. Estos métodos clásicos se pueden usar para generar datos de entrada para simuladores de yacimientos. Por ejemplo, se puede emplear un análisis de acumulación de presión para obtener la permeabilidad de la formación en la caracterización de yacimientos, y los métodos de balance de materia brindan información sobre la invasión de agua y el tamaño del acuífero durante la comparación histórica.

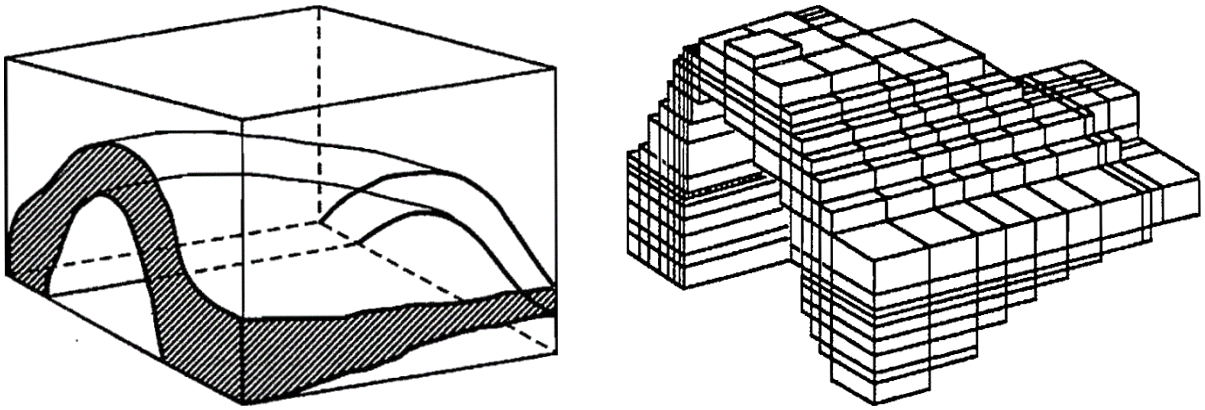


Figura 1.3 Arreglo del Mallado Tridimensional para un Anticlinal, tomada de [1] .

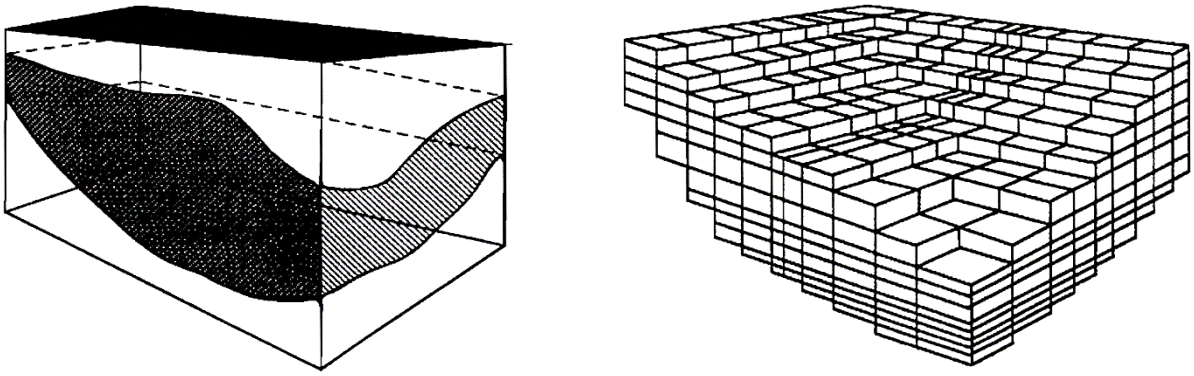


Figura 1.4 Disposición de los bloques de un Mallado Tridimensional para una Capa de Arena con Espesor Variable, tomada de [1] .

El objetivo principal de cualquier estudio de simulación es obtener conocimiento del yacimiento en cuestión. En la mayoría de los estudios de simulación, la mayor parte del conocimiento se obtiene durante las fases de recopilación de datos, coincidencia del historial de producción y predicción. Durante las fases de recopilación de datos y comparación histórica, todos los datos relevantes del yacimiento se recopilan, validan y sintetizan en un modelo de campo coherente. Este proceso inevitablemente arrojará información sobre el yacimiento que se desconocía antes del estudio. Durante la fase de predicción, se pueden abordar las preguntas relacionadas con el yacimiento en cuestión y se cumplen la mayoría de los objetivos del estudio.

1.6 Objetivos Generales y Particulares.

1.6.1 Objetivo General

Desarrollar dos códigos numéricos con el objetivo de llevar a cabo una comparativa de rendimiento computacional de los algoritmos "Implícito en Presión y Explicito en Saturación" para resolver flujo bifásico incompresible en medios porosos.

1.6.2 Objetivos Particulares

1 Investigar el modelo matemático y numérico del flujo bifásico incompresible.
2 Desarrollar los códigos numéricos necesarios en un lenguaje de alto nivel sencillo y portable para resolver el flujo bifásico 1D y 2D, mediante los algoritmos IMPES e IMPES Mejorado.
3 Comparar los resultados del código con los resultados reportados en la literatura.
4 Comparar el rendimiento numérico-computacional de los códigos.

1.7 Estructura de la Tesis

En el siguiente trabajo se desarrollan cuatro capítulos que se ordenan de la siguiente forma:

- En el capítulo 2 se presenta una breve introducción a los métodos de diferencias finitas, también se da un repaso a los métodos numéricos para la solución de sistemas ecuaciones lineales directos e iterativos y una técnica de optimización implementado, matrices dispersas.
- En el capítulo 3 se describe el modelado numérico de flujo bifásico incompresible en medios porosos; abarcando la formulación axiomática, seguido de la formulación de flujo fraccional para obtener las ecuaciones de presión y saturación, finalmente se describe los algoritmos IMPES Clásico e IMPES Mejorado.
- Por otra parte, en el capítulo 4 se presenta la implementación de los algoritmos antes mencionados y se analizan los resultados numéricos obtenidos de estas implementaciones.
- Por último, en el capítulo 5 se presentan las conclusiones de este trabajo. Adicionalmente, se incluyen cuatro anexos: El Anexo A, en el que se indica la nomenclatura utilizada. El Anexo B contiene el desarrollo y aplicación de la técnica de diferencias finitas con el que se desarrolla el modelo numérico. El Anexo C describe el modelo de pozo empleado para el modelo numérico. El Anexo D revisa la implementación del formato CSR para matrices dispersas.

1.8 Justificación

El algoritmo IMPES *(Implicit Pressure Explicit Saturation*) se aplica a modelos de simulación que involucran ecuaciones de conservación para simular el comportamiento del flujo de fluidos en medios porosos, y es eficaz para reproducir el desplazamiento de fases inmiscibles, especialmente en yacimientos de aceite, por lo que se vuelve adecuado en la simulación de un proceso de recuperación secundaria de inyección de agua. También este método es simple de configurar, eficiente de implementar, y requiere menos memoria computacional en comparación con otros métodos [9]. Aunque este método presenta dificultades de estabilidad y convergencia, con pasos de tiempo demasiado grandes y con celdas demasiado pequeñas; para el caso específico de dos fases no inmiscibles, éste describe con detalle las saturaciones dinámicas y perfiles de presiones, porque resuelve problemas que requieren exactitud temporal, por lo que puede describir el movimiento de los fluidos, sí, se seleccionan un paso de tiempo y un tamaño de malla adecuados. Además, el requerimiento de un mallado fino en las regiones donde se tiene fuentes o sumideros se vuelve innecesario debido al uso del modelo de pozos de Peaceman generalizado, lo cual reduce variaciones que pueden arrojar malas predicciones de presiones promedio, y gradientes de presión o saturación, por lo que puede modelar debidamente los perfiles de producción de fluidos atribuidos a la inyección de agua y, su dependencia con la presión y saturación que existen en sus vecindades. Cuando se considera un simulador bidimensional y un espesor pequeño con respecto al área del yacimiento, un simulador basado en el IMPES puede simular adecuadamente los efectos de barrido de aceite al inyectar agua. En la selección de pasos de tiempo adecuados se debe considerar que incrementos muy grandes reducirán la calidad de los resultados, mientras que incrementos demasiado pequeños incrementarán el tiempo de cómputo. Por otra parte, las dos versiones más utilizadas del algoritmo IMPES son: el IMPES Clásico y el IMPES Mejorado. Si se utilizan mallas relativamente finas, en ambos métodos la mayor parte del tiempo de cómputo se dedica al cálculo implícito de la presión, por ello se debe hacer un análisis para determinar el desempeño de los solvers de ecuaciones lineales disponibles. Adicionalmente, de la mecánica del flujo de fluidos en medios porosos se deduce que la presión cambia menos rápidamente en el tiempo que la saturación. Además, para el IMPES Clásico la restricción de los pasos de tiempo se usa principalmente en el cálculo explícito de la saturación, por lo que en esto difiere el IMPES Mejorado, puesto que requiere un paso de tiempo mucho mayor para la presión que para la saturación, o bien dos pasos de tiempo diferentes. Por las razones previamente expuestas, se abre paso la necesidad de estudiar los factores que intervengan en el desempeño de las variantes de los algoritmos, para poder conocer los límites de su uso y saber cuándo utilizar uno u otro, aprovechando las bondades de un lenguaje de alto nivel (Python). En esta Tesis, primero se revisa el IMPES Clásico y luego el IMPES Mejorado. Se presentan experimentos numéricos de ambos algoritmos para comprobar su coste computacional y su estabilidad.

CAPÍTULO II

SOLUCIONADORES DE SISTEMAS LINEALES

2. Solucionadores de Sistemas Lineales

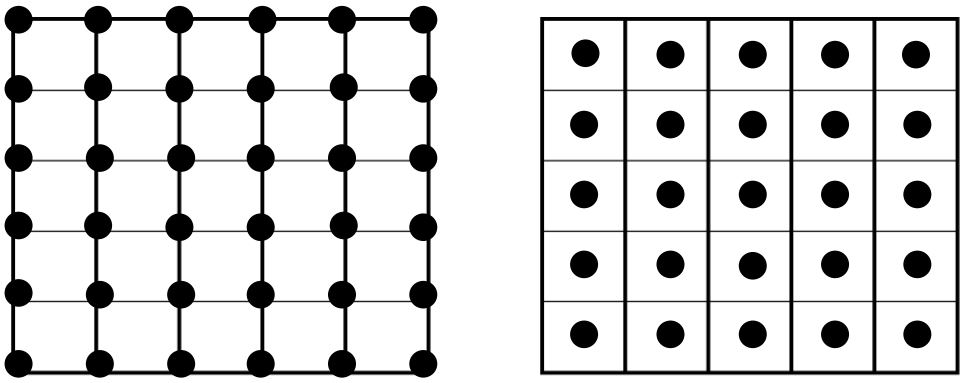
2.1 Introducción

Para un simulador de yacimientos de petróleo con una cantidad de bloques de malla del orden de 100 000, alrededor del 80 % al 90 % del tiempo total de simulación se dedica a la solución del sistema de ecuaciones lineales resultante [2]. Por lo tanto, la elección de un solucionador lineal rápido es crucial en la simulación de yacimientos. En general, la matriz de coeficientes del sistema de ecuaciones que surge en la simulación numérica de yacimientos es escasa o rala, altamente asimétrica y mal condicionada. Si bien es escasa, su estructura bandeada generalmente se ve afectada por pozos que perforan muchos bloques de malla, o por una estructura de malla irregular. Además, la dimensión M de la matriz a menudo oscila entre cientos y millones. Para la solución de tales sistemas, los algoritmos basados en los subespacios de Krylov son una buena opción. Tres de estos algoritmos iterativos líderes son el CGN (la iteración de gradiente conjugado), GMRES (minimización residual en un espacio de Krylov) y BICGSTAB (un método de biortogonalización adaptado de la iteración de gradiente biconjugado). Estos tres algoritmos difieren fundamentalmente en sus capacidades. Se pueden construir ejemplos de matrices para mostrar que cada tipo de iteración puede superar a los demás por un factor del orden de $M \times M$ [10]. Además, estos algoritmos a menudo necesitan un preacondicionamiento. El alcance de este trabajo no profundiza en los algoritmos del subespacio de Krylov y sus versiones preacondicionadas, sino que se proporcionan algunas pautas sobre la elección de un algoritmo adecuado para un problema dado.

2.2 Método de Diferencias Finitas

Las Ecuaciones Diferenciales Parciales (EDP) derivadas durante la etapa de formulación, si se resolvieran analíticamente, darían la presión del yacimiento, las saturaciones de fluidos y los caudales de los pozos como funciones continuas de espacio y tiempo. Sin embargo, debido a la naturaleza altamente no lineal de las EDP, no pueden utilizarse técnicas analíticas y las soluciones deben obtenerse con métodos numéricos. En contraste con las soluciones analíticas, las soluciones numéricas dan los valores de presión y saturación de fluidos sólo en puntos discretos del yacimiento y en tiempos diferentes. La *discretización* es el proceso de convertir las EDP en ecuaciones algebraicas [1]. Se pueden utilizar varios métodos numéricos para discretizar las EDP; sin embargo, el enfoque más común en la industria petrolera actual es el método de las diferencias finitas. El enfoque de diferencias finitas más utilizado se basa esencialmente en la expansión de las series de Taylor y desprecia los términos que se consideran pequeños cuando se consideran pequeñas diferencias en los parámetros espaciales. Esta forma expandida es un conjunto de ecuaciones algebraicas. El método de los elementos finitos, en cambio, utiliza diversas funciones para expresar las variables en las *Ecuaciones Gobernantes*. Estas funciones conducen al desarrollo de una función de error que se minimiza para generar solución a las *Ecuaciones Gobernantes*. Básicamente desde este punto de vista, el método de elementos finitos no debe considerarse como un método diferente.

Su vertiginosidad reside únicamente en su capacidad para utilizar mallas irregulares y no únicamente mallas ortogonales [11]. En el capítulo anterior se describió el concepto de una malla, sin embargo, es importante mencionar como se estructura para resolver ecuaciones diferenciales numéricamente. Existen dos principales tipos de mallados estructurados y ortogonales para el caso de utilizar Diferencias Finitas o Volumen Finito. El primero es el mallado de puntos distribuidos (*Point-distributed grid*), Figura 2.1a), y en esta malla se resuelven las ecuaciones en los vértices o esquinas de la malla. El segundo tipo de mallado es de celdas o bloques centrados (*Block-centered grid*) y en esta malla se resuelven las ecuaciones en los centros de los volúmenes de control, Figura 2.1b). Para llevar a cabo la discretización, se escribe una EDP para un punto dado del espacio en un nivel de tiempo determinado.



a) Malla de Puntos Distribuidos *b) Malla de Bloques Centrados*

Figura 2.1 Tipos de Mallado.

Las diferencias finitas representan un papel importante al momento de realizar una aproximación de una derivada; pues, una diferencia finita consiste en una aproximación de las derivadas parciales por expresiones algebraicas con los valores de la variable dependiente en un limitado número de puntos seleccionados, es decir, gracias al teorema del valor medio del cálculo diferencial es posible comprender lo que significa una derivada numérica de una función, Figura 2.2, mientras que el polinomio de Taylor es una aproximación polinómica de una función que se puede derivar hasta n veces, es decir que es una suma infinita de las derivadas de orden n de la función evaluadas en un punto concreto.

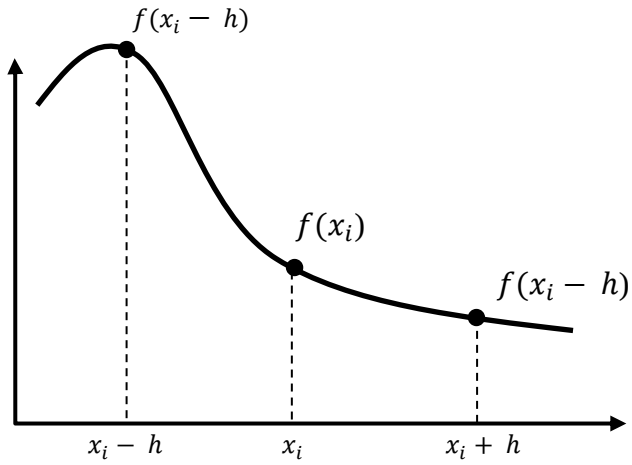


Figura 2.2. Diferencias Finitas a partir del Cálculo Infinitesimal.

Pueden obtenerse tres formas de diferencias finitas a partir de la expansión en series de Taylor:

- Diferencias progresivas o adelantadas :$f(x_i + h)$
- Diferencias regresivas o atrasadas: $f(x_i - h)$
- Diferencias centrales utilizando una combinación de la expansión en series de las anteriores.

Para deducir las diferencias se utiliza una expansión de las series de Taylor, la cual indica que cualquier función evaluada alrededor de un punto a puede expandirse en una serie polinómica dada por la siguiente ecuación [12]:

$$P(x) = \sum_{n=0}^{\infty} \frac{(x-a)^n}{n!} f^n(a) \tag{2.1}$$

por lo que una expansión de Taylor para la función $f(x)$,alrededor de un punto x_i esta dado por la sustitución de $f(x)$ en (2.1):

$$f(x) = f(x_i) + f'(x_i)(x - x_i) + \frac{f''(x_i)}{2!}(x - x_i)^2 + \frac{f'''(x_i)}{3!}(x - x_i)^3 + \cdots + \frac{f^n(x_i)}{n!}(x - x_i)^n \tag{2.2}$$

esta ecuación se usa principalmente para la aproximación de derivadas en problemas de flujo de fluidos en medios porosos para lo cual se considera una función $p(x_1,\ x_2, t)$.

2.2.1 Diferencias Finitas de Primer Orden

2.2.1.1 Diferencias Finitas Progresivas o Adelantadas

Para diferencias finitas progresivas de la ecuación (2.2) al considerar $x = x_i + h$ toma la siguiente forma:

$$f(x_i + h) = f(x_i) + f'(x_i)(x_i + h - x_i) + \frac{f''(x_i)}{2!}(x_i + h - x_i)^2$$

$$f(x_i + h) = f(x_i) + f'(x_i)(h) + \frac{f''(x_i)}{2!}(h)^2$$

despejando la derivada de primer orden en la ecuación

$$\frac{f(x_i + h) - f(x_i)}{h} - \frac{f''(x_i)}{2!}(h)^2 = f'(x_i)$$

$$f'(x_i) = \frac{f(x_i + h) - f(x_i)}{h} - \frac{f''(x_i)}{2!}h^2$$

para esta ecuación el error de truncamiento se asocia al término cuadrático de la serie de Taylor $f''(x_i)h^2/2!$ y se considera despreciable, por lo que la ecuación puede representarse como:

$$f'(x_i) = \frac{f(x_i + h) - f(x_i)}{h} \tag{2.3}$$

2.2.1.2 Diferencias Finitas Regresivas

Para diferencias finitas regresivas de la ecuación (2.2) al considerar $x = x_i - h$ toma la siguiente forma:

$$f(x_i - h) = f(x_i) + f'(x_i)(x_i - h - x_i) + \frac{f''(x_i)}{2!}(x_i - h - x_i)^2$$

$$f(x_i - h) = f(x_i) + f'(x_i)(-h) + \frac{f''(x_i)}{2!}(-h)^2$$

$$f(x_i - h) = f(x_i) + f'(x_i)(-h) + \frac{f''(x_i)}{2!}h^2$$

$$\frac{f(x_i - h) - f(x_i)}{(-h)} = f'(x_i)$$

por lo que la diferencia finita regresiva queda como:

$$f'(x_i) = \frac{f(x_i) - f(x_i - h)}{h} \tag{2.4}$$

2.2.1.3 Diferencias Finitas Centrales

Las expansiones para $f(x_i + h)$ y $f(x_i - h)$ son:

$$f(x_i + h) = f(x_i) + f'(x_i)(h) + \frac{f''(x_i)}{2!}(h)^2$$

$$f(x_i - h) = f(x_i) - f'(x_i)(h) + \frac{f''(x_i)}{2!}h^2$$

restando ambas ecuaciones se tiene:

$$f(x_i + h) - f(x_i - h)$$
$$= f(x_i) + f'(x_i)(h) + \frac{f''(x_i)}{2!}(h)^2 - \left(f(x_i) - f'(x_i)(h) + \frac{f''(x_i)}{2!}h^2\right)$$

$$= f(x_i) + f'(x_i)(h) + \frac{f''(x_i)}{2!}(h)^2 - f(x_i) + f'(x_i)(h) - \frac{f''(x_i)}{2!}h^2$$

$$f(x_i + h) - f(x_i - h) = f'(x_i)(h) + f'(x_i)(h)$$

$$f(x_i + h) - f(x_i - h) = 2f'(x_i)(h)$$

finalmente se tiene que la diferencia finita centrada es:

$$f'(x_i) = \frac{f(x_i + h) - f(x_i - h)}{2h} \tag{2.5}$$

cabe señalar que como los términos de las derivadas de segundo orden se cancelaron, el error de truncamiento para la diferencia centrada es de orden cúbico, por lo que es más exacta que las diferencias progresivas y regresivas.

2.2.2 Diferencias Finitas de Segundo Orden

2.2.2.1 Diferencias Finitas Centrales

Para obtener la diferencia centrada de segundo orden se utilizan las expansiones para $f(x_i + h)$ y $f(x_i - h)$:

$$f(x_i + h) = f(x_i) + f'(x_i)(h) + \frac{f''(x_i)}{2!}(h)^2$$

$$f(x_i - h) = f(x_i) - f'(x_i)(h) + \frac{f''(x_i)}{2!}h^2$$

sumando las ecuaciones anteriores:

$$f(x_i + h) + f(x_i - h) = f(x_i) + f'(x_i)(h) + \frac{f''(x_i)}{2!}(h)^2 + f(x_i) - f'(x_i)(h) + \frac{f''(x_i)}{2!}h^2$$

$$f(x_i + h) + f(x_i - h) = 2f(x_i) + 2\frac{f''(x_i)}{2!}(h)^2$$

despejando la derivada de segundo orden se llega a:

$$\frac{f(x_i + h) + f(x_i - h) - 2f(x_i)}{2h^2} = \frac{f''(x_i)}{2!} \quad (2.6)$$

2.2.3 Diferencias Finitas para Bloques Centrados

Como se señaló anteriormente, existen dos tipos de mallas ampliamente utilizadas en la simulación de yacimientos. Las ecuaciones en diferencias finitas tienen la misma forma para ambas mallas, además el error de truncamiento es el error en el que se incurre al reemplazar una ecuación diferencial por una ecuación en diferencias numéricas. Generalmente el esquema utilizado para problemas bidimensionales es el esténcil de cinco puntos. Para los puntos en las fronteras del dominio, se involucra un nodo ficticio o *nodo fantasma* fuera del dominio. Los valores de p en estos puntos se eliminan según el tipo de malla y las condiciones de frontera que se emplee. Para poder hablar de las diferencias finitas para bloques centrados se debe analizar el caso especial para discretizar la siguiente ecuación:

$$\frac{\partial}{\partial x}\left(\lambda \frac{\partial f(x)}{\partial x}\right) \quad (2.7)$$

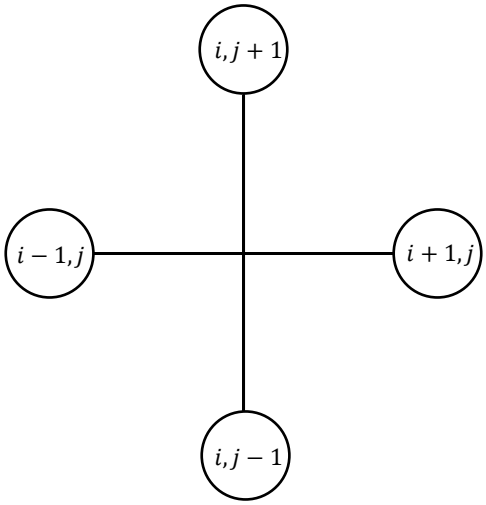


Figura 2.3 Esquema de un esténcil de cinco puntos.

En los problemas de flujo en medios porosos es usual utilizar diferencias centrales debido a su mayor exactitud al momento de resolver los problemas en comparación con las diferencias progresivas y regresivas.

A la ecuación (2.7) se le puede aplicar un cambio de variable donde

$$U = \lambda \frac{\partial f(x)}{\partial x} \tag{2.8}$$

por lo que se tiene:

$$\frac{\partial}{\partial x}\left(\lambda \frac{\partial f(x)}{\partial x}\right) = \frac{\partial U(x)}{\partial x} \tag{2.9}$$

Para resolver en diferencias finitas (2.9), se expande la función con polinomios de Taylor a $n = 1$ considerando nodos centrados donde $x = x_i + \frac{hx}{2}$ para diferencias progresivas y $x = x_i - \frac{hx}{2}$, $\Delta x = \frac{hx}{2}$ para diferencias regresivas, quedando de la siguiente forma respectivamente:

$$U\left(x_i + \frac{hx}{2}\right) = U(x_i) + \frac{\partial U(x_i)}{\partial x}\frac{hx}{2} \tag{2.10}$$

$$U\left(x_i - \frac{hx}{2}\right) = U(x_i) - \frac{\partial U(x_i)}{\partial x}\frac{hx}{2} \tag{2.11}$$

despejando $U(x_i)$:

$$U(x_i) = U\left(x_i + \frac{hx}{2}\right) - \frac{\partial U(x_i)}{\partial x}\frac{hx}{2} \tag{2.12}$$

$$U(x_i) = U\left(x_i - \frac{hx}{2}\right) + \frac{\partial U(x_i)}{\partial x}\frac{hx}{2} \tag{2.13}$$

igualando (2.12) y (2.13) se tiene que :

$$U\left(x_i + \frac{hx}{2}\right) - \frac{\partial U(x_i)}{\partial x}\frac{hx}{2} = U\left(x_i - \frac{hx}{2}\right) + \frac{\partial U(x_i)}{\partial x}\frac{hx}{2} \tag{2.14}$$

desarrollando se tiene que:

$$2\frac{\partial U(x_i)}{\partial x}\frac{hx}{2} = U\left(x_i + \frac{hx}{2}\right) - U\left(x_i - \frac{hx}{2}\right)$$

$$\frac{\partial U(x_i)}{\partial x} = \frac{U\left(x_i + \frac{hx}{2}\right) - U\left(x_i - \frac{hx}{2}\right)}{hx} \tag{2.15}$$

se sustituye ahora (2.8), es decir se retorna del cambio de variable y se iguala su nuevo valor (2.15) en (2.9) y se tiene:

$$\frac{\partial}{\partial x}\left(\lambda(x_i)\frac{\partial f(x_i)}{\partial x}\right) = \frac{\lambda\left(x_i + \frac{hx}{2}\right)\frac{\partial f\left(x_i + \frac{hx}{2}\right)}{\partial x} - \lambda\left(x_i - \frac{hx}{2}\right)\frac{\partial\left(x_i - \frac{hx}{2}\right)}{\partial x}}{hx} \tag{2.16}$$

para simplificar el desarrollo y evitar confusiones (2.16) se reescribe como:

$$\frac{\partial}{\partial x}\left(\lambda(x_i)\frac{\partial f(xi)}{\partial x}\right) = \frac{\lambda_{xi+\frac{hx}{2}}\left(\frac{\partial f}{\partial x}\right)_{x_i+\frac{hx}{2}} - \lambda_{x_i-\frac{hx}{2}}\left(\frac{\partial f}{\partial x}\right)_{x_i-\frac{hx}{2}}}{hx} \tag{2.17}$$

retomando las diferencias finitas regresivas y progresivas (2.3) y (2.4) respectivamente y sustituyendo en (2.17) se llega a:

$$\frac{\partial}{\partial x}\left(\lambda(x_i)\frac{\partial f(x_i)}{\partial x}\right) = \frac{\lambda_{xi+\frac{hx}{2}}\left(\frac{f(x_i\ +\ h_x)\ -\ f(x_i)}{h_{x,e}}\right)_{x_i+\frac{hx}{2}} - \lambda_{x_i-\frac{hx}{2}}\left(\frac{f(x_i)\ -\ f(x_i\ -\ h_x)}{h_{x,w}}\right)_{x_i-\frac{hx}{2}}}{hx} \tag{2.18}$$

finalmente reescribiendo (2.18) :

$$\frac{\partial}{\partial x}\left(\lambda(x_i)\frac{\partial f(x_i)}{\partial x}\right) = \frac{1}{hx}\left[\frac{\lambda}{h_{x,e}}\bigg|_{x_i+\frac{hx}{2}}\left(f(x_i\ +\ h_x)\ -\ f(x_i)\right) - \frac{\lambda}{h_{x,w}}\bigg|_{x_i-\frac{hx}{2}} f(x_i) - f(x_i\ -\ h_x)\right] \tag{2.19}$$

2.2.4 Derivadas Temporales a partir del Esquema θ

Para poder discretizar la parte temporal de una ecuación, el esquema de diferencia más simple seria reemplazar las derivadas parciales en el espacio por una diferencia finita en el tiempo t^n y ∂F(S)/∂t por una diferencia hacia adelante. El esquema resultante es una diferencia central en el espacio y una diferencia hacia adelante en el tiempo, y se llama el esquema de Euler hacia adelante, la cual podría resolver la parte espacial de manera explícita para $f(x)^{n+1}$, sin embargo, este tipo de esquema conduce a problemas de estabilidad, por otra parte, si la derivada parcial espacial se sustituye por una derivada numérica en el tiempo t^{n+1}, a medida que se pasa de n a $n+1$, se define $f(x)^{n+1}$ implícitamente y se denomina esquema de Euler hacia atrás. En cada nivel de tiempo t^{n+1}, un sistema lineal de ecuaciones algebraicas debe ser resuelto. Entonces sí se considera:

$$\frac{\partial f(x,t)}{\partial t} = F(f(x,t)) \tag{2.20}$$

se pueden generalizar ambos esquemas en un esquema numérico tal que:

$$\frac{f^{n+1}-f^n}{\Delta t} = \theta F(f)^{n+1} + (1-\theta)F(f)^n \tag{2.21}$$

donde para $\theta = 0$, se tiene un esquema explicito $F(f)^n$, para $\theta = 1$, se tiene un esquema implícito $F(f)^{n+1}$, y para $\theta = 1/2$, se tiene el esquema conocido como de Crank-Nicolson $[F(f)^{n+1} + F(f)^n]\ 1/2$

2.3 Algoritmos para resolver Sistemas de Ecuaciones Lineales

Se ha visto que una aplicación de métodos de diferencias o elementos finitos a un problema estacionario o a un esquema implícito para un problema transitorio produce un sistema lineal de ecuaciones de la forma:

$$\boldsymbol{A}\,\overline{\boldsymbol{x}} = \bar{B} \tag{2.22}$$

donde $\boldsymbol{A}$ es una matriz $M \times M$. En general, la matriz $\boldsymbol{A}$ que surge en la simulación numérica de yacimientos es escasa, altamente asimétrica y mal condicionada. Su dimensión M a menudo oscila entre cientos y millones. Para la solución de sistemas de este último tamaño, los algoritmos del subespacio de Krylov son una buena opción. Se consideran los algoritmos basados en métodos iterativos y directos para resolver (2.22*)* para varios tipos de matriz $\boldsymbol{A}$.

Estos algoritmos se estudian primero para una matriz tridiagonal y luego se extienden a una matriz dispersa general. Los algoritmos iterativos CG (gradiente conjugado), GMRES (residuo mínimo generalizado) y BICGSTAB (gradiente biconjugado estabilizado) se analizan. La discusión de estos algoritmos, es de su aplicabilidad general. También se proporcionan algunas pautas sobre la elección de un algoritmo adecuado para un problema dado. Los algoritmos del subespacio de Krylov a menudo son inútiles sin un precondicionamiento. Por lo tanto, las versiones precondicionadas de estos algoritmos y la elección de precondicionadores también se estudian brevemente. Las consideraciones prácticas para la elección de precondicionadores en la simulación de yacimientos se analizan en este estudio también.

2.3.1 Métodos Directos e Iterativos

Los métodos clásicos de solución se suelen clasificar en directos o iterativos. Los métodos directos de resolución de sistemas lineales de ecuaciones son aquellos que permiten obtener la solución después de un número finito de operaciones aritméticas. Este número de operaciones es, obviamente, función del tamaño de la matriz. Si los ordenadores pudieran almacenar y operar con todas las cifras de los números reales, es decir, si emplearan una aritmética exacta, con los métodos directos se obtendría la solución exacta del sistema en un número finito de pasos. Puesto que los ordenadores tienen una precisión finita, los errores de redondeo se propagan y la solución numérica obtenida siempre difiere de la solución exacta. La cota del error, para una matriz y término independiente dados, se asocia por lo general al número de operaciones de cada método. Se pretende, por lo tanto, obtener métodos con el mínimo número de operaciones posible. Otra particularidad de los métodos directos es que siempre conducen, después de ciertas operaciones, a la resolución de uno o varios sistemas con solución inmediata. Es decir, sistemas donde la matriz es bandeada o triangular. Los métodos para sistemas de resolución inmediata son, de hecho, métodos directos.

Una clasificación habitual de estos procedimientos de resolución es aquella que considera que los mismos pertenecen a una de las dos categorías siguientes: métodos de eliminación y métodos de descomposición. Un método iterativo es un método que progresivamente va calculando aproximaciones a la solución de un problema. En Matemáticas, en un método iterativo se repite un mismo proceso de mejora sobre una solución aproximada: se espera que lo obtenido sea una solución más aproximada que la inicial. El proceso se repite sobre esta nueva solución hasta que el resultado más reciente satisfaga ciertos requisitos. En los métodos iterativos se puede suspender el proceso al término de una iteración y se obtiene una aproximación a la solución.

Un método iterativo consta de los siguientes pasos.

1 Inicia con una solución aproximada (Semilla).
2 Se ejecuta una serie de cálculos para obtener o construir una mejor aproximación partiendo de la aproximación semilla. La fórmula que permite construir la aproximación usando otra se conoce como ecuación de recurrencia.
3 Se repite el paso anterior, pero usando como semilla la aproximación obtenida.

En las siguientes subsecciones se abordan los métodos directos e iterativos de mayor uso en la simulación de yacimientos.

2.3.2 Algoritmo de la Descomposición $\boldsymbol{LU}$

La técnica de descomposición LU es una clase de método de eliminación, cuyo principal recurso es que el paso de la eliminación que toma mucho tiempo se puede formular de tal manera que involucre únicamente operaciones con la matriz de coeficientes $\boldsymbol{A}$ [13]. Por esto, es muy adecuado para aquellas situaciones donde se deben evaluar muchos vectores $\bar{B}$ del lado derecho para un solo valor de $\boldsymbol{A}$. Aunque hay muchas formas de hacer esto, no obstante, el análisis muestra cómo el método de eliminación de Gauss se implementa como una descomposición $\boldsymbol{LU}$. Del mismo modo, si se considera una matriz tridiagonal, se observa que siguiendo los mismos pasos podemos llegar al algoritmo de Thomas, que se describe con mayor detalle en la siguiente subsección. Si se considera al sistema (2.22) como una matriz tridiagonal, para este tipo de sistema, ningún algoritmo iterativo conocido puede competir con la eliminación directa, luego entonces una matriz de coeficientes $\boldsymbol{A}$ definida positiva, tiene una única factorización $\boldsymbol{LU}$ [14]. Los *métodos de descomposición* $\boldsymbol{LU}$ separan el tiempo usado en las eliminaciones para la matriz $\boldsymbol{A}$ de las manipulaciones en el lado derecho $\bar{B}$. Una vez que $\boldsymbol{A}$ se ha factorizado, los múltiples vectores del lado derecho $\bar{B}$ se pueden evaluar de manera eficiente. La matriz de coeficientes se descompone en los fatores $\boldsymbol{LU}$ como sigue:

$$\boldsymbol{A = LU} \tag{2.23}$$

donde $\boldsymbol{L} = \{l_{i,j}\}$ es una matriz $M \times M$ triangular inferior, es decir, $l_{i,j} = 0$, si $j > i$, y $\boldsymbol{U} = \{u_{i,j}\}$ es una matriz $M \times M$ triangular superior, es decir, $u_{i,j} = 0$, si $j < i$.

$$\boldsymbol{A} = \begin{pmatrix} AP & AE & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ AW & AP & AE & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & AW & AP & AE & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & AW & AP & \ddots & \cdots & \cdots & 0 & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \ddots & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & AW & AP & AE \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & AW & AP \end{pmatrix} \begin{pmatrix} p_i \\ p_i \\ p_i \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ p_n \end{pmatrix} = \begin{pmatrix} B_i \\ B_i \\ B_i \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ B_n \end{pmatrix}$$

Figura 2.4 Matriz Tridiagonal.

Para la matriz tridiagonal especial bajo consideración, se busca que las matrices $\boldsymbol{L}$ y $\boldsymbol{U}$ tengan la forma

$$\boldsymbol{L} = \begin{pmatrix} l_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ AW & l_1 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & AW & l_{i,j} & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & AW & l_{i,j} & \ddots & \cdots & \cdots & 0 & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \ddots & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & AW & l_{M-1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & AW & l_{i,j} \end{pmatrix}$$

Figura 2.5 Matriz Triangular Inferior ***L.***

$$\boldsymbol{U} = \begin{pmatrix} 1 & u_{i,j} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & u_{i,j} & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & 1 & u_{i,j} & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & 0 & 1 & \ddots & \cdots & \cdots & 0 & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \ddots & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & u_{M-1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Figura 2.6 Matriz Triangular Superior ***U.***

Tenga en cuenta que la diagonal inferior de $\boldsymbol{L}$ es la misma que la de $\boldsymbol{A}$, y en la diagonal principal de $\boldsymbol{U}$, todos sus elementos son unos. Ahora que se tiene la identidad (2.23) el sistema (2.22) se puede expresar en dos sistemas triangulares, superior e inferior, utilizando las matrices $\boldsymbol{U}$ y $\boldsymbol{L}$, respectivamente. Es decir,

$$\boldsymbol{U}\,\overline{\boldsymbol{x}} - \overline{D} = 0 \tag{2.24}$$

$$\boldsymbol{L}\,\overline{\boldsymbol{x}} - \bar{B} = 0 \tag{2.25}$$

tal que al efectuar las reglas de multiplicación entre matrices sé satisface que:

$$\boldsymbol{LU} = \boldsymbol{A} \tag{2.26}$$

$$\boldsymbol{L}\,\overline{D} = \bar{B} \tag{2.27}$$

Una estrategia de dos pasos para obtener soluciones se basa en las ecuaciones (2.24), (2.25), (2.26), y (2.27), (véase Figura 2.7) [13]:

1. *Paso de descomposición* $\boldsymbol{LU}$. $\boldsymbol{A}$ se factoriza o "descompone" en las matrices triangulares inferior $\boldsymbol{L}$ y superior $\boldsymbol{U}$
2. *Paso de la sustitución*. $\boldsymbol{L}$ y $\boldsymbol{U}$ se usan para determinar una solución $\overline{\boldsymbol{x}}$ para un lado derecho $\bar{B}$. Este paso, a su vez, se divide en dos. Primero, la ecuación (2.27) se usa para generar un vector intermedio $\overline{D}$ mediante sustitución hacia adelante. Después, el resultado se sustituye en la ecuación (2.24), la que se resuelve por sustitución hacia atrás para $\overline{\boldsymbol{x}}$**.**

Algoritmo para la Descomposición LU

Dados n y $a_{i,j}$,l_{kk}^{0}, o u_{kk}^{0}

for $k = 1,2,..n$:

$$l_{kk}u_{kk} = a_{i,j} - \sum_{1}^{k-1} l_{kk}u_{kk}$$

for $j = k+1, k+2, \ldots n$:

for $i = k+1, k+2, \ldots n$:

$$u_{kj} = \left(a_{i,j} - \sum_{1}^{k-1} l_{kk}u_{kk}\right)/l_{kk}$$

$$l_{ik} = \left(a_{i,j} - \sum_{1}^{k-1} l_{kk}u_{kk}\right)/u_{kk}$$

Figura 2.7 Algoritmo de Descomposición LU, tomada de [42].

2.3.3 Algoritmo de Thomas

El Algoritmo de Thomas es un método directo muy utilizado, particularmente para problemas de flujo monofásico unidimensional, la matriz $\boldsymbol{A}$ es tridiagonal, Figura 2.4.

El sistema tridiagonal de este tipo puede resolverse mediante un algoritmo de eliminación directa o mediante un algoritmo iterativo. Para este tipo de sistema, es muy común utilizar el algoritmo de Thomas. Dicha matriz generalmente se obtiene de la discretización por diferencias finitas y que da origen a la ecuación general:

$$AP_i x_i = AE_i \overline{\boldsymbol{x}}_{i+1} + AW_i \overline{\boldsymbol{x}}_{i-1} + \bar{B}_i \quad i = 1,2,3, \ldots, n \tag{2.28}$$

en un Algoritmo de Thomas ó TDMA, se supone primero una sustitución hacia adelante,

$$\overline{\boldsymbol{x}}_i = P_i \overline{\boldsymbol{x}}_{i+1} + Q_i \tag{2.29}$$

dónde P_i y Q_i se conocen como relaciones de recurrencia, evaluando la ecuación (2.29) para $i = i - 1$:

$$\overline{\boldsymbol{x}}_{i-1} = P_{i-1} \overline{\boldsymbol{x}}_i + Q_{i-1} \tag{2.30}$$

sustituyendo la ecuación (2.30) en la ecuación (2.28), se obtiene:

$$AP_i x_i = AE_i x_{i+1} + AW_i[P_{i-1}x_i + Q_{i-1}] + \bar{B}_i \tag{2.31}$$

despejando a x_i:

$$\bar{\boldsymbol{x}}_i = \frac{AE_i}{[AP_i - AW_i P_{i-1}]} x_{i+1} + \frac{AW_i Q_{i-1} + \bar{B}_i}{[AP_i - AW_i P_{i-1}]} \tag{2.32}$$

igualando las ecuaciones (2.32) y (2.29), después de algunos pasos algebraicos se obtienen las relaciones de recurrencia

$$P_i = \frac{AE_i}{[AP_i - AW_i P_{i-1}]} \tag{2.33}$$

$$Q_i = \frac{AW_i Q_{i-1} + \bar{B}_i}{[AP_i - AW_i P_{i-1}]} \tag{2.34}$$

los valores de P_i y Q_i, $(i = 1,\ AW_i = 0)$ son dados por:

$$P_1 = \frac{AE_1}{AP_1} \tag{2.35}$$

$$Q_1 = \frac{\bar{B}_1}{AP_1} \tag{2.36}$$

de igual manera los valores para las relaciones de recurrencia en $i = n$ $(i = n,\ AE_n = 0)$:

$$P_n = 0 \tag{2.37}$$

$$Q_n = \bar{\boldsymbol{x}}_n \tag{2.38}$$

y finalmente, para calcular el valor de las incógnitas $\bar{\boldsymbol{x}}$ se hace una sustitución regresiva, comenzando por calcular el valor de $\bar{\boldsymbol{x}}_n$ y después se utiliza la ecuación.

Algoritmo de Thomas

Dados AP_0, AE_0, AW_0 y $\bar{B}_0$ establecer:

$$P_0 = \frac{AE_0}{AP_0}$$

$$Q_0 = \frac{\bar{B}_0}{AP_0}$$

for $i = 1, 2, \ldots, n$ **:**

$$P_i = \frac{AE_i}{[AP_i - AW_i P_{i-1}]}$$

$$Q_i = \frac{AW_i Q_{i-1} + \bar{B}_i}{[AP_i - AW_i P_{i-1}]}$$

$$Q_{n-1} = \bar{x}_{n-1}$$

$$i = n - 2$$

while $i \neq -1$ **:**

$$\bar{x}_{i-1} = P_{i-1}\bar{x}_i + Q_{i-1}$$

Figura 2.8 Algoritmo de Thomas 1D.

2.3.4 Gradiente Conjugado (CG)

En matemáticas, el método del gradiente conjugado es un algoritmo para resolver numéricamente los sistemas de ecuaciones lineales cuyas matrices son simétricas y definidas positivas. Es un método iterativo, así que se puede aplicar a los sistemas dispersos que son demasiado grandes para ser tratados por métodos directos como la descomposición de Cholesky. Tales sistemas surgen frecuentemente cuando se resuelve numéricamente las ecuaciones en derivadas parciales.

El método del Gradiente Conjugado se puede utilizar también para resolver los problemas de optimización sin restricciones como la minimización de la energía.

El método del Gradiente Biconjugado proporciona una generalización para matrices no simétricas. Varios métodos del Gradiente Conjugado no lineales buscan los mínimos de las ecuaciones no lineales. Se debe tener en cuenta que el número de condición de la matriz A está definido por:

$$cond(A) = \|A\|\|A^{-1}\| \tag{2.39}$$

donde $\|A\|$ es la norma matricial de $\boldsymbol{A}$ inducida por una norma $\|\cdot\|$ sobre $\mathbb{R}^M$ (por ejemplo, la $l^2 -$ norma$\|\cdot\|_2$ en $\mathbb{R}^M$: $\|\mathbf{v}\|_2 = (\sum_{i=1}^{M}|v_i|^2)^{1/2}$, $\mathbf{v} = (v_1, v_2, \ldots, v_M)$.Aquí se entiende que $cond(\boldsymbol{A})$ es infinito si $\boldsymbol{A}$ es singular. La matriz $\boldsymbol{A}$ en el sistema (2.22) que surge de la discretización de elementos finitos estándar de un problema elíptico de segundo orden, por ejemplo, tiene un número de condición proporcional a $h - 2$ [15, 16] cuando $h \rightarrow 0$, donde h es el tamaño de malla espacial. La práctica habitual para obtener la solución de un sistema a gran escala es utilizar un algoritmo iterativo.

Está más allá del alcance de esta tesis proporcionar incluso una breve introducción a todos los algoritmos iterativos disponibles para la solución del sistema (2.22). Durante el desarrollo de la simulación de yacimientos se analizaron algunos algoritmos iterativos simples, como el punto fijo y el bloque de Jacobi, Gauss-Seidel y los algoritmos de sobre relajación sucesiva (SOR) aplicados a la simulación de yacimientos en el entorno de diferencias finitas [17, 18]. En este capítulo, se revisan los algoritmos del subespacio de Krylov para sistemas lineales. Los dos algoritmos de este tipo que se estudian con poco detalle son los algoritmos CG y GMRES. El algoritmo CG fue introducido como un algoritmo directo [19]. Ha tenido un amplio uso como algoritmo iterativo y, en general, ha reemplazado a los algoritmos iterativos de Jacobi, Gauss-Seidel y SOR. A diferencia de los algoritmos iterativos estacionarios, los algoritmos del subespacio de Krylov no tienen una matriz de iteraciones. Minimizan, en la iteración k-ésima, alguna medida de errores sobre el espacio afín donde $\overline{\boldsymbol{x}}^0$ es una estimación inicial de (2.22) y $\mathcal{K}^k$ está definido por

$$\overline{\boldsymbol{x}}^0 + \mathcal{K}^k \tag{2.40}$$

donde $\bar{\boldsymbol{x}}^0$ es una estimación inicial de (2.22) y $\mathcal{K}^k$ está definido por

$$\mathcal{K}^k = \left(\mathbf{r}^0, \mathbf{A}\mathbf{r}^0, \ldots, \mathbf{A}^{\boldsymbol{k-1}}\mathbf{r}^0\right), \quad k \geq 1 \tag{2.41}$$

el $\mathbf{r}^k$ residual para la k-ésima iteración de $\bar{\boldsymbol{x}}^k$ es

$$\boldsymbol{r}^k = \bar{B} - \boldsymbol{A}\bar{\boldsymbol{x}}^k, \quad k \geq 0 \tag{2.42}$$

sí $\boldsymbol{A}$ es definida positiva simétrica, se deduce un producto escalar $\langle \cdot, \cdot \rangle$ sobre$\mathbb{R}^M$:

$$\langle \boldsymbol{v}, \boldsymbol{w} \rangle = \boldsymbol{v}^T \boldsymbol{A}\boldsymbol{w} = \sum_{i,j=1}^{\boldsymbol{M}} v_i a_{ij} w_j \qquad \boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^M \tag{2.43}$$

la norma $\|\cdot\|_A$ correspondiente a $\langle \cdot, \cdot \rangle$ es la norma energética

$$\|\cdot\|_A = \langle \mathbf{v}, \mathbf{w} \rangle^{1/2} \tag{2.44}$$

sobre $\bar{\boldsymbol{x}}^0 + \mathcal{K}^k$. Note que si $\boldsymbol{f}(\boldsymbol{x}^*)$ es el valor mínimo en $\mathbb{R}^M$, entonces

$$\boldsymbol{\nabla f}(\bar{\boldsymbol{x}}^*) = \boldsymbol{A}\bar{\boldsymbol{x}}^* - \bar{\boldsymbol{B}} = \mathbf{0} \tag{2.45}$$

es decir, $\bar{\boldsymbol{x}}^*$ es la solución.

Dado $\bar{\boldsymbol{x}}^0$, CG busca aproximaciones sucesivas $\bar{\boldsymbol{x}}^k$ de la forma:

$$\bar{\boldsymbol{x}}^k = \bar{\boldsymbol{x}}^{k-1} + \alpha_{k-1}\boldsymbol{d}^{\boldsymbol{k-1}} \qquad \boldsymbol{k} = \mathbf{1}, \mathbf{2}, \ldots, \tag{2.46}$$

donde d^{k-1} es una dirección de búsqueda y α_{k-1} es una longitud de paso. Una vez que se encuentra $\boldsymbol{d}^{\boldsymbol{k-1}}$, α_{k-1} es fácil de calcular a partir de la propiedad de minimización de la iteración:

$$\left.\frac{d\boldsymbol{f}\left(\bar{\boldsymbol{x}}^{k-1} + \alpha_{k-1}\boldsymbol{d}^{\boldsymbol{k-1}}\right)}{d\alpha}\right|_{\alpha=\alpha_{k-1}} = 0 \tag{2.47}$$

se supone que las direcciones de búsqueda $\boldsymbol{d}^{\boldsymbol{k-1}}$ satisfacen la condición de conjugación $\boldsymbol{A}$

$$\left(\boldsymbol{d}^{\boldsymbol{k_1}}\right)^{\boldsymbol{T}} \boldsymbol{A}\boldsymbol{d}^{\boldsymbol{k_2}} = \mathbf{0} \qquad \boldsymbol{si}\ \boldsymbol{k_1} \neq \boldsymbol{k_2} \tag{2.48}$$

La implementación habitual de CG refleja la propiedad de minimización y la condición de conjugación $\boldsymbol{A}$. La entrada para el algoritmo CG es la iteración inicial $\bar{\boldsymbol{x}}^0$, que se puede sobrescribir con la solución, el lado derecho $\bar{B}$ y la matriz de coeficientes $\boldsymbol{A}$ (o una rutina que calcula la acción de A en un vector). Entonces este algoritmo para la solución de (2.22) se puede definir como en la Figura 2.9. La propia matriz $\boldsymbol{A}$ no necesita formarse ni almacenarse; solo se requiere una rutina para productos matriz-vector.

Por esta razón, los algoritmos del espacio de Krylov suelen denominarse algoritmos sin matriz. Se puede comprobar que el algoritmo CG da, en ausencia de errores de redondeo, la solución exacta después de un máximo de *M* pasos; es decir.,

$$\boldsymbol{A\bar{x}^k = \bar{B}}, \qquad \boldsymbol{k \leq 1} \tag{2.49}$$

en la práctica, el número requerido de iteraciones es a veces menor que M. De hecho, para una tolerancia dada $\boldsymbol{\epsilon} > 0$, para satisfacer:

$$\|\boldsymbol{x - x^k}\|_A \leq \epsilon \|\boldsymbol{x - x^0}\|_A \tag{2.50}$$

basta elegir *k* tal que [20]:

$$k \geq \frac{1}{2}\sqrt{cond(\boldsymbol{A})} ln\left(\frac{2}{\epsilon}\right) \tag{2.51}$$

por lo tanto, el número requerido de iteraciones para el algoritmo CG es proporcional a $\sqrt{cond(A)}$ Como es de esperarse en en una aplicación típica de elementos finitos a un problema estacionario. El problema es que los sistemas de ecuaciones algebraicas que surgen de la discretización de las ecuaciones gobernantes en la simulación de yacimientos tienen propiedades especiales.

Algoritmo CG

Dada x^0 se establece $\mathbf{r}^0 = \bar{B} - \mathbf{A}\bar{\boldsymbol{x}}^0$ y $\boldsymbol{d^0} = \boldsymbol{r}^0$

for $k = 1,2,\ldots$, determinar $\boldsymbol{x}^k$ y $\boldsymbol{d^k}$

$$\alpha_{k-1} = \frac{(\boldsymbol{r}^{k-1})^T \boldsymbol{r}^{k-1}}{\langle\, \boldsymbol{d^{k-1}}, \boldsymbol{d^{k-1}} \,\rangle}$$

$$\bar{\boldsymbol{x}}^k = \bar{\boldsymbol{x}}^{k-1} + \alpha_{k-1}\boldsymbol{d^{k-1}}$$

$$\boldsymbol{r}^k = \boldsymbol{r}^{k-1} - \alpha_{k-1}\boldsymbol{d^{k-1}}$$

$$\beta_{k-1} = \frac{(\boldsymbol{r}^k)^T \boldsymbol{r}^k}{(\boldsymbol{r}^{k-1})^T \boldsymbol{r}^{k-1}}$$

$$\boldsymbol{d^k} = \boldsymbol{r}^k + \beta_{k-1}\boldsymbol{d^{k-1}}$$

Figura 2.9 Algoritmo CG, adaptada de [2].

Las matrices de coeficientes (rigidez) de estos sistemas son escasas, pero no simétricas e indefinidas. Si bien es escasa, su estructura natural en bandas generalmente se ve afectada por pozos que perforan muchos bloques de malla y/o por una estructura de bloque de malla irregular. Para tales sistemas, el algoritmo CG puede sufrir un deterioro severo en el rendimiento.

2.3.5 Algoritmo BICGSTAB

Los Algoritmos CG y GMRES son algoritmos del subespacio de Krylov que se basan en alguna forma de ortogonalización de los vectores de Krylov para obtener una solución aproximada. Esta sección considera una familia de algoritmos del subespacio de Krylov que, en cambio, están definidos por un enfoque de biortogonalización [21].

Estos algoritmos son métodos de proyección que son intrínsecamente no ortogonales. Tienen algunas propiedades atractivas, pero son más difíciles de analizar teóricamente. El método más antiguo de este tipo es el algoritmo BCG (gradiente biconjugado) [21]. BCG no aplica un principio de minimización; en cambio, el residuo k-ésimo debe satisfacer la condición de biortogonalidad

$$\left(\mathbf{r}^{\boldsymbol{k}}\right)^{\boldsymbol{T}}\mathbf{v} = \mathbf{0} \quad \forall \mathbf{v} \in \widehat{\mathcal{K}}^{k} \tag{2.52}$$

donde el espacio de Krylov $\widehat{\mathcal{K}}^{k}$ de $\boldsymbol{A}^{T}$ está definido por:

$$\widehat{\mathcal{K}}^{k} = \left(\mathbf{r}^{\mathbf{0}}, \boldsymbol{A}^{T}\mathbf{r}^{\mathbf{0}}, \ \ldots, (\boldsymbol{A}^{T})^{k-1}\mathbf{r}^{\mathbf{0}}\right) \tag{2.53}$$

Un problema con BCG es que se necesita un producto de vector transpuesto, que en el mejor de los casos requerirá programación adicional y, en el peor de los casos, puede no ser factible. Un remedio para este problema es el algoritmo CGS (gradiente conjugado al cuadrado) [22]. CGS reemplaza el producto de vector de transposición con un producto de vector de matriz adicional y se basa en elevar al cuadrado el polinomio residual. Un problema con este enfoque es que pueden acumularse errores de redondeo sustanciales. BICGSTAB [23] fue desarrollado para superar esta dificultad y suavizar la convergencia de CGS; Figura 2.10. No existe una teoría de convergencia para BICGSTAB. La iteración puede fallar en los pasos que calculan los coeficientes α_{k-1} y β_{k-1}. El costo en almacenamiento y en operaciones de punto flotante por iteración permanece acotado en toda la iteración. Una sola iteración requiere cuatro productos escalares. En el caso de que se necesiten muchas iteraciones de GMRES y un vector de matriz producto es rápido, BICGSTAB puede tener un costo promedio por iteración mucho más bajo que GMRES. La razón es que el costo de la ortogonalización en el último algoritmo puede ser mucho mayor que el del producto matriz-vector en BICGSTAB si la dimensión del espacio de Krylov es grande.

Algoritmo BiCGSTAB

Dada x^0 se establece $\mathbf{r}^0 = \bar{B} - \mathbf{A}\boldsymbol{x}^0$ y $\boldsymbol{d^0} = \boldsymbol{r}^0$; $\hat{\boldsymbol{r}}^0$ aleatorio

for $k = 1,2,\ldots$, determinar $\overline{\boldsymbol{x}}^k$ y $\boldsymbol{d^k}$

$$\alpha_{k-1} = \frac{(\boldsymbol{r}^k)^T \hat{\boldsymbol{r}}^0}{(\boldsymbol{A}\boldsymbol{d^{k-1}})^T \hat{\boldsymbol{r}}^0}$$

$$\overline{\boldsymbol{x}}_2^{k-1} = \overline{\boldsymbol{x}}^{k-1} + \alpha_{k-1}\boldsymbol{A}\boldsymbol{d^{k-1}}$$

$$\omega_{k-1} = \frac{\left(\overline{\boldsymbol{x}}_2^{k-1}\right)^T \overline{\boldsymbol{x}}_2^{k-1}}{\left(\boldsymbol{A}\overline{\boldsymbol{x}}_2^{k-1}\right)^T \boldsymbol{A}\overline{\boldsymbol{x}}_2^{k-1}}$$

$$\overline{\boldsymbol{x}}^k = \overline{\boldsymbol{x}}^{k-1} + \alpha_{k-1}\boldsymbol{d^{k-1}} + \omega_{k-1}\overline{\boldsymbol{x}}_2^{k-1}$$

$$\boldsymbol{r}^k = \overline{\boldsymbol{x}}_2^{k-1} - \omega_{k-1}\overline{\boldsymbol{x}}_2^{k-1}$$

$$\beta_{k-1} = \frac{(\boldsymbol{r}^k)^T \hat{\boldsymbol{r}}^0}{(\boldsymbol{r}^{k-1})^T \hat{\boldsymbol{r}}^0} \frac{\alpha_{k-1}}{\omega_{k-1}}$$

$$\boldsymbol{d^k} = \boldsymbol{r}^k + \beta_{k-1}\left(\boldsymbol{d^{k-1}} - \omega_{k-1}\boldsymbol{A}\boldsymbol{d^{k-1}}\right)$$

Figura 2.10 Algoritmo BiCGSTAB, adaptada de [2].

2.3.6 Algoritmo GMRES

Se sabe que el algoritmo GMRES es un algoritmo muy eficiente para resolver sistemas no simétricos dispersos generales [24, 25].

La k-ésima iteración de GMRES es la solución al problema de los mínimos cuadrados

$$\min_{\boldsymbol{x}\in\boldsymbol{x}^0+\mathcal{K}^k} \|\bar{B}-\boldsymbol{A}\bar{\boldsymbol{x}}\|_2 \tag{2.54}$$

suponga que uno tiene una proyección ortogonal V^k sobre $\mathcal{K}^k$. Entonces cualquier z $\in \mathcal{K}^k$ puede ser representado por:

$$z=\sum_{i=1}^{k} q_i\mathbf{v}^{\boldsymbol{i}} \tag{2.55}$$

para algunos $\mathbf{q} = (q_1, q_2,\ldots,q_k)^T \in \mathbb{R}^k$, donde $\mathbf{v}^{\boldsymbol{i}}$ es la i-ésima columna de $\mathbf{V}^k$. Colocar

$$\bar{\boldsymbol{x}}-\bar{\boldsymbol{x}}^{\mathbf{0}}=\mathbf{V}^k\mathbf{q} \tag{2.56}$$

para algún $\mathbf{q} \in \mathbb{R}^k$, ya que:

$$\bar{B}-\boldsymbol{A}\bar{\boldsymbol{x}}=\bar{B}-\boldsymbol{A}\bar{\boldsymbol{x}}^{\mathbf{0}}-\boldsymbol{A}\mathbf{V}^k\mathbf{q}=\mathbf{r}^{\mathbf{0}}-\boldsymbol{A}\mathbf{V}^k\mathbf{q} \tag{2.57}$$

el problema (2.54) se puede convertir al problema de mínimos cuadrados

$$\min_{\boldsymbol{q}\in\mathbb{R}^k}\left\|\boldsymbol{r}^{\mathbf{0}}-\boldsymbol{A}\boldsymbol{V}^k\boldsymbol{q}\right\|_2 \tag{2.58}$$

Este es un problema estándar de mínimos cuadrados que se puede resolver mediante la factorización *QR*, por ejemplo. El problema con un método tan directo es que el producto vectorial matricial de $\boldsymbol{A}$ con $\mathbf{V}^k$ debe realizarse en cada iteración. Si se aplica la técnica de ortogonalización de Gram-Schmidt a (2.58) , el problema de mínimos cuadrados resultante no requiere ningún producto adicional de $\boldsymbol{A}$ con vectores. La técnica para construir una base ortonormal para $\mathcal{K}^k$ se denomina algoritmo de Arnoldi [26], Figura 2.11.Los datos de entrada para este algoritmo son $\bar{\boldsymbol{x}}^{\mathbf{0}}$, $\bar{\boldsymbol{B}}$, $\boldsymbol{A}$ y una dimensión k. Si el algoritmo de Arnoldi no se detiene antes del k-ésimo paso, los vectores $\mathbf{v}^{\mathbf{1}},\mathbf{v}^{\mathbf{2}},\ldots,\mathbf{v}^{\boldsymbol{k}}$ forman una base ortonormal para $\mathcal{K}^k$. Denotar por $\mathbf{V}^k$ la matriz $M \times k$ con estos vectores columna, y por H^k la matriz superior de Hessenberg $(k+1)\times k$ cuyas entradas distintas de cero h_{ij} son calculadas por el algoritmo de Arnoldi. Este algoritmo (a menos que termine prematuramente con una solución) genera la relación:

$$\boldsymbol{A}\boldsymbol{V}^k=\boldsymbol{V}^{k+1}+H^k \tag{2.59}$$

sea $e_1 = (1, 0, \ldots, 0)^T \in \mathbb{R}^{k+1}$ y $\beta = \|\mathbf{r^0}\|_2$. Para la k-ésima iteración $\bar{\boldsymbol{x}}^{\boldsymbol{k}}$ de GMRES, defina

$$\bar{\boldsymbol{x}}^{\boldsymbol{k}} = \bar{\boldsymbol{x}}^{\boldsymbol{0}} - \mathbf{V}^k \mathbf{q}^{\boldsymbol{k}} \tag{2.60}$$

para algún $q^k \in \mathbb{R}^k$. Entonces se sigue de (2.59) y (2.60) que

$$\mathbf{r}^{\boldsymbol{k}} = \boldsymbol{B} - \boldsymbol{A}\bar{\boldsymbol{x}}^{\boldsymbol{k}} = \boldsymbol{A}\left(\bar{\boldsymbol{x}}^{\boldsymbol{k}} - \bar{\boldsymbol{x}}^{\boldsymbol{0}}\right) = \mathbf{V}^{k+1}(\beta \boldsymbol{e}_1 - \boldsymbol{H}^k \boldsymbol{q}^k) \tag{2.61}$$

usando la ortogonalidad de $\mathbf{V}^{k+1}$

$$\|\mathbf{r}^k\|_2 = \|\mathbf{V}^{k+1}(\beta \boldsymbol{e}_1 - \boldsymbol{H}^k \boldsymbol{q}^k)\|_2 = \|\beta \boldsymbol{e}_1 - \boldsymbol{H}^k \boldsymbol{q}^k\|_2 \tag{2.62}$$

es decir, $\boldsymbol{q}^k$ minimiza $\|\beta \boldsymbol{e}_1 - \boldsymbol{H}^k \boldsymbol{q}^k\|_2$. El minimizador $\boldsymbol{q}^k$ es económico de obtener porque requiere la solución de un problema de mínimos cuadrados $(k + 1) \times k$ cuando k es pequeño.

Los datos de entrada para GMRES son $\boldsymbol{x^0}$,$\bar{B}$ y $\boldsymbol{A}$ (o una rutina que calcula la acción de $\boldsymbol{A}$ en un vector); Figura 2.12

En cuanto a CG, si $\boldsymbol{A}$ no es singular, el algoritmo GMRES encontrará, en ausencia de errores de redondeo, la solución dentro de M iteraciones. Para obtener información más precisa sobre las tasas de convergencia, considere el caso donde $\boldsymbol{A}$ es diagonalizable.

Dada x^0 se establece $\mathbf{r}^0 = \bar{B} - \mathbf{A}\bar{\boldsymbol{x}}^0$ y $v^1 = \mathbf{r}^0 / \|\mathbf{r}^0\|_2$

$for\ j = 1,2, \ldots, k$

$for\ i = 1,2, \ldots, j$

$h_{i,j} = \left(v^i\right)^T \boldsymbol{A} v^j$

$\boldsymbol{w^j} = \boldsymbol{A} v^j - \sum_{\boldsymbol{i,j=1}}^{\boldsymbol{M}} h_{ij} v^i$

$h_{j+1,j} = \|\boldsymbol{w^j}\|_2$

$if\ \ h_{j+1,j} = 0$:

break;

$v^{j+1} = \boldsymbol{w^j} / h_{j+1,j}$

Figura 2.11 Algoritmo de Arnoldi, adaptada de [2].

Recuerde que $\boldsymbol{A}$ es diagonalizable si existe una matriz no singular $\boldsymbol{E}$ tal que

$$\mathbf{A} = \mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^{-1} \tag{2.63}$$

donde $\boldsymbol{\Lambda}$ es una matriz diagonal con los valores propios de $\boldsymbol{A}$ en su diagonal. En este caso, la k-ésima GMRES iteración $\bar{\boldsymbol{x}}^k$ satisface [27]

$$\frac{\|\boldsymbol{r}^k\|_2}{\|\boldsymbol{r}^{\boldsymbol{0}}\|_2} \leq cond(E) \quad \inf_{x_k \in X_k, x_k(0)=1} \left\{ \max_{z \in \sigma(\boldsymbol{A})} |x_k(z)| \right\} \tag{2.64}$$

donde $cond(\boldsymbol{E})$ es el número de condición de $\boldsymbol{E}$, X_k es el conjunto de polinomios de grado máximo k, y $\sigma(\boldsymbol{A})$ es el conjunto de valores propios de $\boldsymbol{A}$ (el espectro de $\boldsymbol{A}$). No está claro cómo estimar $cond(\boldsymbol{E})$. Si $\boldsymbol{A}$ es normal, por supuesto, $cond(\boldsymbol{E}) = 1$.

En el algoritmo GMRES, X_k se evalúa solo al finalizar y no se requiere dentro de la iteración. Es importante que la base para el espacio de Krylov se almacene a medida que avanza la iteración. Esto implica que para realizar k iteraciones GMRES, se deben almacenar k vectores de longitud M y que GMRES se vuelve poco práctico cuando k es grande debido a los requisitos de memoria de la computadora. Hay dos remedios. El primero es "truncar" la ortogonalización en el algoritmo de Arnoldi; es decir, se selecciona y fija un entero k, y se realiza una ortogonalización "incompleta", llamada también algoritmo ORTHOMIN. El segundo remedio es reiniciar la iteración después de cada k pasos para algún número entero k, con p^k utilizado como suposición inicial en la siguiente iteración. Esta versión reiniciada del algoritmo se denomina GMRES (k) [25]. No existe una teoría de convergencia general para GMRES reiniciado; sin embargo, para una matriz definida positiva $\boldsymbol{A}$, GMRES(k) converge para cualquier $k \geq 1$. Reiniciar ralentizará la convergencia; cuando funciona, sin embargo, reducirá significativamente el almacenamiento.

Algoritmo GMRES

Dado $\boldsymbol{x^0} \in \mathbb{R}^M$, estableciendo $\mathbf{r}^0 = \bar{\bar{B}} - \mathbf{A}\bar{\boldsymbol{x}}^0$, $\beta = \left\|\mathbf{r^0}\right\|_2$ y $\mathbf{v^1} = \mathbf{r^0}/\beta$

for $(k+1) \times k - \acute{e}sima$ matriz $\boldsymbol{H}^k = \left(h_{ij}\right)$, estableciendo que $\boldsymbol{H}^k = 0$

for $j = 1,2,\ldots,k$,

for $i = 1,2,\ldots,j$: *hacer:*

$w^j = A\mathrm{v}^j$

$h_{ij} = \left(\mathrm{v}^i\right)^T w^j$

$h_{j+1,j} = \left\|w^j\right\|_2$

if $h_{j+1,j} = 0$,

$k = j$ y salta al siguiente paso;

$\mathrm{v}^{j+1} = w^j / h_{j+1,j}$

Determinar el minimizador $\boldsymbol{q}^k$ de $\|\beta \boldsymbol{e}_1 - \boldsymbol{H}^k \boldsymbol{q}^k\|_2$

Establece: $\bar{\boldsymbol{x}}^{\boldsymbol{k}} = \bar{\boldsymbol{x}}^{\boldsymbol{0}} - \mathbf{V}^k \mathbf{q}^{\boldsymbol{k}}$

Figura 2.12 Algoritmo GMRES, adaptada de [2].

2.3.7 Optimización de Algoritmos mediante la Implementación de Matrices Dispersas CSR

La idea original de aprovechar los ceros de una matriz y su ubicación fue iniciada por ingenieros en diversas disciplinas. En el caso más simple que involucra matrices diagonales, las técnicas especiales son factibles de llevar a cabo. El principal problema, y el primero en ser discutido por las ciencias de matriz dispersa, fue diseñar métodos de solución directa para sistemas lineales. Estos tenían que ser económicos, tanto en términos de trabajo computacional como en esfuerzo de cálculo. Los solucionadores directos para matrices escasas pueden administrar problemas muy grandes que no pueden ser abordados por los solucionadores de matrices densas habituales. Esencialmente, hay dos tipos principales de matrices dispersas: estructuradas y no estructuradas. Una matriz estructurada es aquella cuyas entradas distintas de cero forman un patrón regular, usualmente suelen ser simétricas con un determinado número de diagonales [27]. Alternativamente, los elementos distintos de cero se pueden ubicar en bloques (densos) del mismo tamaño, que forman un patrón regular. Una matriz con entradas ubicadas irregularmente se estructuraría de manera irregular. La distinción entre los dos tipos de matrices puede no afectar significativamente las técnicas de solución directa, sin embargo, esta distinción puede ser importante para los métodos de solución iterativa [27]. Dentro de estos métodos, una de las operaciones esenciales son los productos matriciales y Vectoriales. El rendimiento de estas operaciones puede diferir considerablemente en las computadoras de alto rendimiento, dependiendo de si están estructuradas regularmente o no. Por ejemplo, en las computadoras que trabajan sobre vectores (arrays lineales de números), el almacenamiento de la matriz por diagonal es ideal, pero los diagramas más generales pueden sufrir porque requieren una dirección indirecta.

Para aprovechar la gran cantidad de elementos cero, se requieren esquemas especiales para almacenar matrices dispersas. El objetivo principal es representar solo los elementos distintos de cero y poder realizar las operaciones de matriz comunes. El esquema de almacenamiento más simple para matrices dispersas es el llamado formato de coordenadas.

La estructura de datos consta de tres matrices:

1. Una matriz real que contiene todos los valores reales (o complejos) de los elementos distintos de cero de $\boldsymbol{A}$ en cualquier orden.
2. Una matriz de enteros que contiene sus índices de fila.
3. Una segunda matriz de enteros que contiene sus índices de columna.

Las tres matrices tienen una longitud igual a el número de elementos distintos de cero. Por ejemplo, en la Figura 2.13 la matriz $\boldsymbol{A} = \{a_{i,j}\}$ es una matriz bandeada, a saber, penta diagonal y cuyas coordenadas de interés únicamente serán los elementos distintos de cero (escritos en rojo) los cuales se pueden enumerar en un orden arbitrario, generalmente se enumeran por filas o columnas.

$$\boldsymbol{A} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{pmatrix} AP & AE & 0 & 0 & 0 & AN \\ AW & AP & AE & 0 & 0 & 0 \\ 0 & AW & AP & AE & 0 & 0 \\ 0 & 0 & AW & AP & AE & 0 \\ 0 & 0 & 0 & AW & AP & AE \\ AS & 0 & 0 & 0 & AW & AP \end{pmatrix} \end{array}$$

$j \setminus i$	0	1	2	3	4	5
0	(0,0)	(1,0)	(2,0)	(3,0)	(4,0)	(5,0)
1	(0,1)	(1,1)	(2,1)	(3,1)	(4,1)	(5,1)
2	(0,2)	(1,2)	(2,2)	(3,2)	(4,2)	(5,2)
3	(0,3)	(1,3)	(2,3)	(3,3)	(4,3)	(5,3)
4	(0,4)	(1,4)	(2,4)	(3,4)	(4,4)	(5,4)
5	(0,5)	(1,5)	(2,5)	(3,5)	(4,5)	(5,5)

Figura 2.13 Esquema de Matriz Dispersa.

Si los elementos se enumeraran por fila, supondría un ahorro nada desdeñable en almacenamiento. La nueva estructura de datos tiene tres matrices con las siguientes funciones:

1. Una matriz real V contiene los valores reales $\{a_{i,j}\}$ almacenados fila por fila, desde la fila 0 hasta la n. La longitud de V es de $N_z = i \times j$.
2. Una matriz de enteros JV contiene los índices de columna de los elementos $\{a_{i,j}\}$ almacenados en la matriz V.
3. Una matriz de enteros IV contiene los punteros al comienzo de cada fila en las matrices V y JV. Por lo tanto, el contenido de $IV(i)$ es la posición en los arreglos V y JV donde comienza la i-ésima fila. La longitud de IV es $n + 1$ con $IV(n + 1)$ que contiene el número $IV(1) + N_z$, es decir, la dirección en V y JV del comienzo de una fila ficticia número $n + 1$.

Por lo tanto, la matriz anterior se puede almacenar como lo indica la Figura 2.14. Este formato es probablemente el más popular para almacenar matrices dispersas generales. Se llama el formato Compressed Sparse Row (CSR) por sus siglas en inglés. Se prefiere este esquema al esquema de coordenadas porque suele optimizar el ahorro de memoria en la mayoría de los cálculos. Por otro lado, el esquema de coordenadas es ventajoso por su simplicidad y su flexibilidad.

$V =$

AP	AE	AN	AW	AP	AE	AW	AP	AE	AW	AP	AE	AW	AP	AE	AS	AW	AP

$JV =$

0	0	0	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5

$IV =$

AP	AW	AW	AW	AW	AS	AS

Figura 2.14 Almacenamiento de Matrices en formato CSR, adaptado de [27].

CAPÍTULO III

SIMULACIÓN NUMÉRICA DE FLUJO BIFÁSICO INCOMPRESIBLE EN MEDIOS POROSOS

3. Modelado Numérico de Flujo Bifásico.

En este capítulo se estudian los modelos matemáticos, las técnicas numéricas y algoritmos computacionales para llevar a cabo las simulaciones numéricas para flujo bifásico incompresible en medios porosos, tomando como referencia el modelo simétrico de los cinco pozos reportados por Chen. El modelo matemático está basado en la formulación presión – saturación, la discretización se lleva a cabo en un esténcil bidimensional mediante las diferencias finitas para bloques centrados. Los esquemas de solución numéricos son los métodos IMPES Clásico e IMPES Mejorado.

3.1 Flujo de Fluidos en Medios Porosos

La ecuación de balance local de una propiedad intensiva está dada por [28]:

$$\frac{\partial \psi}{\partial t} + \nabla(\vec{v}\psi) = q + \nabla\tau \tag{3.1}$$

donde t representa el tiempo, $\vec{v}$ es la velocidad del fluido, τ es la difusión y q la fuente. En términos generales, el primer término del lado izquierdo de (3.1) representa el cambio temporal de la propiedad intensiva, mientras que el segundo es el transporte de esta (o advección). Del lado derecho se observa a la fuente (o sumidero) y a la difusión.

En un sistema multifásico dentro de un medio poroso, la masa de fluido de la fase α es una propiedad extensiva que se escribe como:

$$E = \int_{B(t)} \phi \rho_\alpha S_\alpha \, d\vec{x} \tag{3.2}$$

donde ϕ es la porosidad del medio, ρ_α y S_α Son la densidad y la saturación de la fase α respectivamente. De acuerdo con la formulación axiomática, la correspondiente propiedad intensiva para (3.2) en este caso es

$$\psi_\alpha = \phi \rho_\alpha S_\alpha. \tag{3.3}$$

suponiendo que no hay difusión, es decir $\tau = 0$, y aplicando la ecuación (3.1), la ecuación de balance local para la masa de fluido de la fase α se escribe como

$$\frac{\partial(\phi \rho_\alpha S_\alpha)}{\partial t} + \nabla(\vec{v}_\alpha \phi \rho_\alpha S_\alpha) = q_\alpha \tag{3.4}$$

tomando en cuenta que la velocidad de Darcy la fase α se escribe como $\vec{u}_\alpha = \vec{v}_\alpha \phi S_\alpha$, la ecuación diferencial de balance de masa para la fase α, que forma parte de un sistema multifásico, se escribe como:

$$\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla(\rho_\alpha \vec{u}_\alpha) = q_\alpha \tag{3.5}$$

la ley de Darcy de forma diferencial para un sistema multifásico en 3D se expresa cómo

$$\vec{u}_\alpha = -\frac{\bar{\bar{k}}\kappa_{r\alpha}}{\mu_\alpha}(\nabla p_\alpha - \rho_\alpha \wp \nabla z) \tag{3.6}$$

donde, z, $\wp$, $k_{r\alpha}$, μ_α, p_α y ρ_α son la altura, la gravedad, así como la permeabilidad relativa, la viscosidad, la presión y la densidad, para la fase α, respectivamente.

Sustituyendo (3.6) en (3.5), se obtiene la ecuación general de balance de masa:

$$\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla\left(\rho_\alpha \frac{\bar{\bar{k}}\kappa_{r\alpha}}{\mu_\alpha}(\nabla p_\alpha - \rho_\alpha \wp \nabla z)\right) = q_\alpha \tag{3.7}$$

la ecuación (3.7) representa un conjunto de ecuaciones diferenciales parciales de segundo orden y en general no lineales para la conservación de la masa igual al número de fases que se encuentran presentes en el medio poroso.

Estas ecuaciones se completan con una serie de fórmulas constitutivas, las cuales describen la dependencia de una variable con respecto de las otras e integran el conocimiento científico y tecnológico del problema que se está estudiando. En un sistema multifásico totalmente saturado, generalmente, se usan las siguientes ecuaciones constitutivas

$$\sum_{\alpha=1}^{n} S_\alpha = 1 \tag{3.8}$$

$$p_{C,\alpha1\alpha2} = p_{\alpha1} - p_{\alpha2}, \qquad \alpha_1 \neq \alpha_2 \tag{3.9}$$

donde n es el número de fases, α_1 y α_2 representan la fase no mojante y la fase mojante, respectivamente, p_c se refiere a la presión capilar la cual es función de la saturación de la fase mojante. Además, existen expresiones empíricas las cuales relacionan la permeabilidad relativa como función de la saturación $k_{r\alpha} = k_{r\alpha}(S_\alpha)$, y la porosidad como función de la presión $\phi = \phi(p)$.

3.2 Discretización del Yacimiento

La discretización del yacimiento significa que el yacimiento se describe mediante un conjunto de bloques que componen una malla (o puntos de malla) cuyas propiedades, dimensiones, límites y ubicaciones en el yacimiento están bien definidos. Este trabajo se ocupa de los yacimientos discretizados que utilizan una malla de bloques centrados, aunque también es importante mencionar los yacimientos discretizados utilizando una malla difundida por puntos. La Figura 3.1 muestra la discretización del yacimiento en la dirección x porque el punto se coloca en el bloque i. La figura muestra cómo los bloques están vinculados entre sí: Bloque i y sus bloques vecinos $i-1$ e $i+1$ – las dimensiones de los bloques $(\Delta x_i, \Delta x_{i-1}, \Delta x_{i+1})$, las pardes internas de los bloques $(x_{i-1/2}, x_{i+1/2})$ respecto el punto i y las distancias entre las paredes $(\delta x_{i^-}, \delta x_{i^+})$, así como las distancias entre puntos $(\Delta x_{i-1/2}, \Delta x_{i+1/2})$. La terminología presentada en la Figura 3.1 es aplicable a los sistemas de mallas por bloques centrados y se distribuye en el punto en un flujo 1D en la dirección del eje x. La discretización de los yacimientos en las direcciones y y z utilizan una terminología similar. Además, a cada punto de la malla se le asigna la elevación y las propiedades de la roca, como la porosidad y la permeabilidad en las direcciones x, y y z. La transferencia de líquidos desde un bloque al resto del yacimiento tiene lugar a través de los bloques vecinos inmediatos. Cuando todo el yacimiento está discretizado, cada bloque está rodeado por un conjunto (grupo) de bloques vecinos. La Figura 3.2(a) muestra que hay dos bloques vecinos en el flujo de identificación a lo largo del eje x, la Figura 3.2(b) muestra que hay cuatro bloques vecinos en 2D y la Figura 3.2(c) muestra que son 6 bloques vecinos en 3D en el espacio $x-y-z$.

Debe estar claramente indicado que una vez que el yacimiento está discretizado y las propiedades de la roca se asignan a los bloques de la malla (o puntos de la malla), el espacio ya no es una variable y las funciones que dependen del espacio, como las propiedades intermedias , quedan bien definidas. En otras palabras, la discretización de yacimientos elimina el espacio como variable en la formulación del problema.

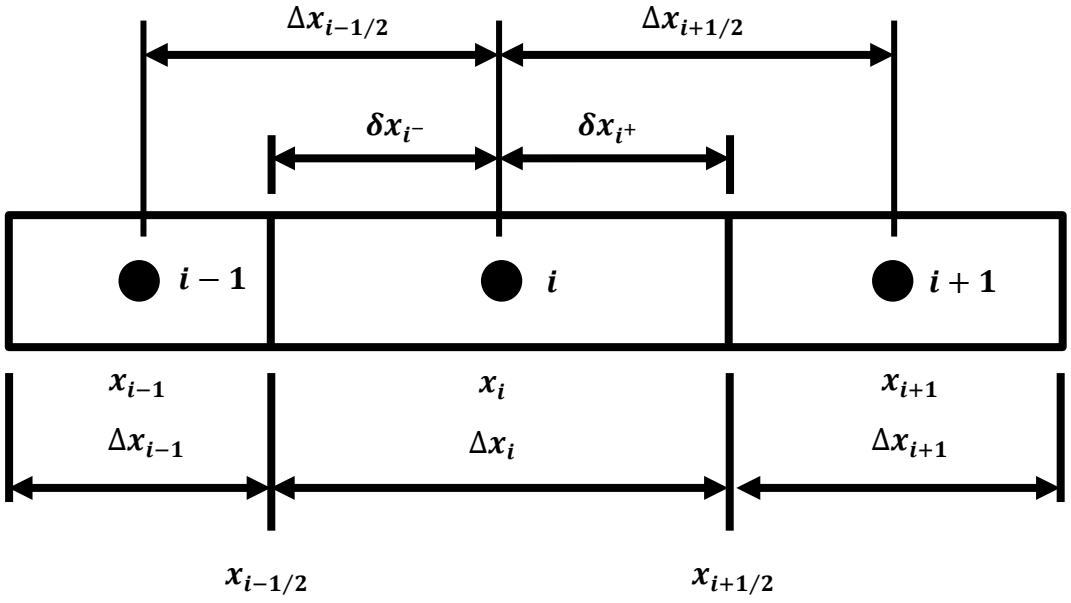


Figura 3.1 Relación entre un bloque y sus bloques adyacentes, tomada de [1].

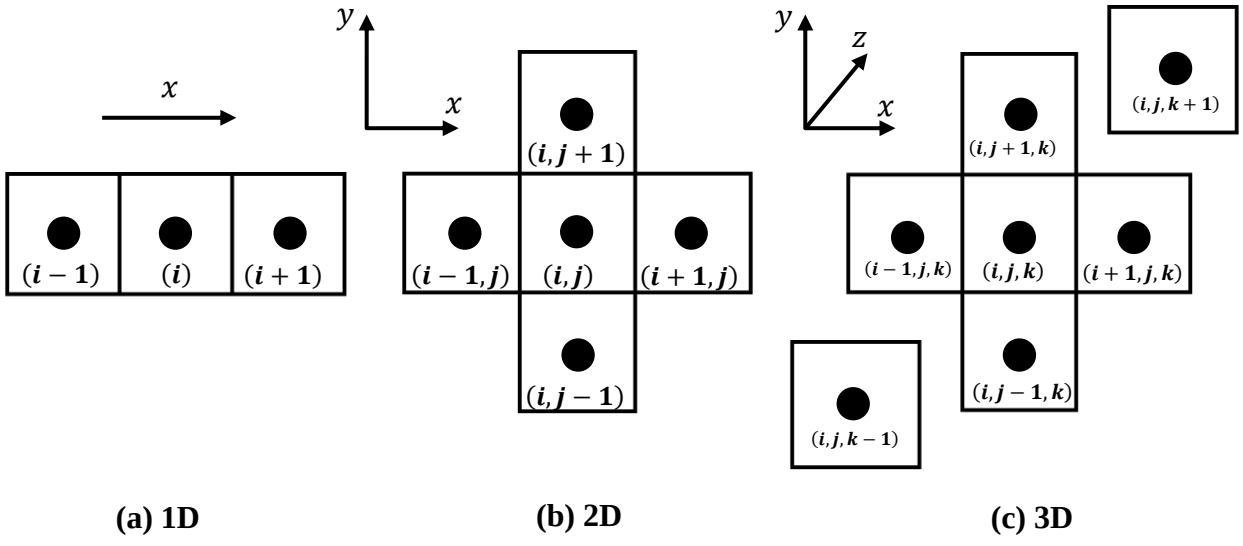


Figura 3.2 Relación de un bloque y sus bloques adyacentes para flujo en 1D, 2D y 3D utilizando notación de ingeniería, tomada de [1].

3.3 Flujo Multidimensional en Coordenadas Cartesianas

3.3.1 Identificación de Bloques y Ordenamiento de Nodos

Antes de escribir la ecuación de flujo para un yacimiento de 1D, 2D o 3D, se deben identificar y ordenar los bloques en el yacimiento discretizado. Cualquier bloque en el yacimiento puede identificarse ya sea por notación de ingeniería o por el número que tiene el bloque en un esquema de ordenamiento dado. La notación de ingeniería utiliza el orden del bloque en las direcciones x, y y z; es decir, identifica un bloque como (i, j, k), donde i, j y k son los órdenes del bloque en las tres direcciones x, y y z, respectivamente. La notación de ingeniería para la identificación de bloques es la más conveniente para ingresar la descripción del yacimiento (entrada) y para imprimir los resultados de la simulación (salida). La Figura 3.3 muestra la notación de ingeniería para la identificación de bloques en un yacimiento $2D$ que consta de 4×5 bloques. El ordenamiento de bloques no solo sirve para identificar bloques en el yacimiento, sino que también minimiza los cálculos en la matriz para obtener la solución de ecuaciones lineales. Hay muchos esquemas de ordenamiento de bloques, que incluyen el ordenamiento natural, el ordenamiento de cebra, el ordenamiento en diagonal (D2), el ordenamiento en diagonal alternada (D4), el ordenamiento cíclico y el ordenamiento de doble ciclo. Si el yacimiento tiene bloques activos dentro de sus límites externos, dichos bloques se omitirán y se continuará ordenando bloques activos [29].

(0,4)	(1,4)	(2,4)	(3,4)
(0,3)	(1,3)	(2,3)	(3,3)
(0,2)	(1,2)	(2,2)	(3,2)
(0,1)	(1,1)	(2,1)	(3,1)
(0,0)	(1,0)	(2,0)	(3,0)

Figura 3.3 Notación ingenieril para identificación de bloques.

Para yacimientos multidimensionales, el ordenamiento natural es el más simple de programar, pero es el menos eficiente para resolver ecuaciones lineales, mientras que el ordenamiento D4 requiere una programación complicada, pero es el más eficiente para obtener la solución cuando el número de bloques es grande. Sin embargo, si el número de bloques es muy grande, el esquema de ordenamiento de cebra se vuelve dos veces más eficiente que el ordenamiento D4 para obtener la solución [30]. La Figura 3.4 muestra los diversos esquemas de ordenamiento de bloques para el yacimiento 2D que se muestra en la Figura 3.3. Dada la notación de ingeniería para la identificación

de bloques, el ordenamiento de bloques se genera internamente en un simulador. Cualquier esquema de ordenamiento se vuelve aún más eficiente computacionalmente si el ordenamiento se realiza a lo largo de la dirección más corta, seguida por la dirección intermedia y finalmente la dirección más larga [29]. Los detalles relacionados con varios esquemas de ordenamiento y la eficiencia computacional en la resolución de ecuaciones lineales no se analizan más en esta tesis, pero se pueden encontrar en otros lugares [31, 32, 30].

16	17	18	19
12	13	14	15
8	9	10	11
4	5	6	7
0	1	2	3

a) Ordenamiento Natural

8	9	10	11
16	17	18	19
4	5	6	7
12	13	14	15
0	1	2	3

b) Ordenamiento de Cebra

13	16	18	19
9	12	15	17
5	8	11	14
2	4	7	10
0	1	3	6

c) Ordenamiento en Diagonal D2

7	18	9	19
15	6	17	8
3	14	5	16
11	2	13	4
0	10	1	12

d) Ordenamiento en Diagonal Alternada

10	9	8	7
11	18	17	6
12	19	16	5
13	14	15	4
0	1	2	3

e) Ordenamiento Cíclico

8	18	9	19
16	6	17	7
4	14	5	15
12	2	13	3
0	10	1	11

f) Ordenamiento de Doble Ciclo

Figura 3.4 Esquemas de Ordenamiento usados en la Simulación de Yacimientos, adaptad de [1].

3.4 Formulación Presión - Saturación

Si se define la movilidad de la fase λ_α como:

$$\lambda_\alpha = \frac{\kappa_{r\alpha}}{\mu_\alpha} \tag{3.10}$$

entonces se puede reescribir la ecuación de balance de masa en términos de la movilidad de la fase de la siguiente manera:

$$\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla\left(\rho_\alpha \bar{\bar{k}}\lambda_\alpha(\nabla p_\alpha - \rho_\alpha g \nabla z)\right) = q_\alpha \tag{3.11}$$

el modelo matemático para el flujo de fluidos en medios porosos es el descrito por la ecuación (3.11) la cual explica la acumulación de masa en un volumen diferencial por unidad de tiempo para cada fase, donde cada fase tiene su propia velocidad de Darcy y su gasto másico, obteniendo un sistema bifásico considerando las fases agua y aceite.

Dicho sistema se encuentra fuertemente acoplado, lo que significa que las incógnitas de las ecuaciones de presión y saturación de la fase dependen una de la otra y se encuentran explícitamente en la misma ecuación, además existen propiedades como las permeabilidades relativas y la presión capilar que dependen de éstas.

Entonces, mediante la aplicación de la formulación del flujo fraccional, se desacopla el sistema generado por (3.11) en términos de la movilidad para cada fase, en una ecuación para la presión y N-1 ecuaciones de transporte para la saturación.

Tal formulación produce un sistema de ecuaciones, que es débilmente acoplado y que puede resolverse de manera iterativa y secuencial.

3.4.1 Ecuación de presión

Para obtener la ecuación de presión, primero se divide la ecuación (3.11) por ρ_α, con lo que se obtiene

$$\frac{1}{\rho_\alpha}\left[\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla\cdot(\rho_\alpha \vec{u}_\alpha) - q_\alpha\right] = 0 \tag{3.12}$$

y sumando sobre todas las fases se tiene:

$$\sum_\alpha \left\{\frac{1}{\rho_\alpha}\left[\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla\cdot(\rho_\alpha \vec{u}_\alpha) - q_\alpha\right]\right\} = 0 \tag{3.13}$$

desarrollando las derivadas de la ecuación anterior se llega a lo siguiente

$$\sum_{\alpha}\left\{\frac{1}{\rho_\alpha}\left[\rho_\alpha S_\alpha \frac{\partial \phi}{\partial t}+\phi S_\alpha \frac{\partial \rho_\alpha}{\partial t}+\rho_\alpha \phi \frac{\partial S_\alpha}{\partial t}+\rho_\alpha \nabla\cdot\vec{u}_\alpha+\vec{u}_\alpha\cdot\nabla\rho_\alpha-q_\alpha\right]\right\}=0 \tag{3.14}$$

aplicando la ecuación constitutiva (3.8) a la ecuación (3.14) se puede escribir

$$\frac{\partial \phi}{\partial t}+\sum_{\alpha}\nabla\cdot\vec{u}_\alpha+\sum_{\alpha}\frac{1}{\rho_\alpha}\left[\phi S_\alpha \frac{\partial \rho_\alpha}{\partial t}+u_\alpha\cdot\nabla\rho_\alpha\right]-\sum_{\alpha}\frac{q_\alpha}{\rho_\alpha}=0 \tag{3.15}$$

ahora se define ahora la velocidad $\vec{u}$ total como

$$\vec{u}=\sum_{\alpha}\vec{u}_\alpha \tag{3.16}$$

y aplicando la divergencia a esta relación se llega a

$$\nabla\cdot\vec{u}=\nabla\cdot\sum_{\alpha}\vec{u}_\alpha=\vec{u}=\sum_{\alpha}\nabla\cdot\vec{u}_\alpha \tag{3.17}$$

ahora, de la definición de la ley de Darcy (3.6) para flujo multifásico y (3.10) se tiene que:

$$\vec{u}=\sum_{\alpha}\vec{u}_\alpha=\sum_{\alpha}-\bar{\bar{k}}\lambda_\alpha(\nabla p_\alpha-\rho_\alpha g\nabla z) \tag{3.18}$$

se puede también definir una función de flujo fraccional para la fase α como:

$$f_\alpha=\frac{\lambda_\alpha}{\lambda}\rightarrow\lambda_\alpha=f_\alpha\lambda \tag{3.19}$$

donde $\lambda=\sum_\alpha\lambda_\alpha$ es la movilidad total y por lo tanto se cumple $\Sigma_\alpha f_\alpha=1$. Luego sustituyendo (3.19) en (3.18) se obtiene

$$\vec{u}=\bar{\bar{k}}\lambda\left[\sum_{\alpha}f_\alpha\nabla p_\alpha-\sum_{\alpha}f_\alpha\rho_\alpha g\nabla z\right] \tag{3.20}$$

sustituyendo la ecuación (3.20) en (3.15) se obtiene la ecuación del flujo fraccional:

$$\frac{\partial \phi}{\partial t}-\bar{\bar{k}}\lambda\left[\sum_{\alpha}f_\alpha\nabla p_\alpha-\sum_{\alpha}f_\alpha\rho_\alpha g\nabla z\right]+\sum_{\alpha}\frac{1}{\rho_\alpha}\left[\phi S_\alpha\frac{\partial \rho_\alpha}{\partial t}+\vec{u}_\alpha\cdot\nabla\rho_\alpha\right]-\sum_{\alpha}\frac{q_\alpha}{\rho_\alpha}=0 \tag{3.21}$$

3.4.2 Ecuación de Saturación

Para identificar cada una de las fases se utiliza el subíndice *w* para indicar la fase agua y el subíndice *o* para indicar la fase aceite. Entonces las relaciones constitutivas, se convierten en:

$$S_w + S_o = 1 \tag{3.22}$$

$$p_c \equiv p_{cow} = p_o - p_w \tag{3.23}$$

considerando el sistema (3.11) y sustituyendo para las fases de aceite y agua

$$\frac{\partial(\phi\rho_o S_o)}{\partial t} + \nabla\left(\rho_o \bar{\bar{k}}\lambda_o(\nabla p_o - \rho_o g \nabla z)\right) = q_o \tag{3.24}$$

$$\frac{\partial(\phi\rho_w S_w)}{\partial t} + \nabla\left(\rho_w \bar{\bar{k}}\lambda_w(\nabla p_w - \rho_w g \nabla z)\right) = q_w \tag{3.25}$$

donde la ecuación (3.24) es la ecuación de balance de masa para la fase aceite y la ecuación (3.25) es la ecuación de balance de masa para la fase agua. Para la ecuación de la fase agua, se tiene:

$$\frac{\partial(\phi\rho_w S_w)}{\partial t} + \nabla\left(\rho_w \bar{\bar{k}}\lambda_w(\nabla(p_o - p_c) - \rho_w g \nabla z)\right) = q_w \tag{3.26}$$

expandiendo las derivadas del gradiente de la ecuación (3.26), se llega a la ecuación de conservación de masa de la fase *w* en función la presión capilar y la presión del aceite:

$$\frac{\partial(\phi\rho_w S_w)}{\partial t} + \nabla\left(\rho_w \bar{\bar{k}}\lambda_w(\nabla p_o - \nabla p_c - \rho_w g \nabla z)\right) = q_w \tag{3.27}$$

$$\phi\frac{\partial S_w}{\partial t} + \nabla\left(\bar{\bar{k}}\lambda_w(\nabla p_o - \nabla p_c - \rho_w g \nabla z)\right) = \frac{q_w}{\rho_w} \tag{3.28}$$

las ecuaciones (3.21) ,(3.27), (3.28) son el punto de partida para desarrollar los simuladores que se implementaran a partir de los algoritmos IMPES Clásico e IMPES Mejorado.

3.5 Modelo Conceptual para Flujo Incompresible

Se considera que el flujo bifásico ocurre en un medio poroso en el que el desplazamiento de aceite es provocado por inyección de agua con una tasa de inyección constante en donde no existe transferencia de calor:

1. Se consideran los efectos de la presión capilar.
2. Se consideran los efectos de la gravedad.
3. Los fluidos son incompresibles e inmiscibles.

Con las consideraciones, anteriores la ecuación de conservación de masa de la fase *w* (3.27) es la ecuación de saturación correspondiente al modelo conceptual planteado. La presión capilar es función de la saturación de la fase mojante por lo que el gradiente de ésta puede escribirse como:

$$\nabla p_c = \frac{\partial p_c}{\partial S_w} \nabla S_w \tag{3.29}$$

sustituyendo (3.29) en la ecuación (3.27), se llega a:

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot \left[\bar{\bar{k}} \lambda_w \left(\nabla p_o - \frac{\partial p_c}{\partial S_w} \nabla S_w - \rho_w g \nabla z \right) \right] = \frac{q_w}{\rho_w} \tag{3.30}$$

ahora aplicando las consideraciones hechas a la ecuación de conservación de masa de la fase aceite (3.24) , se obtiene:

$$-\phi \frac{\partial S_w}{\partial t} - \nabla \cdot \left[\bar{\bar{k}} \lambda_o (\nabla p_o - \rho_o g \nabla z) \right] = \frac{q_o}{\rho_o} \tag{3.31}$$

por lo que finalmente se obtiene un sistema de dos ecuaciones parciales con dos incógnitas, a saber, la presión del aceite p_o y la saturación de agua S_w, si se consideran las propiedades del medio y de los fluidos como conocidas en el entendido de que si se conoce la saturación de agua también es posible conocer la presión del aceite. Por otra parte, si se conoce la presión capilar, la presión de la fase agua puede calcularse mediante la presión de la fase agua mediante $p_w = p_o - p_c$.

Para simplificar el sistema (3.30) y (3.31) se utiliza la formulación Presión-Saturación y se aplican las consideraciones del modelo conceptual a la ecuación de presión se llaga a:

$$\frac{\partial \phi}{\partial t} - \bar{\bar{k}} \lambda \left[\sum_\alpha f_\alpha \nabla p_\alpha - \sum_\alpha f_\alpha \rho_\alpha g \nabla z \right] + \sum_\alpha \frac{1}{\rho_\alpha} \left[\phi S_\alpha \frac{\partial \rho_\alpha}{\partial t} + \vec{u}_\alpha \cdot \nabla \rho_\alpha \right] = \sum_\alpha \frac{q_\alpha}{\rho_\alpha}$$

simplificando :

$$-\nabla \cdot \bar{\bar{k}}\lambda[f_w \nabla p_w + f_o \nabla p_o - (\rho_o + \rho_o)\wp \nabla z] = \frac{q_o}{\rho_o} + \frac{q_w}{\rho_w} \tag{3.32}$$

también se tiene que $f_o + f_w = 1$ y $p_w = p_o - p_c$, por lo tanto, desarrollando la ecuación (3.32)

$$-\nabla \cdot \bar{\bar{k}}\lambda[f_w(\nabla p_o - \nabla p_c) + f_o \nabla p_o - (f_w \rho_w + f_o \rho_o)\wp \nabla z] = \frac{q_o}{\rho_o} + \frac{q_w}{\rho_w}$$

$$-\nabla \cdot \bar{\bar{k}}\lambda[\nabla p_o - f_w \nabla p_c - (f_w \rho_w + f_o \rho_o)\wp \nabla z] = \frac{q_o}{\rho_o} + \frac{q_w}{\rho_w}$$

$$-\nabla \cdot \bar{\bar{k}}\lambda\left[\nabla p_o - f_w \frac{\partial p_c}{\partial S_w}\nabla S_w - (f_w \rho_w + f_o \rho_o)\wp \nabla z\right] = \frac{q_o}{\rho_o} + \frac{q_w}{\rho_w}$$

finalmente se obtiene:

$$-\nabla \cdot \left[\bar{\bar{k}}\lambda \nabla p_o - \bar{\bar{k}}\lambda_w \nabla p_c - \bar{\bar{k}}(\lambda_w \rho_w + \lambda_o \rho_o)\wp \nabla z\right] = \frac{q_o}{\rho_o} + \frac{q_w}{\rho_w} \tag{3.33}$$

y

$$-\nabla \cdot \left[\bar{\bar{k}}\lambda \nabla p_o - \bar{\bar{k}}\lambda_w \frac{\partial p_c}{\partial S_w}\nabla S_w - \bar{\bar{k}}(\lambda_w \rho_w + \lambda_o \rho_o)\wp \nabla z\right] = \frac{q_o}{\rho_o} + \frac{q_w}{\rho_w} \tag{3.34}$$

tal que la ecuación (3.34) es la ecuación de presión correspondiente al modelo conceptual anterior.

3.6 Modelo Numérico de Flujo Bifásico Incompresible

En esta sección se muestran las discretizaciones que se utilizan para ambos modelos matemáticos y esquemas numéricos bajo los que se realizan las comparaciones de eficiencia y rendimiento. Las nomenclaturas para cada modelo se muestran en el ANEXO A, del mismo modo los modelos para los pozos se muestran con más detalle en el ANEXO B

3.6.1 Solución Numérica de Flujo Bifásico Mediante los Métodos IMPES Clásico e IMPES Mejorado

Observe que los sistemas de ecuaciones diferenciales (3.28) y (3.33), (3.30) y (3.34) son no lineales y acopladas. En general el método IMPES es un método muy poderoso para resolver los sistemas anteriores, el flujo de dos fases (particularmente para fluidos incompresibles o ligeramente compresibles), solo se analiza este enfoque de solución para este tipo de flujo.

Originalmente la idea básica del método IMPES Clásico para resolver (3.28)−(3.33) y (3.30)−(3.34), es separar el cálculo de la presión del de la saturación.

Es decir, el sistema acoplado se divide en una ecuación de presión y una ecuación de saturación, las cuales se resuelven utilizando enfoques de aproximación de tiempo implícitos y explícitos, respectivamente.

En el **IMPES Clásico** resuelve la ecuación de presión con un esquema de discretización en el tiempo implícito y la saturación explícitamente, considerando un mismo paso de tiempo para ambas ecuaciones. Es importante señalar que, este paso de tiempo debe ser lo suficientemente pequeño para que este método sea estable. Por otra parte, el **IMPES Mejorado** utiliza un paso de tiempo diferente en la solución de la saturación, esto con el objetivo de controlar la variación máxima de la saturación y con ello estabilizar el método. Cabe recalcar que, para este método, el paso de tiempo para la presión siempre será mayor o en su defecto bajo ciertas condiciones, igual al de la saturación. Como ejemplo, se consideran los sistemas de ecuaciones (3.28) y (3.33), (3.30) y (3.34) discretizados por diferencias finitas para bloques centrados utilizando el esquema de los cinco puntos y cuya discretización se realiza en un medio bidimensional y sin considerar el efecto de la gravedad; entonces reescribiendo (3.28) y (3.33), (3.30) y (3.34) en términos de cada dirección coordenada (x, y) se obtiene:

$$\left\{\frac{\partial}{\partial x}\left(k\lambda\frac{\partial p_o}{\partial x}\right)-\frac{\partial p_c}{\partial S_w}\frac{\partial}{\partial x}\left(k\lambda_w\frac{\partial S_w}{\partial x}\right)\right\}-\left\{\frac{\partial}{\partial y}\left(k\lambda\frac{\partial p_o}{\partial y}\right)-\frac{\partial p_c}{\partial S_w}\frac{\partial}{\partial y}\left(k\lambda_w\frac{\partial S_w}{\partial y}\right)\right\}=\tilde{q}_w+\tilde{q}_o \quad (3.35)$$

$$\phi\frac{\partial S_w}{\partial t}-\left[\left\{\frac{\partial}{\partial x}\left(k\lambda\frac{\partial p_o}{\partial x}\right)-\frac{\partial p_c}{\partial S_w}\frac{\partial}{\partial x}\left(k\lambda_{\mathrm{w}}\frac{\partial p_c}{\partial x}\right)\right\}+\left\{\frac{\partial}{\partial y}\left(k\lambda\frac{\partial p_o}{\partial y}\right)-\frac{\partial p_c}{\partial S_w}\frac{\partial}{\partial y}\left(k\lambda_{\mathrm{w}}\frac{\partial S_w}{\partial y}\right)\right\}\right]=\tilde{q}_w \quad (3.36)$$

para (3.30) y (3.34), y:

$$\left\{\frac{\partial}{\partial x}\left(k\lambda\frac{\partial p_o}{\partial x}\right)-\frac{\partial}{\partial x}\left(k\lambda_w\frac{\partial p_c}{\partial x}\right)\right\}-\left\{\frac{\partial}{\partial y}\left(k\lambda\frac{\partial p_o}{\partial y}\right)-\frac{\partial}{\partial y}\left(k\lambda_w\frac{\partial p_c}{\partial y}\right)\right\}=\tilde{q}_w+\tilde{q}_o \tag{3.37}$$

$$\phi\frac{\partial S_w}{\partial t}-\left[\left\{\frac{\partial}{\partial x}\left(k\lambda\frac{\partial p_o}{\partial x}\right)-\frac{\partial}{\partial x}\left(k\lambda_w\frac{\partial p_c}{\partial x}\right)\right\}+\left\{\frac{\partial}{\partial y}\left(k\lambda\frac{\partial p_o}{\partial y}\right)-\frac{\partial}{\partial y}\left(k\lambda_w\frac{\partial p_c}{\partial y}\right)\right\}\right]=\tilde{q}_w \tag{3.38}$$

para (3.28) y (3.33).

Entonces las discretizaciones para los sistemas de ecuaciones (3.35) y (3.36) quedaría

$$\begin{aligned}
&-\left\{\bar{\bar{k}}_{xx}\lambda\big|_e\left(\frac{p_{o,E}-p_{o,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda\big|_w\left(\frac{p_{o,P}-p_{o,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&-\left\{\bar{\bar{k}}_{yy}\lambda\big|_n\left(\frac{p_{o,N}-p_{o,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda\big|_s\left(\frac{p_{o,P}-p_{o,S}}{\delta y_s}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_e\left(\frac{S_{w,E}-S_{w,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_w\left(\frac{S_{w,P}-S_{w,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_n\left(\frac{S_{w,N}-S_{w,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_s\left(\frac{S_{w,P}-S_{w,S}}{\delta y_s}\right)\right\}\Delta x\Delta z\\
&=\frac{\tilde{q}_w}{\rho_w}+\frac{\tilde{q}_o}{\rho_o}
\end{aligned} \tag{3.39}$$

$$\begin{aligned}
&\phi\frac{\partial S_w}{\partial t}-\left\{\bar{\bar{k}}_{xx}\lambda_w\big|_e\left(\frac{p_{o,E}-p_{o,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda_w\big|_w\left(\frac{p_{o,P}-p_{o,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&-\left\{\bar{\bar{k}}_{yy}\lambda_w\big|_n\left(\frac{p_{o,N}-p_{o,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda_w\big|_s\left(\frac{p_{o,P}-p_{o,S}}{\delta y_s}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_e\left(\frac{S_{w,E}-S_{w,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_w\left(\frac{S_{w,P}-S_{w,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_n\left(\frac{S_{w,N}-S_{w,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_s\left(\frac{S_{w,P}-S_{w,S}}{\delta y_s}\right)\right\}\Delta x\Delta z\\
&=\tilde{q}_w
\end{aligned} \tag{3.40}$$

y para los sistemas de ecuaciones (3.37) y (3.38) sería

$$\begin{aligned}
&-\left\{\overline{\overline{k}}_{xx}\lambda\big|_e\left(\frac{p_{o,E}-p_{o,P}}{\delta x_e}\right)-\overline{\overline{k}}_{xx}\lambda\big|_w\left(\frac{p_{o,P}-p_{o,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&-\left\{\overline{\overline{k}}_{yy}\lambda\big|_n\left(\frac{p_{o,N}-p_{o,P}}{\delta y_n}\right)-\overline{\overline{k}}_{yy}\lambda\big|_s\left(\frac{p_{o,P}-p_{o,S}}{\delta y_s}\right)\right\}\Delta y\Delta z\\
&+\left\{\overline{\overline{k}}_{xx}\lambda_w\big|_e\left(\frac{p_{c,E}-p_{c,P}}{\delta x_e}\right)-\overline{\overline{k}}_{xx}\lambda_w\big|_w\left(\frac{p_{c,P}-p_{c,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&+\left\{\overline{\overline{k}}_{yy}\lambda_w\big|_n\left(\frac{p_{c,N}-p_{c,P}}{\delta y_n}\right)-\overline{\overline{k}}_{yy}\lambda_w\big|_s\left(\frac{p_{c,P}-p_{c,S}}{\delta y_s}\right)\right\}\Delta x\Delta z=\tilde{q}_w+\tilde{q}_o
\end{aligned} \tag{3.41}$$

$$\begin{aligned}
&\phi\frac{\partial S_w}{\partial t}-\left\{\overline{\overline{k}}_{xx}\lambda_w\big|_e\left(\frac{p_{o,E}-p_{o,P}}{\delta x_e}\right)-\overline{\overline{k}}_{xx}\lambda_w\big|_w\left(\frac{p_{o,P}-p_{o,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&-\left\{\overline{\overline{k}}_{yy}\lambda_w\big|_n\left(\frac{p_{o,N}-p_{o,P}}{\delta y_n}\right)-\overline{\overline{k}}_{yy}\lambda_w\big|_s\left(\frac{p_{o,P}-p_{o,S}}{\delta y_s}\right)\right\}\Delta y\Delta z\\
&+\left\{\overline{\overline{k}}_{xx}\lambda_w\big|_e\left(\frac{p_{c,E}-p_{c,P}}{\delta x_e}\right)-\overline{\overline{k}}_{xx}\lambda_w\big|_w\left(\frac{p_{c,P}-p_{c,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&+\left\{\overline{\overline{k}}_{yy}\lambda_w\big|_n\left(\frac{p_{c,N}-p_{c,P}}{\delta y_n}\right)-\overline{\overline{k}}_{yy}\lambda_w\big|_s\left(\frac{p_{c,P}-p_{c,S}}{\delta y_s}\right)\right\}\Delta x\Delta z=\tilde{q}_w
\end{aligned} \tag{3.42}$$

los sistemas discretizados (3.39) − (3.40) y (3.41) − (3.42) corresponden a las discretizaciones espaciales, donde $E=i+1$, $W=i-1$, $N=j+1$, $S=j-1$. La discretización de los modelos se describe en el ANEXO C.

3.7 Enfoques de Solución en el Tiempo

3.7.1 IMPES Clásico

Sea $J=(0,T]$ $(T>0)$ el intervalo de tiempo de interés, y para un entero positivo N, sea $0=t^0<t^1<\cdots<t^N=\boldsymbol{\tau}$ una partición de J. Para el cálculo de la presión en el método IMPES Clásico, se supone que se conoce la saturación S_w en (3.39) y (3.41) se resuelven implícitamente para p_o. Es decir, para cada $n=0,1,\ldots,p_o^n$ satisface:

$$
\begin{aligned}
&-\{T_e^n(p_{o,E}^n - p_{o,P}^n) - T_w^n(p_{o,P}^n - p_{o,W}^n)\}\Delta y\Delta z \\
&-\{T_n^n(p_{o,N}^n - p_{o,P}^n) - T_s^n(p_{o,N}^n - p_{o,P}^n)\}\Delta y\Delta z \\
&+\left\{T_{w,e}^n \frac{\partial p_c}{\partial S_w}\Big|_e^n (S_{w,E}^n - S_{w,P}^n) - T_{w,w}^n \frac{\partial p_c}{\partial S_w}\Big|_w^n (S_{w,P}^n - S_{w,W}^n)\right\}\Delta y\Delta z \\
&+\left\{T_{w,n}^n \frac{\partial p_c}{\partial S_w}\Big|_N^n (S_{w,N}^n - S_{w,P}^n) - T_{w,s}^n \frac{\partial p_c}{\partial S_w}\Big|_s^n (S_{w,P}^n - S_{w,S}^n)\right\}\Delta x\Delta z \\
&= \tilde{q}_o^n + \tilde{q}_w^n
\end{aligned}
\tag{3.43}
$$

$$
\begin{aligned}
&-\{T_e^n(p_{o,E}^n - p_{o,P}^n) - T_w^n(p_{o,P}^n - p_{o,W}^n)\}\Delta y\Delta z \\
&-\{T_n^n(p_{o,N}^n - p_{o,P}^n) - T_s^n(p_{o,N}^n - p_{o,P}^n)\}\Delta y\Delta z \\
&+\{T_{w,e}^n(p_{c,E}^n - p_{c,P}^n) - T_{w,w}^n(p_{c,P}^n - p_{c,W}^n)\}\Delta y\Delta z \\
&+\{T_{w,n}^n(p_{c,N}^n - p_{c,P}^n) - T_{w,s}^n(p_{c,P}^n - p_{c,S}^n)\}\Delta x\Delta z = \tilde{q}_o^n + \tilde{q}_w^n
\end{aligned}
\tag{3.44}
$$

donde se S_w^n y la notación análoga es válida para otras cantidades.

Con el IMPES Clásico, (3.40) y (3.42) se resuelve explícitamente para S_w; es decir, para cada $n = 0, 1, 2, \ldots, S_w^{n+1}$ satisface

$$
\begin{aligned}
\phi\Delta V\frac{{S_w}^{n+1} - {S_w}^n}{\Delta t} &- \{T_e^n(p_{o,E}^n - p_{o,P}^n) - T_w^n(p_{o,P}^n - p_{o,W}^n)\}\Delta y\Delta z \\
&- \{T_n^n(p_{o,N}^n - p_{o,P}^n) - T_s^n(p_{o,N}^n - p_{o,P}^n)\}\Delta y\Delta z \\
&+ \left\{T_{w,e}^n \frac{\partial p_c}{\partial S_w}\Big|_e^n (S_{w,E}^n - S_{w,P}^n) - T_{w,w}^n \frac{\partial p_c}{\partial S_w}\Big|_w^n (S_{w,P}^n - S_{w,W}^n)\right\}\Delta y\Delta z \\
&+ \left\{T_{w,n}^n \frac{\partial p_c}{\partial S_w}\Big|_N^n (S_{w,N}^n - S_{w,P}^n) - T_{w,s}^n \frac{\partial p_c}{\partial S_w}\Big|_s^n (S_{w,P}^n - S_{w,S}^n)\right\}\Delta x\Delta z = \tilde{q}_w^n
\end{aligned}
\tag{3.45}
$$

$$
\begin{aligned}
\phi\Delta V\frac{{S_w}^{n+1} - {S_w}^n}{\Delta t} &- \{T_e^n(p_{o,E}^n - p_{o,P}^n) - T_w^n(p_{o,P}^n - p_{o,W}^n)\}\Delta y\Delta z \\
&- \{T_n^n(p_{o,N}^n - p_{o,P}^n) - T_s^n(p_{o,N}^n - p_{o,P}^n)\}\Delta y\Delta z \\
&+ \{T_{w,e}^n(p_{c,E}^n - p_{c,P}^n) - T_{w,w}^n(p_{c,P}^n - p_{c,W}^n)\}\Delta y\Delta z \\
&+ \{T_{w,n}^n(p_{c,N}^n - p_{c,P}^n) - T_{w,s}^n(p_{c,P}^n - p_{c,S}^n)\}\Delta x\Delta z = \tilde{q}_w^n
\end{aligned}
\tag{3.46}
$$

donde $T_e = \overline{\overline{k}}_{xx}\lambda/\delta x_e\big|_e$, $T_w = \overline{\overline{k}}_{xx}\lambda/\delta x_w\big|_w$, $T_n = \overline{\overline{k}}_{yy}\lambda/\delta y_n\big|_n$, $T_s = \overline{\overline{k}}_{yy}\lambda/\delta y_s\big|_s$, $T_{w,e} = \overline{\overline{k}}_{xx}\lambda_w/\delta x_e\big|_e$, $T_{w,w} = \overline{\overline{k}}_{xx}\lambda_w/\delta x_w\big|_w$, $T_{w,n} = \overline{\overline{k}}_{yy}\lambda_w/\delta y_n\big|_n$, $T_{w,s} = \overline{\overline{k}}_{yy}\lambda_w/\delta y_s\big|_s$, $T_e = \overline{\overline{k}}_{xx}\lambda/\delta x_e\big|_e$, $\tilde{q}_w = \frac{q_w}{\rho_w}$ y $\tilde{q}_o = \frac{q_o}{\rho_o}$.

El método IMPES es el siguiente: Después del inicio, para n = 0, 1,. . . , se usa (3.43) y/o (3.44) y S_w^n para evaluar p_o^n; a continuación, se utiliza S_w^n, p_o^n y (3.45) y/o (3.46) para calcular S_w^{n+1}. Como se señaló, el paso de tiempo Δt debe ser lo suficientemente pequeño para que este método sea estable. Para controlar la variación de la saturación, se necesita encontrar un paso de tiempo Δt adecuado antes de resolver la ecuación (3.45) y/o (3.46) para S_w^{n+1} para cada $n = 0, 1, \ldots$.

La estrategia de control se define como sigue: Se calcula el valor máximo de ∂S_w^{n+1} / ∂t en todos los nodos computacionales, denotado por $(\partial S_w^{n+1} / \partial t)_{max}$, que es, para (3.45) y/o (3.46)

$$\left(\frac{\partial S_w^{n+1}}{\partial t}\right)_{max} = \max_{i,j,k}\left(\frac{G(p_o^n, S_w^n)}{V\phi}\right)_{i,j,k}$$

donde $G\ (p_o^n, S_w^n)$ representa el lado derecho de la ecuación (3.45) y/o (3.46) y el máximo se toma sobre todos los bloques de la malla. Luego se aplica la siguiente fórmula para encontrar Δt:

$$\Delta t = {DS_{max}} \Big/ \left(\frac{\partial S_w^{n+1}}{\partial t}\right)_{max} \tag{3.47}$$

donde DS_{max}es la variación máxima de la saturación permitida (establecida por el usuario). Ahora, se usa este paso de tiempo en (3.45) y/o (3.46) para obtener S_w^{n+1}. Este enfoque garantiza que la variación de saturación no exceda DS_{max}. Tenga en cuenta que DS_{max} puede depender del nivel de tiempo n.

3.7.1 IMPES Mejorado.

Nuevamente, para un entero positivo N, sea $0 = t^0 < t^1 < \cdots < t^N = T$ una partición de J en subintervalos $J^n = (t^{n-1}, t^n]$, con longitud $\Delta t_p^n = t^n - t^{n-1}$

Esta partición de tiempo se utiliza para la presión. Para la saturación, cada subintervalo J^n se divide en subintervalos $J^{n,m} = (t^{n-1,m-1}, t^{n-1,m}]$:

$$t^{n-1,m} = t^{n-1} + m\Delta t_p^n / M^{n,m}, \qquad m = 1, 2, \ldots, M^n.$$

La longitud de $J^{n,m}$ se denota por $\Delta t_S^{n,m} = t^{n-1,m} - t^{n-1,m-1}, m = 1, \ldots, M^n, n = 0, 1, \ldots$. El número de pasos, M^n, puede depender de n. A continuación, simplemente se escribe $t^{n-1,0} = t^{n-1}$ y se estable $v^{n,m} = v\ (\cdot, t^{n,m})$.

En el método IMPES Mejorado, el cálculo de la presión es el mismo que en el sistema (3.45) y/o (3.46): Para cada $n = 0, 1, \ldots, p_o^n$ satisface:

$$\begin{aligned}
&-\{T_e^n(p_{o,E}^n - p_{o,P}^n) - T_w^n(p_{o,P}^n - p_{o,W}^n)\}\Delta y\Delta z \\
&\quad - \{T_n^n(p_{o,N}^n - p_{o,P}^n) - T_s^n(p_{o,N}^n - p_{o,P}^n)\}\Delta y\Delta z \\
&\quad + \left\{T_{w,e}^n \frac{\partial p_c}{\partial S_w}\Big|_e^n (S_{w,E}^n - S_{w,P}^n) - T_{w,w}^n \frac{\partial p_c}{\partial S_w}\Big|_w^n (S_{w,P}^n - S_{w,W}^n)\right\}\Delta y\Delta z \\
&\quad + \left\{T_{w,n}^n \frac{\partial p_c}{\partial S_w}\Big|_N^n (S_{w,N}^n - S_{w,P}^n) - T_{w,s}^n \frac{\partial p_c}{\partial S_w}\Big|_s^n (S_{w,P}^n - S_{w,S}^n)\right\}\Delta x\Delta z \\
&\quad = \tilde{q}_o^n + \tilde{q}_w^n
\end{aligned} \tag{3.48}$$

$$\begin{aligned}
&-\{T_e^n(p_{o,E}^n - p_{o,P}^n) - T_w^n(p_{o,P}^n - p_{o,W}^n)\}\Delta y\Delta z \\
&\quad - \{T_n^n(p_{o,N}^n - p_{o,P}^n) - T_s^n(p_{o,N}^n - p_{o,P}^n)\}\Delta y\Delta z \\
&\quad + \{T_{w,e}^n(p_{c,E}^n - p_{c,P}^n) - T_{w,w}^n(p_{c,P}^n - p_{c,W}^n)\}\Delta y\Delta z \\
&\quad + \{T_{w,n}^n(p_{c,N}^n - p_{c,P}^n) - T_{w,s}^n(p_{c,P}^n - p_{c,S}^n)\}\Delta x\Delta z = \tilde{q}_o^n + \tilde{q}_w^n
\end{aligned} \tag{3.49}$$

la diferencia está en el cálculo de la saturación: Para $m = 1, \ldots, M^n, n = 0, 1, \ldots$, se encuentra $S_w^{n+1,m}$ tal que:

$$\begin{aligned}
&\phi\Delta V \frac{{S_w}^{n+1,m} - {S_w}^{n+1,m-1}}{\Delta t} \\
&\quad - \{T_e^{n+1,m-1}(p_{o,E}^{n+1,m-1} - p_{o,P}^{n+1,m-1}) \\
&\quad - T_w^{n+1,m-1}(p_{o,P}^{n+1,m-1} - p_{o,W}^{n+1,m-1})\}\Delta y\Delta z \\
&\quad - \{T_n^{n+1,m-1}(p_{o,N}^{n+1,m-1} - p_{o,P}^{n+1,m-1}) \\
&\quad - T_s^{n+1,m-1}(p_{o,N}^{n+1,m-1} - p_{o,P}^{n+1,m-1})\}\Delta y\Delta z \\
&\quad + \left\{T_{w,e}^{n+1,m-1} \frac{\partial p_c}{\partial S_w}\Big|_e^{n+1,m-1} (S_{w,E}^{n+1,m-1} - S_{w,P}^{n+1,m-1})\right. \\
&\quad \left. - T_{w,w}^{n+1,m-1} \frac{\partial p_c}{\partial S_w}\Big|_w^{n+1,m-1} (S_{w,P}^{n+1,m-1} - S_{w,W}^{n+1,m-1})\right\}\Delta y\Delta z \\
&\quad + \left\{T_{w,n}^{n+1,m-1} \frac{\partial p_c}{\partial S_w}\Big|_N^{n+1,m-1} (S_{w,N}^{n+1,m-1} - S_{w,P}^{n+1,m-1})\right. \\
&\quad \left. - T_{w,s}^{n+1,m-1} \frac{\partial p_c}{\partial S_w}\Big|_s^{n+1,m-1} (S_{w,P}^{n+1,m-1} - S_{w,S}^{n+1,m-1})\right\}\Delta x\Delta z = \tilde{q}_w^{n+1,m-1}
\end{aligned} \tag{3.50}$$

$$\begin{aligned}\phi \Delta V \frac{S_w{}^{n+1,m} - S_w{}^{n+1,m-1}}{\Delta t} \\ - \left\{T_e^n\left(p_{o,E}^{n+1,m-1} - p_{o,P}^{n+1,m-1}\right) - T_w^n\left(p_{o,P}^{n+1,m-1} - p_{o,W}^{n+1,m-1}\right)\right\}\Delta y \Delta z \\ - \left\{T_n^{n+1,m-1}\left(p_{o,N}^{n+1,m-1} - p_{o,P}^{n+1,m-1} - p_{o,P}^{n}\right)\right. \\ \left. - T_s^n\left(p_{o,P}^{n+1,m-1} - p_{o,S}^{n+1,m-1}\right)\right\}\Delta y \Delta z \\ + \left\{T_{w,e}^{n+1,m-1}\left(p_{c,E}^{n+1,m-1} - p_{c,P}^{n+1,m-1}\right)\right. \\ - T_{w,w}^{n+1,m-1}\left(p_{c,P}^{n+1,m-1} - p_{c,W}^{n+1,m-1}\right)\Big\}\Delta y \Delta z \\ + \left\{T_{w,n}^{n+1,m-1}\left(p_{c,N}^{n+1,m-1} - p_{c,P}^{n+1,m-1}\right) - T_{w,s}^{n}\left(p_{c,P}^{n+1,m-1} - p_{c,S}^{n+1,m-1}\right)\right\}\Delta x \Delta z \\ = \tilde{q}_w^{n+1,m-1}\end{aligned} \tag{3.51}$$

todos los coeficientes de transmisibilidad y el término fuente $\tilde{q}_w$ y sumidero $\tilde{q}_o$, la presión y la saturación, se evalúan en los niveles de tiempo t^n y $t^{n+1,m-1}$, respectivamente.

El intervalo de tiempo $\Delta t_S^{n+1,m}$ en (3.50) y/o (3.51) se elige de la siguiente manera:

$$\left(\frac{\partial S_w^{n+1,m}}{\partial t}\right)_{max} = \max_{i,j,k}\left(\frac{G\left(p_o^n, S_w^{n+1,m-1}\right)}{V\phi}\right)_{i,j} \tag{3.52}$$

y luego se calcula

$$\Delta t_S^{n+1,m} = DS_{max} \Big/ \left(\frac{\partial S_w^{n+1,m}}{\partial t}\right)_{max} \quad m = 1, 2, 3 \ldots M^n, n = 0,1,2,3 \tag{3.53}$$

donde $G\ (p_o^n, S_w^{n+1,m-1})$ denota el lado derecho de la ecuación (3.50) y/o (3.51)

CAPÍTULO IV

COMPARATIVA DE RENDIMIENTO DEL IMPES CLÁSICO VERSUS IMPES MEJORADO

4. Comparativa de Rendimiento del IMPES Clásico Versus IMPES Mejorado

En este capítulo se estudiará el comportamiento de los simuladores desarrollados con las ecuaciones (3.43)−(3.45) y (3.44)−(3.46) , implementados en el lenguaje de programación Python mediante el uso de las bibliotecas numéricas Numpy y *SciPy* y otras bibliotecas auxiliares.

4.1 Implementación Computacional

El problema representado por las ecuaciones discretizadas (3.43)−(3.45) y (3.44)−(3.46) es no lineal y está fuertemente acoplado, por lo tanto, se debe utilizar una estrategia de linealización, para este ejemplo se propone utilizar el algoritmo IMPES *(Implicit Pressure Explicit Saturation).*

4.1.1 Algoritmo General

Este algoritmo de solución contiene principalmente los siguientes pasos:

1. Definir condiciones iniciales y de frontera del problema S_w^0; p_o^0; T_{max}, Δt.
2. *while* t < T_{max} do
    1. Calcular los coeficientes de la ecuación de presión. Estos coeficientes se calculan utilizando los valores de saturación del paso anterior: S_w^0.
    2. Resolver la ecuación de presión de manera implícita: A continuación, se utiliza S_w^0 para evaluar p_o^0.
    3. Calcular los coeficientes de la ecuación de saturación: A continuación, se utiliza S_w^n y p_o^n
    4. Resolver la ecuación de saturación de manera explícita.:
    5. Se Utilizar S_w^n y p_o^n para calcular S_w^{n+1}.
    6. Sumar al tiempo transcurrido t el paso de tiempo t.
3. *end while*

La diferencia con el IMPES Mejorado está en el cálculo del paso de tiempo para la saturación en la Figura 4.1 y la Figura 4.2 se explica esta diferencia entre los algoritmos IMPES Clásico e IMPES Mejorado respectivamente.

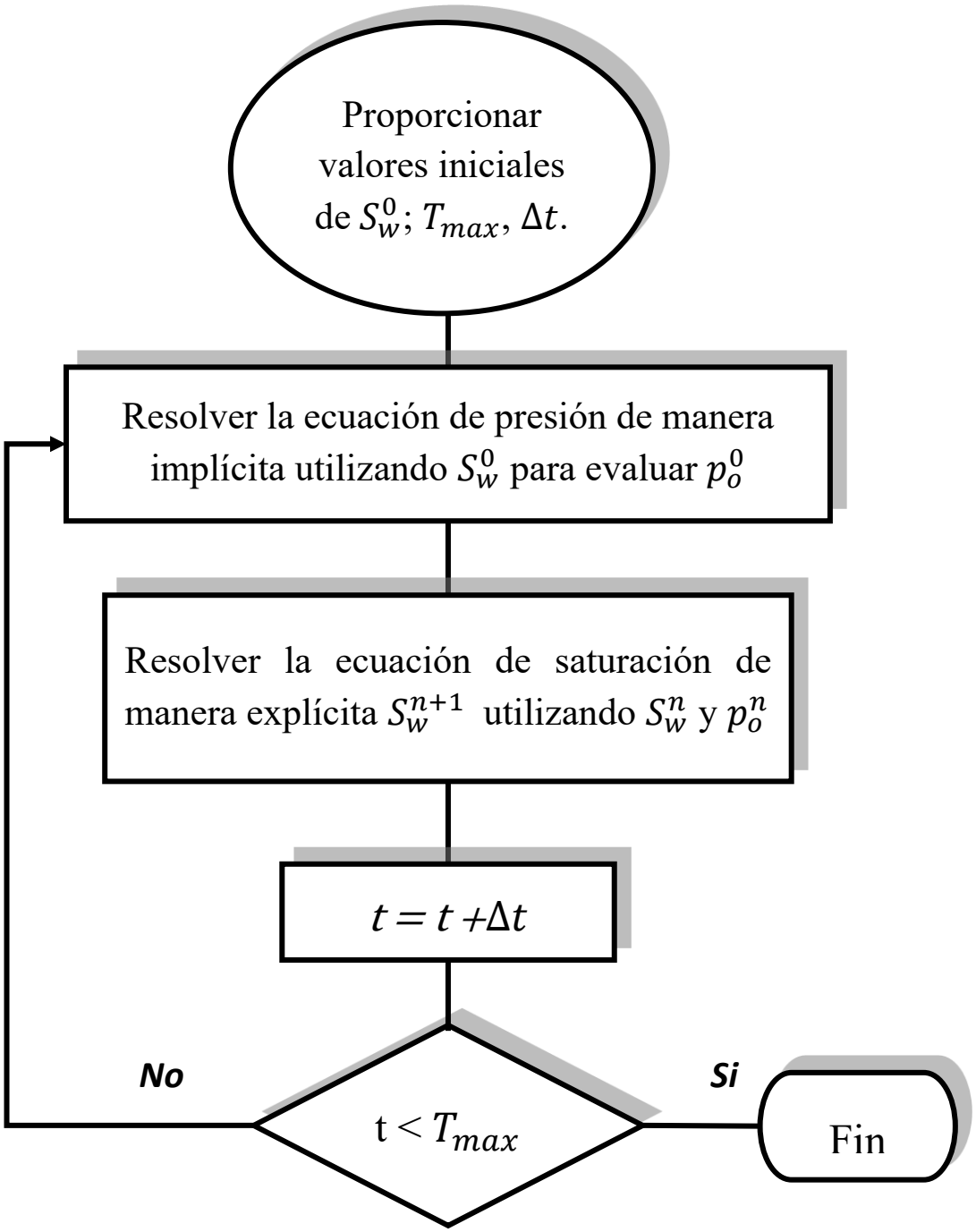


Figura 4.1.Algoritmo IMPES Clásico.

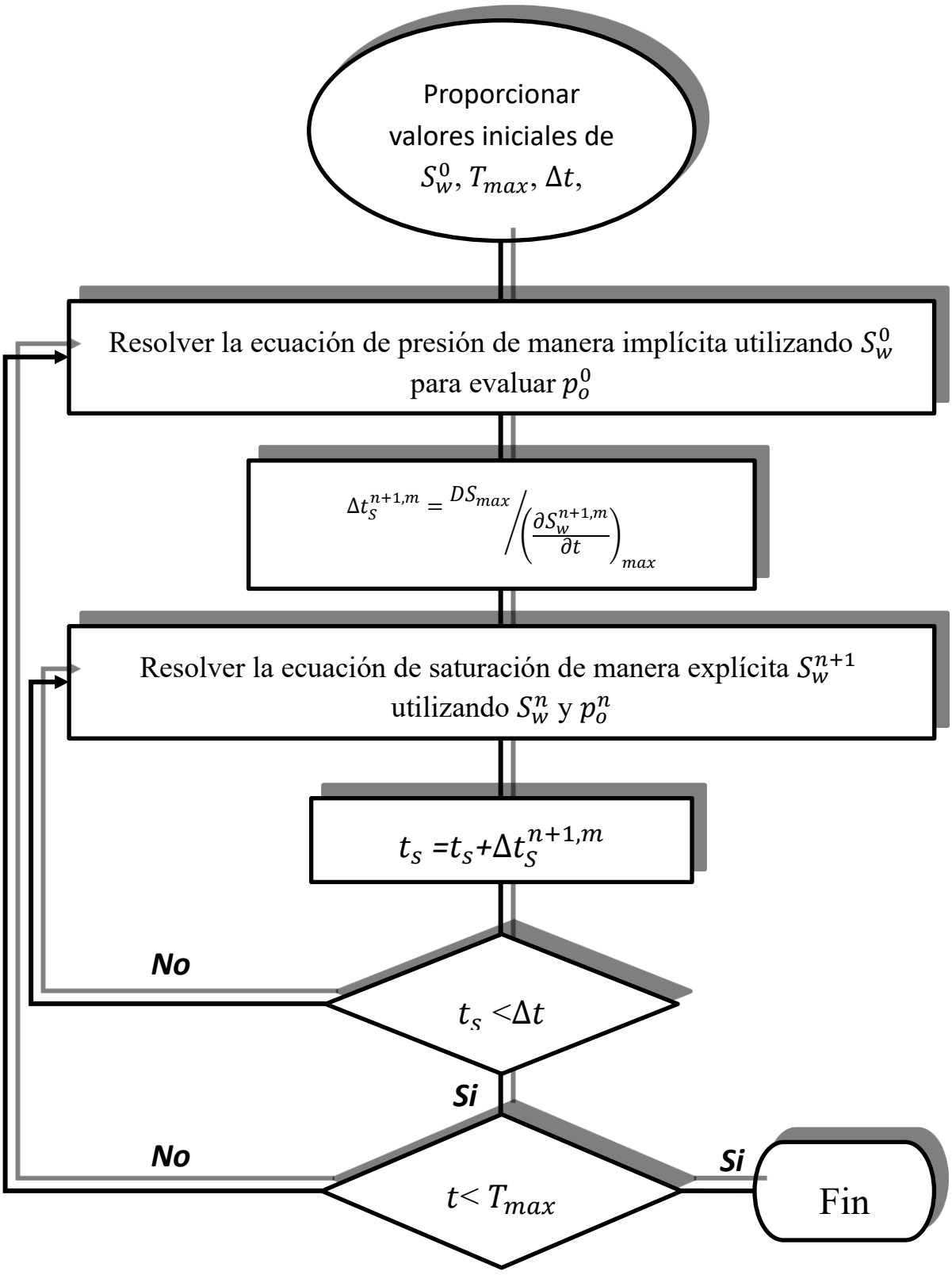


Figura 4.2. Algoritmo IMPES Mejorado.

4.1.2 Implementación del formato de Matrices dispersas CSR

La biblioteca *SciPy* permite dar lectura a matrices en diferentes formatos que evitan el uso de matrices densas. La ventaja de esta forma de implementar los algoritmos es que se evita el problema de colapsar la memoria que los ordenadores pueden llegar a proporcionar. En la sección 2.3.7 se habló del formato CSR y de su estructura. Una manera de codificar este formato para aprovechar la biblioteca *SciPy* es parecida a la estructura del *formato de coordenadas*, solo que respetando el orden por filas que dicta el formato CSR y tomando como referencia la Figura 2.13, el almacenamiento de la matriz dispersa sería el ilustrado por la Figura 4.3. Esta manera de implementar la creación de los vectores V, JV, e IV ayudan a utilizar los comandos *from scipy.sparse import csr_matrix*, que sirven para llamar a la biblioteca integrada de Python, y después generar la matriz dispersa con la función *A=csr_matrix((data, (row, col)), shape=(M, M))* y así poder trabajar con matrices de grandes magnitudes.

$V =$	AP	AE	AN	AW	AP	AE	AW	AP	AE	AW	AP	AE	AW	AP	AE	AS	AW	AP
$JV =$	0	0	0	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5
$IV =$	0	1	5	0	1	2	1	2	2	2	3	4	3	4	5	0	4	5

Figura 4.3 Almacenamiento de Matrices en formato CSR para implementación con biblioteca SciPy.

La construcción del código, que formará parte de una función individual y que será llamada por el código principal, será estructurada a partir de varios ciclos anidados individuales para cada diagonal que conforme nuestro sistema generado por la discretización del yacimiento. Esta toma en consideración los nodos en las diagonales que formen parte de las fronteras y que por su ausencia en estas valgan cero. La implementación computacional se describe con más detalle en el ANEXO D.

4.2 Verificación de los Resultados

4.2.1 Buckley-Leverett

El modelo de Buckley-Leverett describe el desplazamiento de aceite por agua en un dominio horizontal. Este modelo matemático es ampliamente utilizado en la validación de simuladores de flujo bifásico en medios porosos, puesto que cuenta con una solución analítica para el perfil de saturación de agua. En el modelo conceptual de Buckley-Leverett se considera lo siguiente:

1. El desplazamiento ocurre en un medio unidimensional (Figura 4.4).
2. El medio es isotrópico.
3. No se consideran los efectos de la presión capilar, ni de la fuerza de gravedad.
4. No hay fuentes ni sumideros
5. Se inyecta agua a caudal constante por la frontera izquierda del dominio y se producen los fluidos (agua y aceite) por la frontera derecha del dominio a presión constante.
6. Los fluidos son incompresibles e inmiscibles.

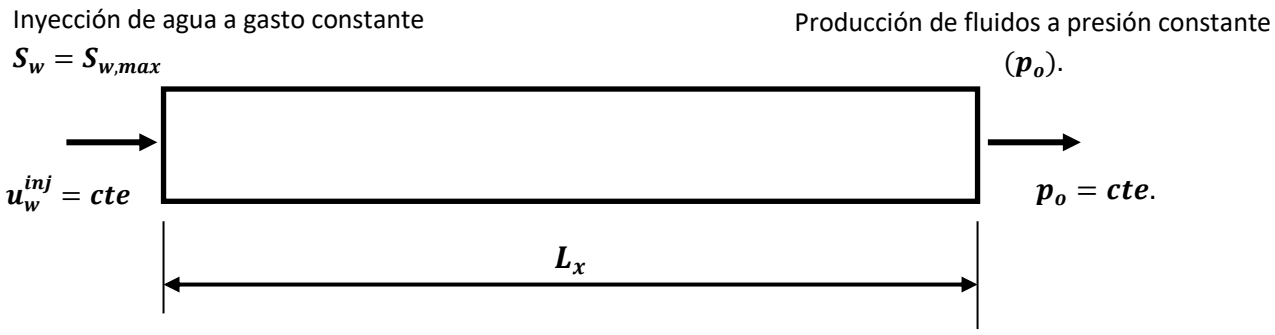


Figura 4.4. Medio Homogéneo de Longitud L_x, inicialmente Saturado con Aceite y Agua.

Tomando en cuenta las consideraciones del modelo conceptual descrito, las ecuaciones (3.30) y (3.34), se reducen a:

$$-\nabla \cdot \left(\bar{\bar{k}}\lambda \nabla p_o\right) = 0 \tag{4.1}$$

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot \left(\bar{\bar{k}} \lambda_w (\nabla p_o) \right) = 0 \tag{4.2}$$

las ecuaciones (4.1) y (4.2) son las ecuaciones con las que se genera el simulador unidimensional. Para verificar la correcta implementación del propio simulador unidimensional, se tomó como referencia los resultados publicados en [33] . Donde se comparan los perfiles de saturación de agua para cada 30 días de simulación, con una malla de 100 bloques para los datos de *Tabla 4.1* y *Tabla 4.2*, que presentan los datos petrofísicos del problema y las curvas de permeabilidad relativa, respectivamente.

Propiedad	***Valor***
Longitud del dominio L_x	1000 [ft]
Permeabilidad absoluta k	100 [mD]
Porosidad ϕ	0.20
Viscosidad del agua μ_w	0.42
Viscosidad del aceite μ_o	15.2
Saturación residual del agua S_{rw}	0.4
Saturación residual del aceite S_{ro}	0.18
Velocidad de inyección u_w^{inj}	2.0E-06 [ft/s]
Presión de produccion p_o^{out}	1000 [psi]

Tabla 4.1-Parámetros para la Simulación del Problema de Buckley-Leverett, adaptada de [33].

S_w	k_{rw}	k_{ro}	S_w	k_{rw}	k_{ro}
0.4	0.0000	1.0000	0.62	0.0605	0.2025
0.42	0.0005	0.9025	0.64	0.0720	0.1600
0.44	0.0020	0.8100	0.66	0.0845	0.1225
0.46	0.0045	0.7225	0.68	0.0980	0.0900
0.48	0.0080	0.6400	0.70	0.1125	0.0625
0.50	0.0125	0.5625	0.72	0.1280	0.0400
0.52	0.0180	0.4900	0.74	0.1445	0.0225
0.54	0.0245	0.4225	0.76	0.1620	0.0100
0.56	0.0320	0.3600	0.78	0.1805	0.0025
0.58	0.0405	0.3025	0.80	0.0200	0.0000
0.60	0.0500	0.2500	0.82	0.4500	0.0000

Tabla 4.2. Permeabilidades Relativas k_{rw} y k_{ro}, adaptada de [33].

En la Figura 4.5 y la Figura 4.6 se muestran los perfiles de Saturación S_w y Presión p_o para una malla de un tamaño de 100 x100 celdas reportados cada 30 días y hasta para un total de 120 días de simulación, y obtenidas de [33], las cuales se comparan con la Figura 4.7 y la Figura 4.8, que se obtuvieron con el simulador propio y con las mismas condiciones.

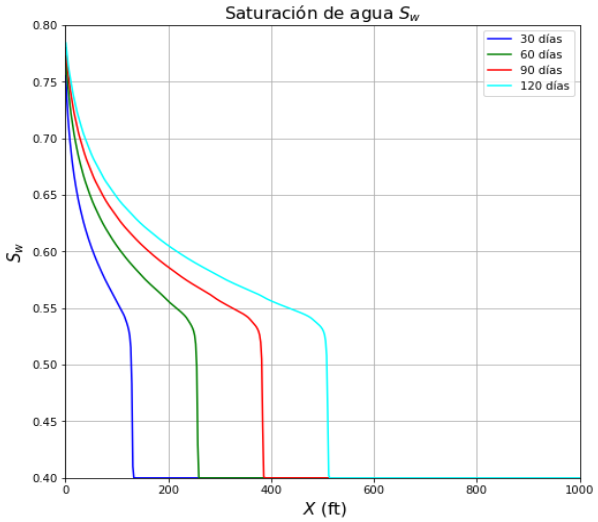


Figura 4.5 Perfiles de SaturaciónS_w, tomada de [33].

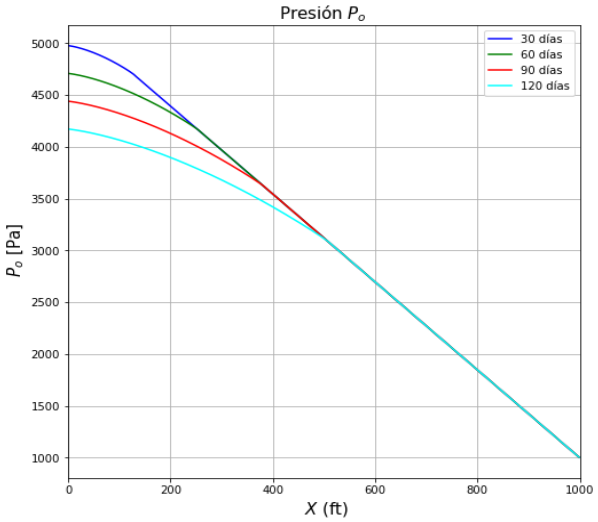


Figura 4.6 Perfiles de Presión p_o, tomada de [33].

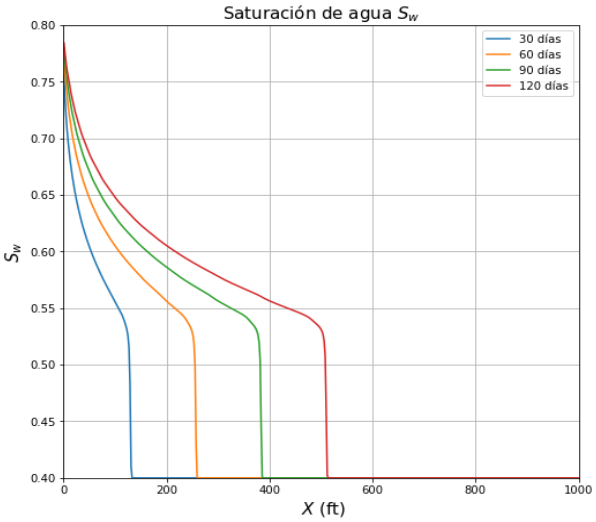


Figura 4.7 *Perfiles de Saturación S_w.*

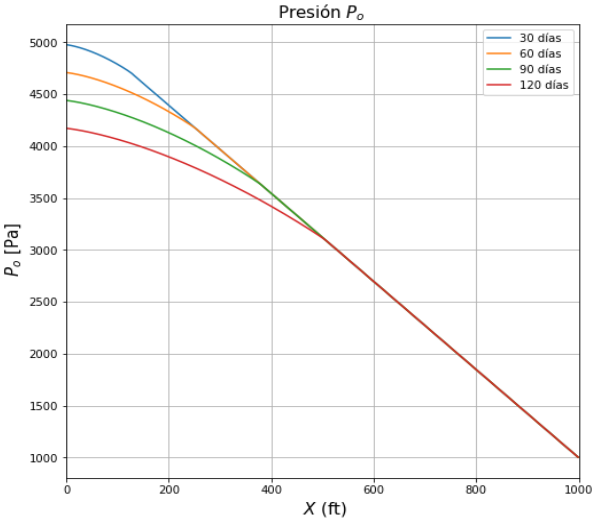


Figura 4.8 Perfiles de Presión p_o.

4.2.2 Problema de los 5 Pozos

El modelo simétrico conocido como *"five spot"* o de los cinco pozos, el cual detalla el proceso de desplazamiento de aceite por agua en un medio poroso, en el que existe un pozo inyector y un pozo productor en las esquinas opuestas del dominio. La forma en que se colocan los pozos emula la simetría de colocar 1 pozo inyector y cuatro pozos productores equidistantes al pozo inyector. Para este modelo conceptual se toman las siguientes consideraciones:

4. El desplazamiento ocurre en un medio bidimensional.
5. Se consideran los efectos de la presión capilar.
6. No se consideran los efectos de la gravedad.
7. El medio es isotrópico.
8. Existen fuentes (Pozo inyector) y sumideros (Pozo productor).

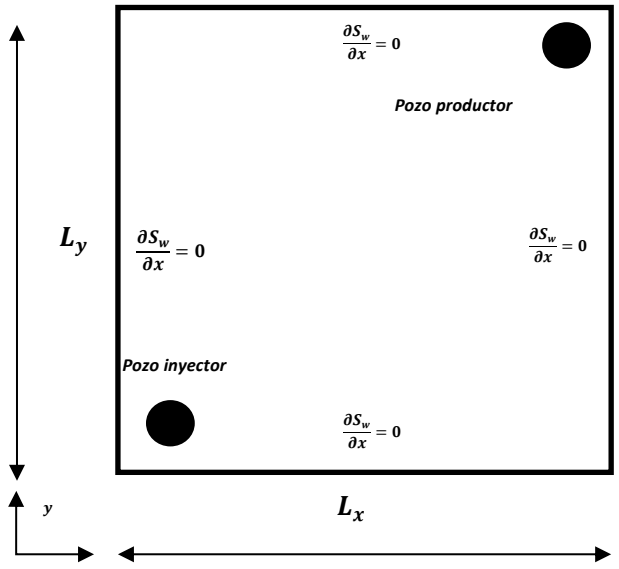


Figura 4.9. Dominio físico del modelo "five spot", adaptada de [33].

Para verificar la implementación del código numérico bidimensional se toma como referencia este modelo simétrico, los parámetros para llevar a cabo la simulación numérica se muestran en la Tabla 4.3.

Tabla 4.3.Parámetros Para la Simulación del Problema de los 5 pozos

Propiedad	**Valor**
Longitud del dominio $L_x \times L_y \times L_z$	1000 x 1000x100 [ft]
Permeabilidad absoluta k	100 [mD]
Porosidad ϕ	0.20
Viscosidad del agua μ_w	0.96 [cP]
Viscosidad del aceite μ_o	1.14 [cP]
Saturación residual del agua S_{rw}	0.22
Saturación residual del aceite S_{ro}	0.20
Posición de pozo inyector	50,50 [ft]
Posición de pozo productor	950, 950 [ft]
Presión de Inyección en el fondo del pozo p_{wb}^{inj}	3700 [psi]
Presión de produccion en el fondo del pozo p_{wb}^{out}	3500 [psi]

adaptada de [33].

Las permeabilidades relativas, están definidas por:

$$k_{rw} = k_{rw,max}\left(\frac{S_w - S_{rw}}{1 - S_{ro} - S_{rw}}\right)^2 \tag{4.3}$$

$$k_{ro} = \left(\frac{S_w - S_{rw}}{1 - S_{ro} - S_{rw}}\right)^2 \tag{4.4}$$

donde: $k_{rw,max} = 0.65$, representa el valor máximo de la permeabilidad relativa de la fase agua, $S_{rw} = 0.22$, se refiere al valor de la saturación de agua residual y $S_{ro} = 0.2$, es el valor de la saturación de aceite residual. La presión capilar generalmente es función de la saturación de la fase mojante. En la literatura existen muchas relaciones empíricas obtenidas de experimentos de laboratorio, para este trabajo se utiliza la siguiente expresión:

$$p_c(S_w) = p_{cow,min} - \hat{B}\, ln\left(\frac{S_w - S_{rw}}{1 - S_{rw}}\right) \tag{4.5}$$

tal que $\hat{B}$ es una constante cuyo valor es igual a -10.1335 tal que $p_{cow,min} = p_{cow,max}$ cuando $S_w - S_{rw}$ donde $p_{cow,min} = 0\ y\ p_{cow,max} = 70$ [psi].

Las derivadas fueron programadas a partir de la derivada analítica de la ecuación (4.5) para fines prácticos.

La Figura 4.10 muestra a los perfiles de presión p_o para 8000 días de simulación para tiempos específicos y un tamaño de malla de 90×90 celdas. El comportamiento de la presión se ve afectado por la presión capilar, lo que se traduce al hecho de que los perfiles ya no se muestran estáticos como lo sería en un entorno donde se desprecia la presión capilar y estos cambian. Por ejemplo, a los 500 días de simulación se puede apreciar que las isobaras de presión cambian de posición a través del dominio a medida que la inyección se mantiene constante, lo que se puede visualizar en la Figura 4.11, que nos muestra los perfiles de Saturación del agua S_w. Sí se observa el perfil de saturación para 500 días de simulación, se nota que el agua ya ha barrido aproximadamente una tercera parte del dominio. Para el perfil de 1,000 de simulación, se aprecia que el agua ya cubrió casi todo el dominio y está por llegar al pozo productor. Los perfiles correspondientes a 1,500 y 2,000 días, indican que el agua se está produciendo y prácticamente ya se ha barrido más de 60% del aceite. Por último, los perfiles para 4,000 y 8,000 días de simulación muestran que el agua ha cubierto por completo el dominio y el aceite casi ha sido producido completamente. En la Figura 4.12, se puede hacer una descripción análoga para un medio anisotrópico donde la permeabilidad del medio varía de entre los 10 [mD] a los 100 [mD].

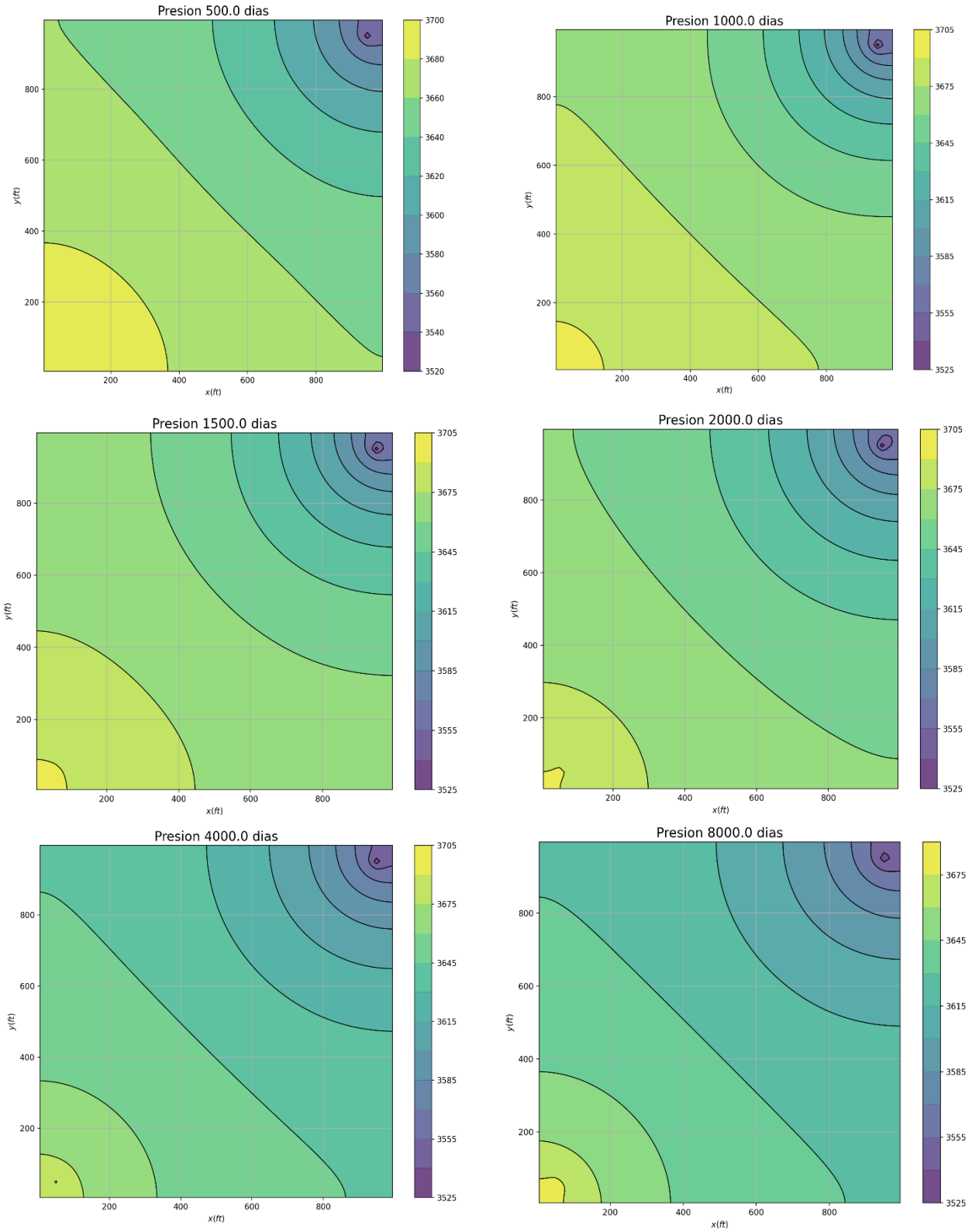


Figura 4.10. Perfiles de Presión de la fase aceite p_o (psi) para un tamaño de malla de 90x90 celdas.

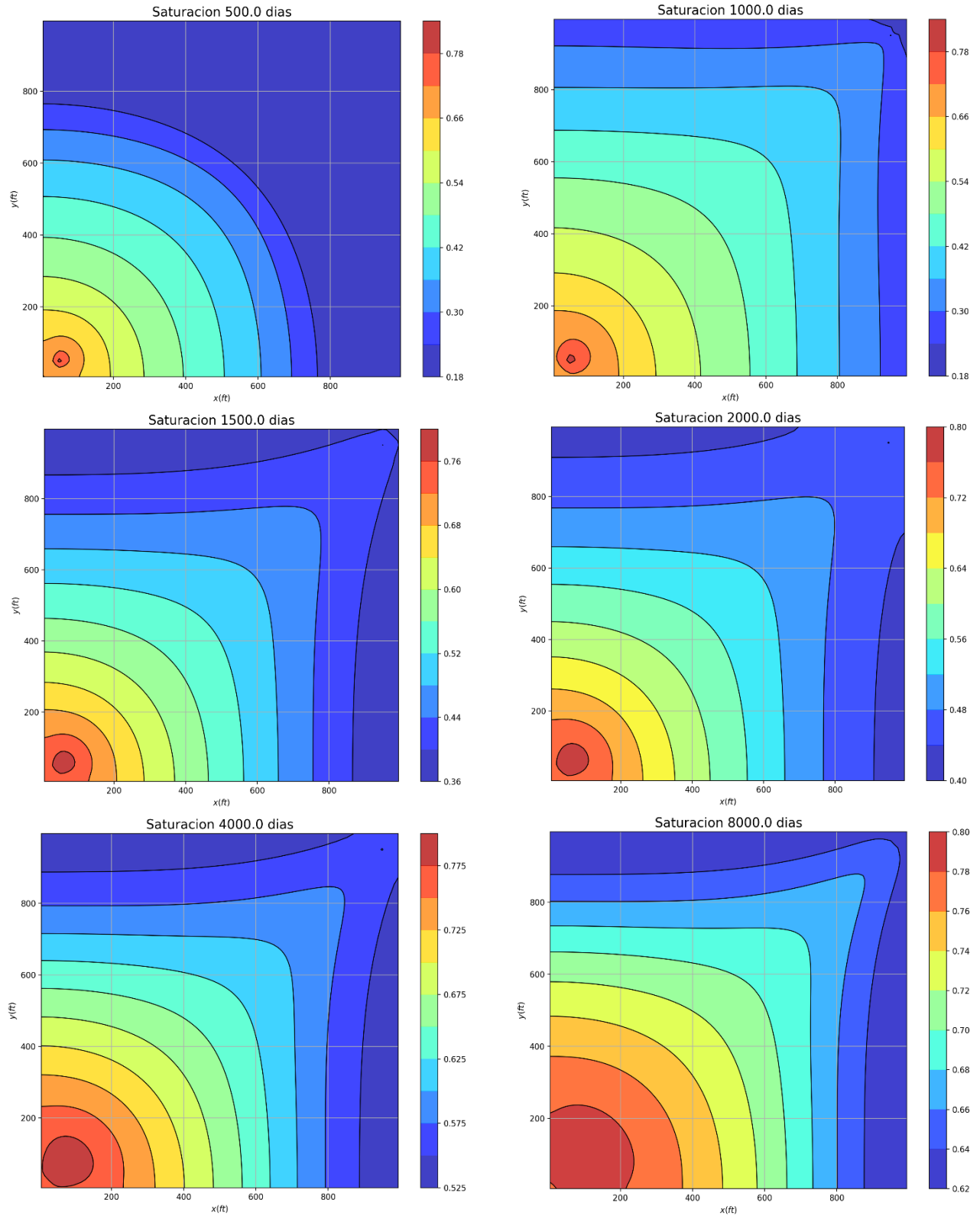


Figura 4.11. Perfiles de Saturación S_w para un tamaño de malla de 90x90 celdas.

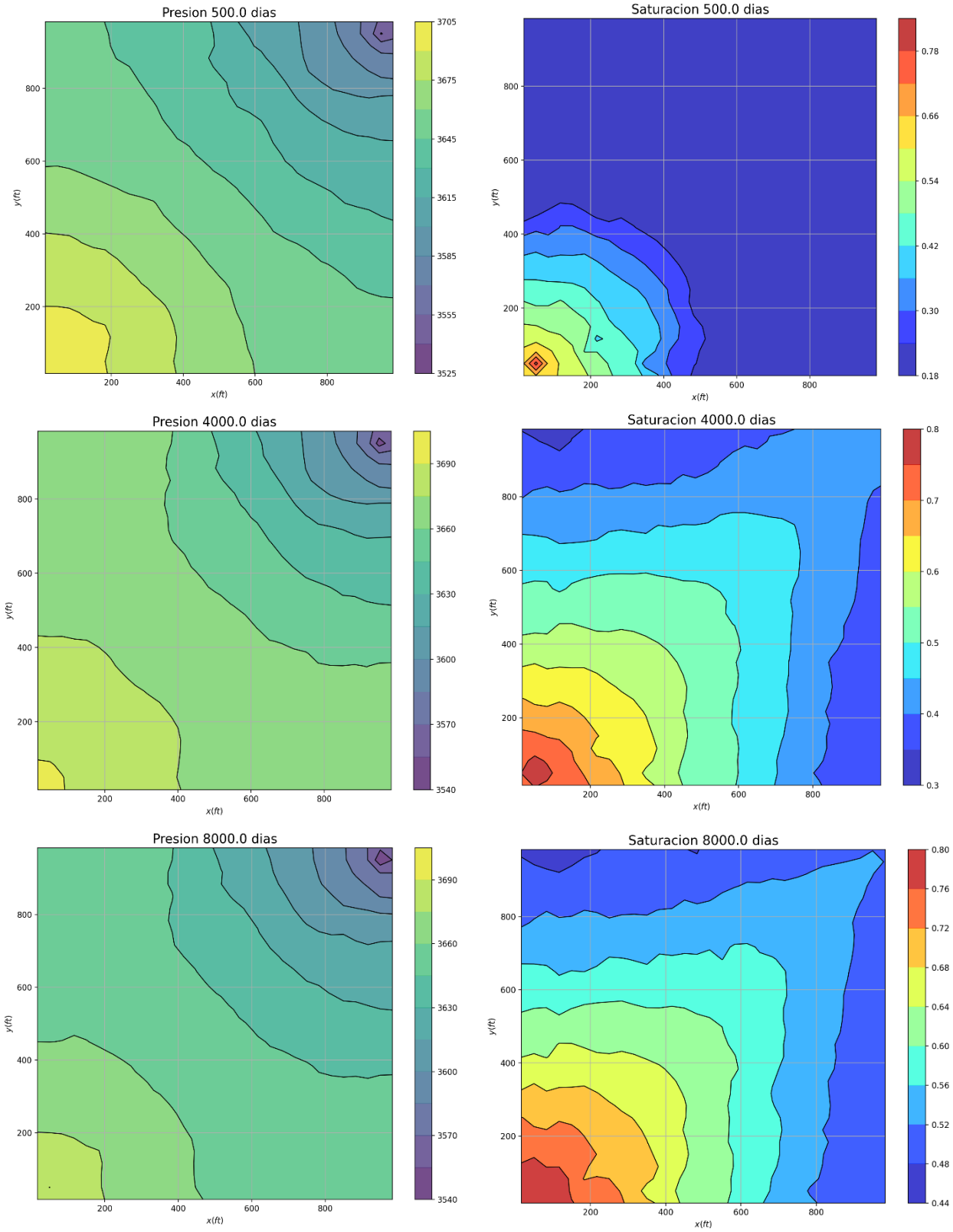


Figura 4.12 Perfiles de Presión p_o y Saturación S_w en un medio Anisotrópico para una malla de 90 x 90 celdas.

En la Figura 4.13 y en la Figura 4.14 se muestra el campo de velocidad normalizado para las fases del aceite, representado con flechas rojas, y del agua, representado con flechas azules, en los medios isotrópico y anisotrópico, respectivamente. Los tiempos de simulación mostrados son los mismos que para la saturación y presión, tomando en cuenta ambas figuras, se nota que el comportamiento corresponde a lo descrito para el perfil de saturación y presión, en ambos casos. Por ejemplo, en el caso isotrópico para 500 días de simulación se muestra el perfil de saturación que corresponde al campo de velocidad del agua de 500 días, indicando que el agua ha avanzado hasta casi la mitad del dominio. El perfil de presión del aceite a los 500 días muestra que su mayor valor es en el pozo inyector, coincidiendo con su perfil de velocidad, debido a que gran parte del aceite ha sido desplazado, presentando una mayor movilidad por parte del aceite, y, por lo tanto, mayor velocidad en el pozo productor y en toda su vecindad. Para 500 días de simulación, en el caso anisotrópico, de igual manera la saturación y la velocidad de la fase agua son correspondientes, la distribución del avance es discontinuo, sin embargo, no supera los 800 metros. Entre tanto, el perfil de presión del aceite, corresponde con su perfil de velocidad, siendo mayor en el pozo productor y su vecindad, aunque también mostrando que su distribución es irregular. Estos perfiles se complementan con los perfiles de las líneas de corriente de ambas fases para los mismos medios y utilizando los mismos colores en las trazas que en los campos de velocidad. En la Figura 4.15 se muestra el caso isotrópico, y se observa que las líneas son continuas y curvas para el flujo de aceite, para el flujo de agua también son continuas, pero más rectas. La Figura 4.16 muestra las líneas de corriente para el caso anisotrópico, en tal caso se aprecia que las líneas son correspondientes a las del caso isotrópico, pero son más irregulares. Es importante mencionar que los perfiles de las líneas de flujo de corriente en esta tesis se obtuvieron mediante la biblioteca integrada de Python *Matplotlib*. Una línea de corriente es una trayectoria que en cada uno de sus puntos es tangente al campo de velocidad en un instante de tiempo dado [34], las líneas se trazan usando los campos de velocidades instantáneos. Su utilidad se debe a que en una simulación numérica se busca entender y cuantificar el impacto de los elementos desconocidos del modelo estático sobre el flujo de los fluidos y el transporte de hidrocarburos. Conjuntamente, la física del desplazamiento permite modelar flujos en yacimientos donde existen muchas heterogeneidades, con un enfoque basado en la geometría y la densidad, de las líneas de corriente y los perfiles de velocidad, en virtud de que reflejan el impacto geológico sobre los caminos preferenciales del flujo, introduciendo mayor densidad de líneas en regiones de alta porosidad y permeabilidad. Las líneas de corriente y los campos de velocidad proveen imágenes instantáneas de los patrones de flujo en el campo. Los ingenieros las utilizan para acortar el ciclo de ajuste histórico, al validar los modelos de los yacimientos, desarrollar estrategias de inyección y mejorar la eficiencia del desplazamiento, mediante el análisis de los patrones de flujo y, estimando las relaciones inyección-producción en el campo, durante los cálculos de distintos escenarios de predicción. La evolución de los frentes de inyección y su interacción con las heterogeneidades del yacimiento, pueden observarse claramente con estos perfiles, brindando una manera natural e intuitiva para caracterizar dinámicamente un yacimiento [17, 35, 36, 37].

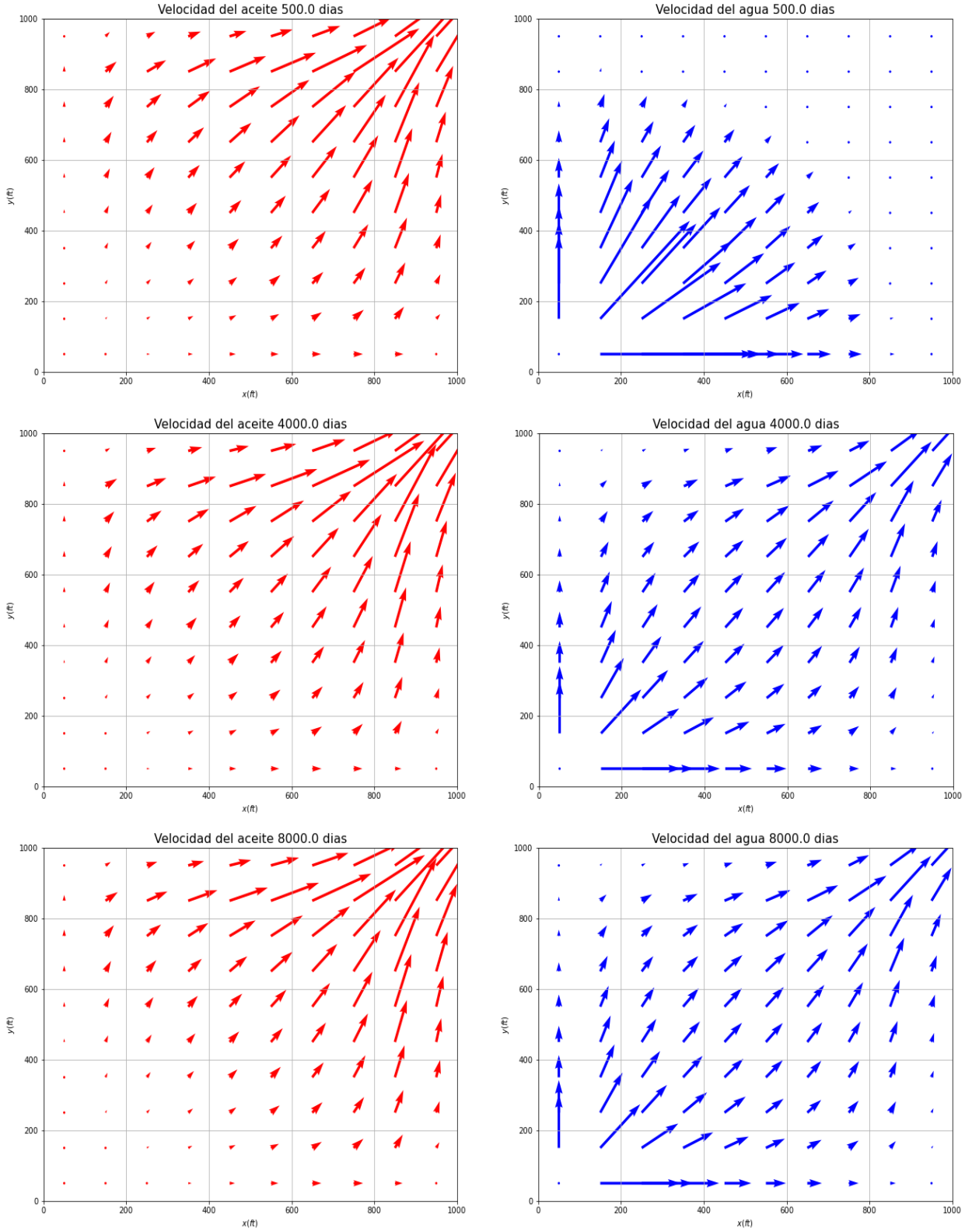


Figura 4.13 Perfiles de Velocidad de las Fases Aceite y Agua en un Medio Isotrópico.

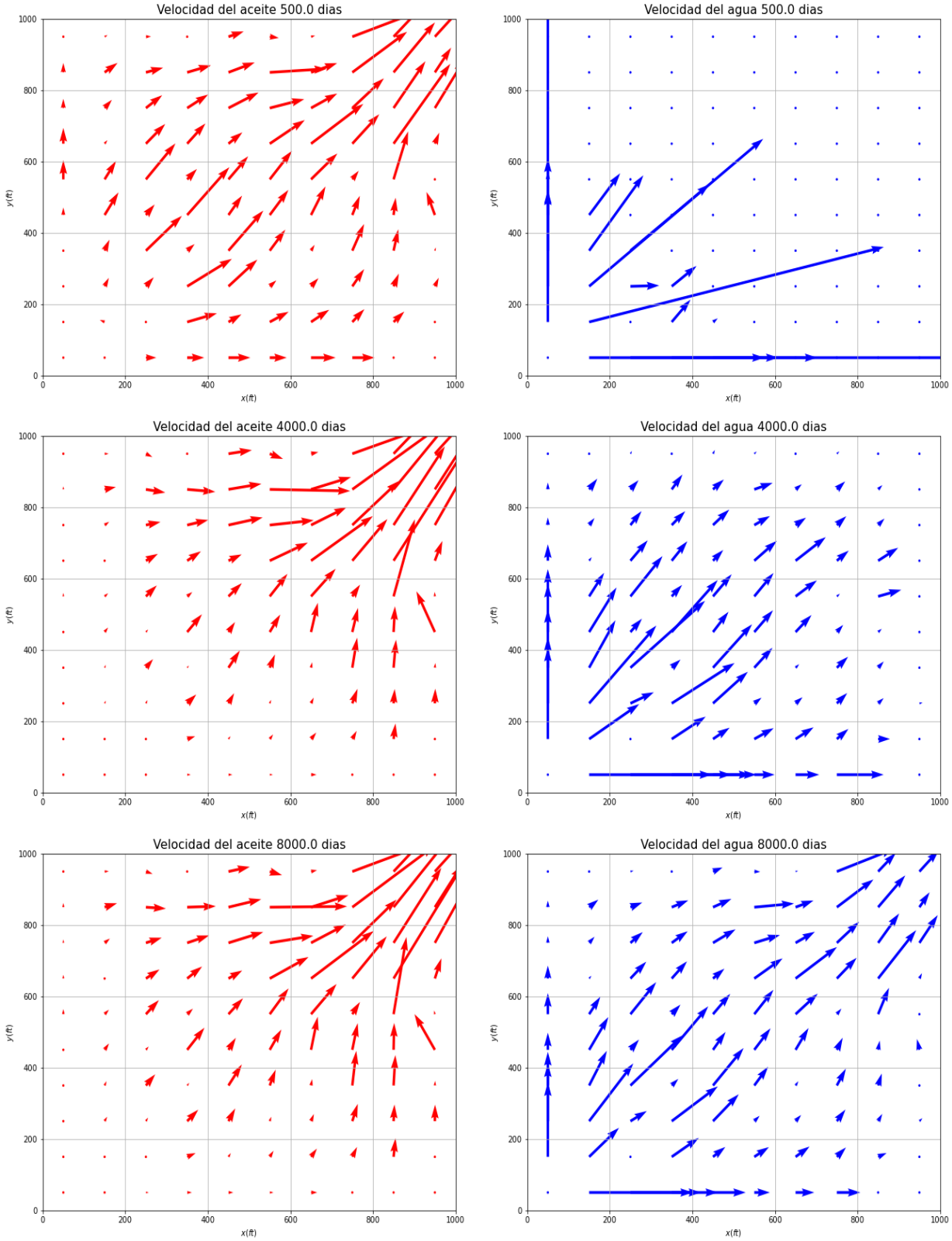


Figura 4.14 Perfiles de Velocidad de las Fases Aceite y Agua en un Medio Anisotrópico.

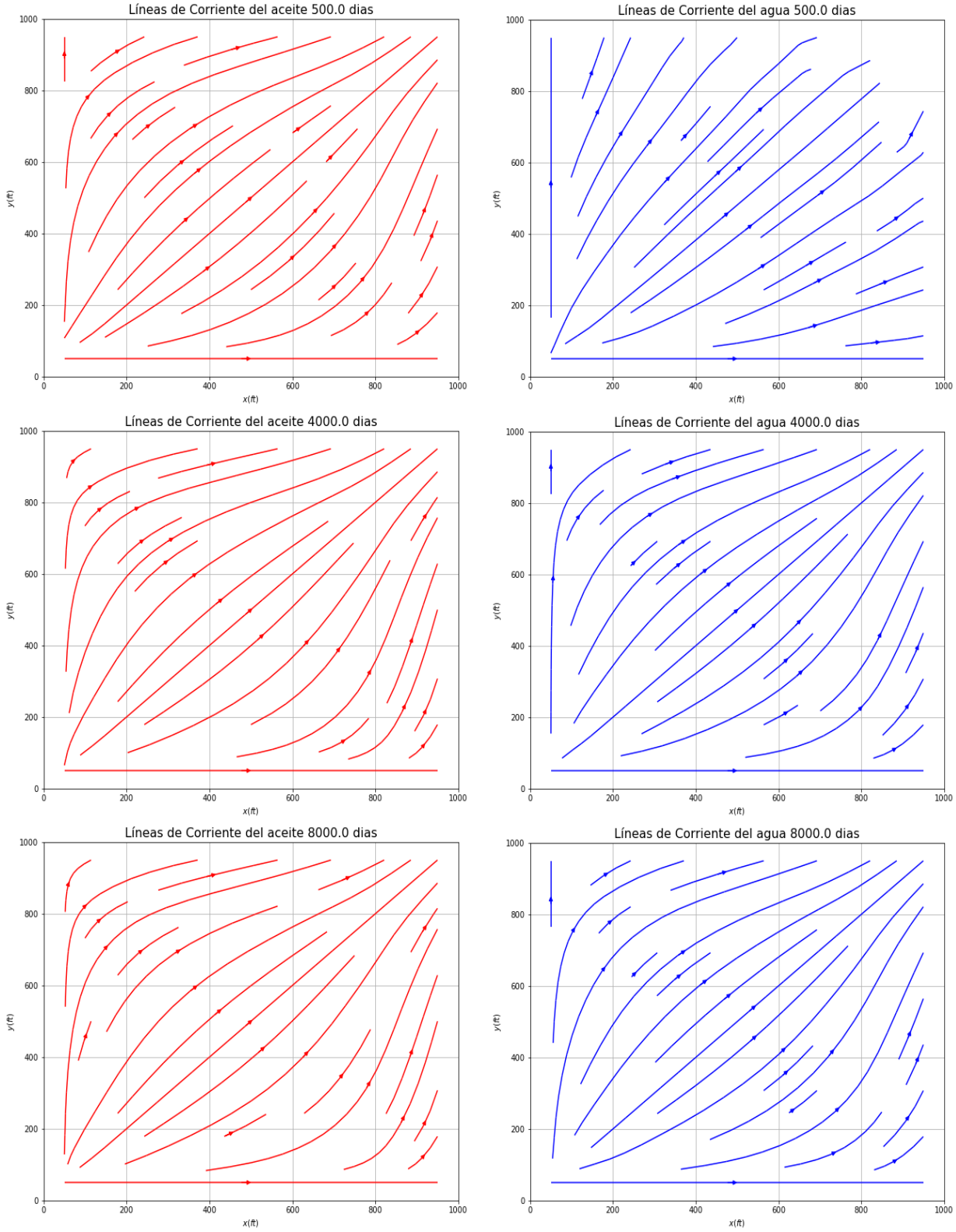


Figura 4.15 Perfiles de Líneas de Corriente de las Fases Aceite y Agua en un Medio Isotrópico.

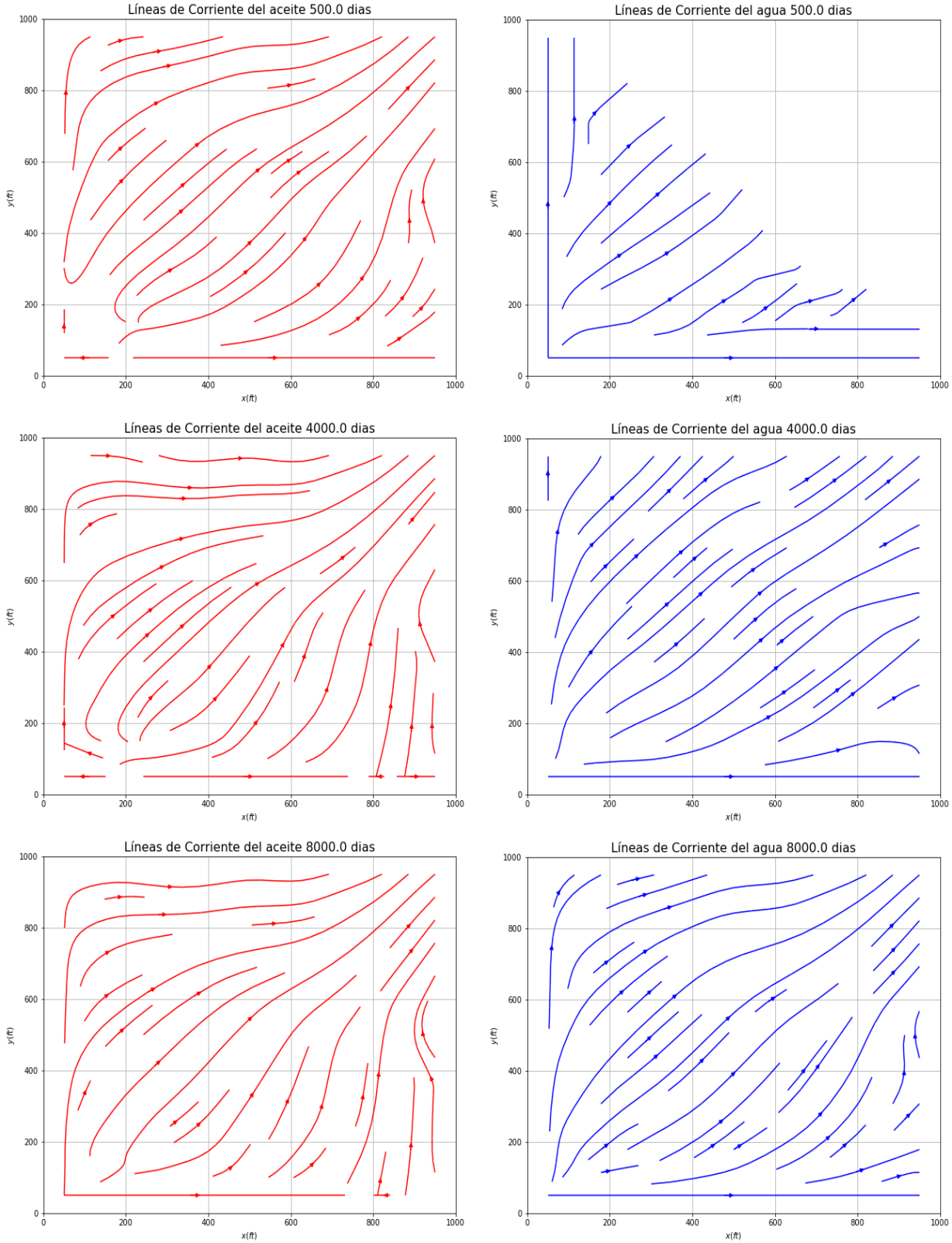
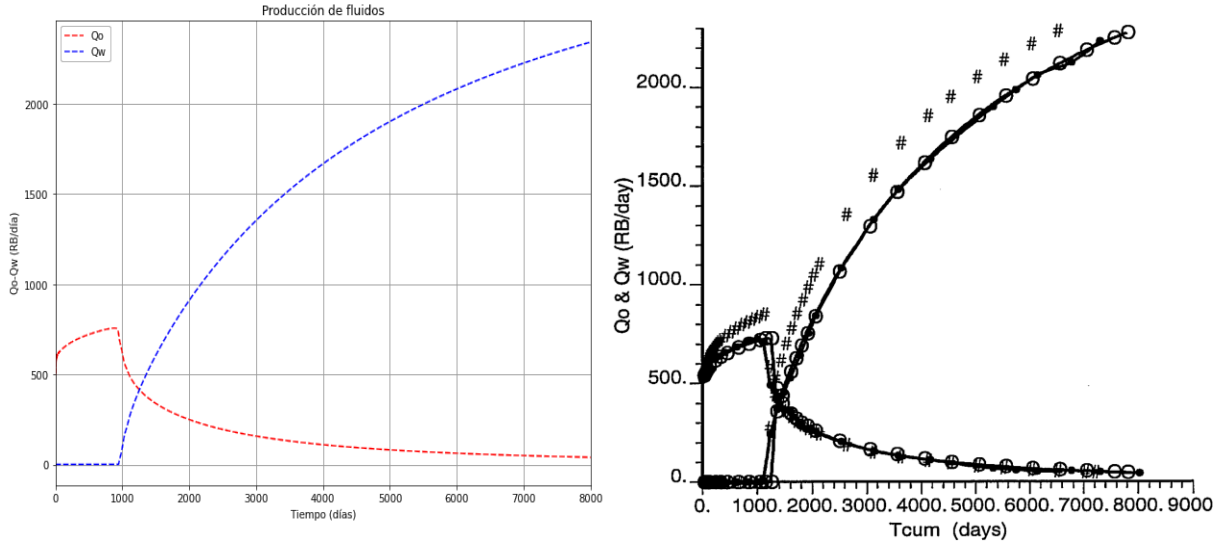


Figura 4.16 Perfiles de Líneas de Corriente de las Fases Aceite y Agua en un Medio Anisotrópico.

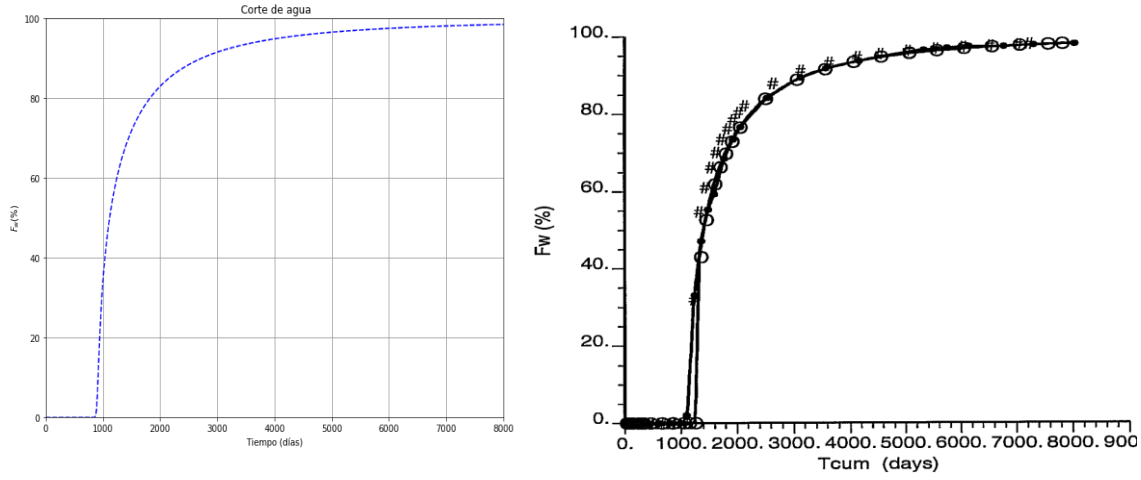
Para verificar la correcta implementación del propio simulador Bidimensional se plantea el ejercicio simétrico de los 5 pozos reportado por Chen [2]. Se comparan los perfiles de producción de fluidos y corte de agua para 8000 días de simulación, con una malla de 91 x 91 bloques. La Figura 4.17a) muestra los resultados obtenidos con el propio simulador y la Figura 4.17b) muestra los resultados de [2], de donde se observa que ambas gráficas son cualitativamente similares. La Figura 4.17 muestra la producción de agua y aceite durante todo el proceso de simulación.



a) b)

Figura 4.17 Producción de Fluidos.

En la Figura 4.18, se aprecia que el corte de agua ocurre pasados los 1,000 días de simulación lo que coincide con los perfiles de presión y saturación analizados anteriormente. Esta proeza ayuda a la comparativa a concluir que el modelo numérico bidimensional esta implementado correctamente. La Figura 4.18 a) muestra los resultados propios y la Figura 4.18b) muestra los resultados de [2].



a) b)

Figura 4.18 Flujo Fraccional.

4.3 Resultados de la Comparación de la Simulación Numérica

4.3.1 Comparativa de Rendimiento

En esta sección se realizan las comparativas de rendimiento en cuanto a los solvers utilizados, diferentes tamaños de mallas y diferentes pasos de tiempo para un total de 1500 días de simulación; debido a que la verificación de la correcta implementación del algoritmo ya se ha realizado, no es necesario correr el código hasta los 8000 días. El objetivo de hacer estas comparativas está enfocado en observar el comportamiento de la estabilidad, convergencia y eficiencia de cómputo de tanto de los algoritmos como de los solvers, pues al tratarse de métodos directos e iterativos, la utilización de matrices dispersas densas o vacías influyen directamente en el comportamiento del tiempo de trabajo. La variación del tamaño de la malla no solo afecta de manera directa el tiempo de cómputo, sino que tiene una relación directa con la estabilidad de los métodos, ya que ésta puede generar alteraciones en la precisión y generar un error local en cualquier etapa del cálculo que pueden romper la estabilidad si las perturbaciones que genera se propagan a etapas posteriores del cálculo generando errores más grandes y afectando también de manera indirecta la convergencia.

4.3.1.1 Modelo Unidimensional: Medio Isotrópico

El análisis comienza con el problema unidimensional en un medio isotrópico, variando ligeramente el tamaño de las mallas y utilizando el algoritmo de Thomas como solver fijo, realizando una comparativa en el rendimiento del solver, donde la comparativa en la eficiencia se realiza con un paso de tiempo general (mismo para presión y saturación) de 0.25 días para el IMPES Clásico y de 1 día, solo para la presión para el IMPES Mejorado, puesto que el paso de tiempo para la saturación se calcula mediante la condición presentada en la sección 3.7. Se pueden observar tiempos de ejecución del solver ligeramente más cortos para el IMPES Mejorado y también un mejor comportamiento para mallas más finas, aunque los tiempos de ejecución del solver tienen el mismo orden de magnitud como es de esperarse. También, es importante señalar que el IMPES Clásico con pasos de tiempo de un día el algoritmo comienza a oscilar y se vuelve inestable. La Tabla 4.4 muestra que el IMPES Clásico ejecuta cuatrocientos ochenta veces el algoritmo de Thomas, mientras que en la Tabla 4.5 muestra que lo hace ciento veinte veces, entonces se puede inferir que el IMPES Mejorado es cuatro veces más rápido, lo que con cierta justicia se le atribuye a su paso de tiempo que es también cuatro veces más grande; pero su estabilidad mayor en cuanto se suponga que en general el algoritmo IMPES es más inestable con pasos de tiempo cada vez más grandes. A continuación, se interpreta en la Tabla 4.6 y en la Tabla 4.7 el rendimiento del algoritmo, donde por las razones anteriormente planteadas se observa que no importa que el IMPES Mejorado resuelve más veces la ecuación de saturación que el IMPES Clásico, este se comporta de manera más eficiente con tiempos de ejecución notoriamente más cortos que el IMPES Clásico.

Tiempo Promedio de Ejecución del Solver mediante el IMPES Clásico en [s]			Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	**Ejec.**	
100	**0.00186**	**480**	**0.25**
300	**0.00210**	**480**	
500	**0.00226**	**480**	

Tabla 4.4 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Isotrópico.

Tiempo Promedio de Ejecución del Solver mediante el IMPES Mejorado en [s]			Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	**Ejec.**	
100	**0.00185**	**120**	**1**
300	**0.00235**	**120**	
500	**0.00208**	**120**	

Tabla 4.5 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Isotrópico.

Tiempo Total de Ejecución del Algoritmo IMPES Clásico en [s]		Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	
100	**50.776**	**0.25**
300	**138.163**	
500	**199.638**	

Tabla 4.6 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Isotrópico.

Tiempo Total de Ejecución del Algoritmo IMPES Mejorado en [s]		Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	
100	**25.265**	**1**
300	**54.273**	
500	**58.496**	

Tabla 4.7 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Isotrópico.

En la Tabla 4.8 se puede observar como para mallas más grandes el IMPES Mejorado se acelera considerablemente, ya que la estrecha relación entre, el paso de tiempo y las veces que se calcula la saturación en el IMPES Mejorado, se pueden apreciar indirectamente mediante la razón de cambio de los tiempos de ejecución. Se puede apreciar que se desempeña considerablemente con mallas más finas, como es la de 500 nodos, puesto que es casi tres veces más rápido, y en comparación con las mallas de 100 y 300 nodos, en la Tabla 4.6 , se aprecia que el tiempo de ejecución es casi del doble para el IMPES Clásico, este evento se ve reflejado en la Tabla 4.8, en donde dicha aceleración es casi del doble para ambos casos. El comportamiento del desempeño mostrado en la Tabla 4.8 podría ser un indicativo cuantitativo atribuido a la naturaleza del algoritmo de Thomas, pues al tratarse de un método directo, su tiempo de ejecución dependerá del tamaño de los nodos en la malla y por consiguiente en el número de elementos que formarán la matriz tridiagonal, por lo que se espera que su tiempo de ejecución, incremente de manera proporcional al incremento en el número de nodos. Esta característica se puede observar en la Figura 4.19, la cual describe los tiempos de ejecución del algoritmo de Thomas para ambos algoritmos, donde si se presta atención a las mallas pequeñas no cabe mucha diferencia entre sus valores, aunque conforme aumenta el tamaño de la malla, las curvas si se alejan entre sí con mayor evidencia, en tal caso este acontecimiento se puede asociar a la diferencia de estabilidad que se suscita entre los algoritmos IMPES Clásico e IMPES Mejorado. La Figura 4.20 muestra que las curvas del tiempo de ejecución de los algoritmos IMPES Clásico e IMPES Mejorado son cualitativamente similares, se puede apreciar que la línea punteada en color azul correspondiente al IMPES Clásico se encuentra por encima de la línea punteada color rojo correspondiente al IMPES Mejorado claramente por tener valores más altos, pero sus tendencias se mantienen ajustadamente iguales y correspondientes, lo que tiene sentido si se asemeja que el paso de tiempo es más grande para el IMPES Mejorado, por lo que se puede corroborar que la estabilidad de los métodos recae en el tamaño del paso de tiempo y el tiempo de ejecución dependerá estrictamente del tamaño de la malla.

Cociente de Aceleración de los Algoritmos IMPES Clásico e IMPES Mejorado

Tamaño de malla	Algoritmo de Thomas
100	**1.9815**
300	**2.545**
500	**3.412**

Tabla 4.8. Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Isotrópico.

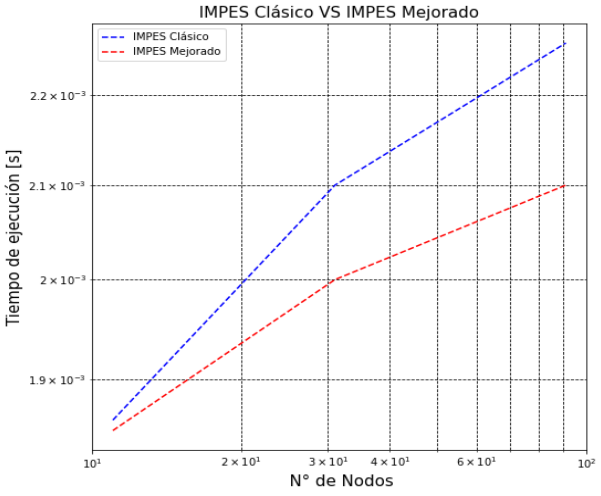


Figura 4.19 Gráfica del Rendimiento en el Tiempo de Ejecución del Solver entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico.

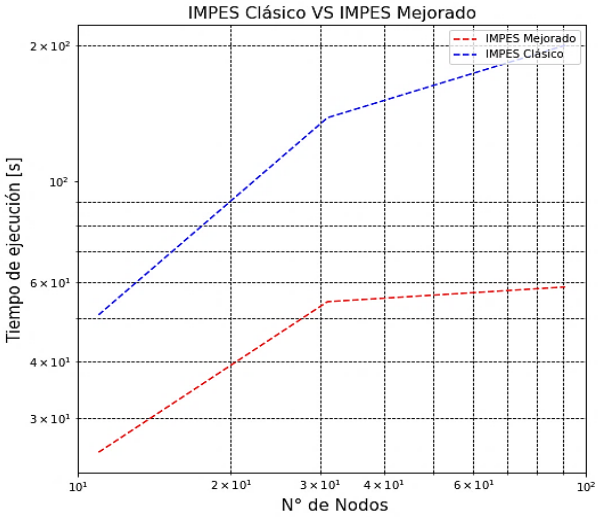


Figura 4.20 Gráfica del Rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico.

4.3.1.1 Modelo Unidimensional: Medio Anisotrópico

En esta sección se presentan los resultados del tiempo de ejecución obtenidos considerando un medio anisotrópico (permeabilidad variable en todo el eje x), primero se comparan los resultados de los tiempos promedio de ejecución del solver del algoritmo de Thomas para los IMPES Clásico e IMPES Mejorado registrados en la Tabla 4.9 y en la Tabla 4.10 respectivamente, se nota que numéricamente los resultados son casi idénticos a los mostrados en la Tabla 4.4 y en la Tabla 4.5 correspondientemente, pese a que cuantitativamente sus magnitudes son del mismo orden, difieren aproximadamente solo por algunas cienmilésimas, siendo más pequeñas para el medio anisotrópico.

Observando la Tabla 4.11 y la Tabla 4.12, se visualiza el mismo comportamiento presentado en la sección anterior, en los tiempos de ejecución de los algoritmos, para un paso de tiempo cuatro veces más grande en el IMPES Mejorado, a saber, para la malla de 500 nodos, el tiempo de ejecución es cuatro veces menor que para el IMPES Clásico, mientras que, para la malla de 100 y 200 nodos, se puede contemplar que es alrededor del doble para el IMPES Clásico. De la Tabla 4.13 se puede apreciar que el IMPES Mejorado es al menos dos veces más rápido que el IMPES Clásico.

Prácticamente, el comportamiento del solver y de los algoritmos muestran similitud con el caso isotrópico, además en la Figura 4.21 y en la Figura 4.22 se ve que las tendencias de las curvas son parecidas, y también coinciden en el hecho de que el algoritmo se acelera más con mallas muy finas.

Gráficamente, la Figura 4.19, Figura 4.20, Figura 4.21 y la Figura 4.22 ilustran que la tendencia del tiempo de simulación en mallas unidimensionales uniformemente espaciadas son cualitativamente similares, sin importar la anisotropía del medio.

Tiempo Promedio de Ejecución del Solver mediante el IMPES Clásico en [s]			Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	**Ejec.**	
100	**0.00019**	**480**	**0.25**
300	**0.00021**	**480**	
500	**0.00023**	**480**	

Tabla 4.9 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Anisotrópico.

Tiempo Promedio de Ejecución del Solver mediante el IMPES Mejorado en [s]			Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	**Ejec.**	
100	**0.00018**	**120**	**1**
300	**0.00020**	**120**	
500	**0.00021**	**120**	

Tabla 4.10 Tiempo de Ejecución del Algoritmo de Thomas con el IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Anisotrópico.

Tiempo Total de Ejecución del Algoritmo IMPES Clásico en [s]		Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	
100	**47.561**	**0.25**
300	**130.876**	
500	**209.658**	

Tabla 4.11 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Anisotrópico.

Tiempo Total de Ejecución del Algoritmo IMPES Mejorado en [s]		Δt [días]
Tamaño de malla	**Algoritmo de Thomas**	
100	**24.771**	**1**
300	**51.875**	
500	**56.073**	

Tabla 4.12 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Anisotrópico

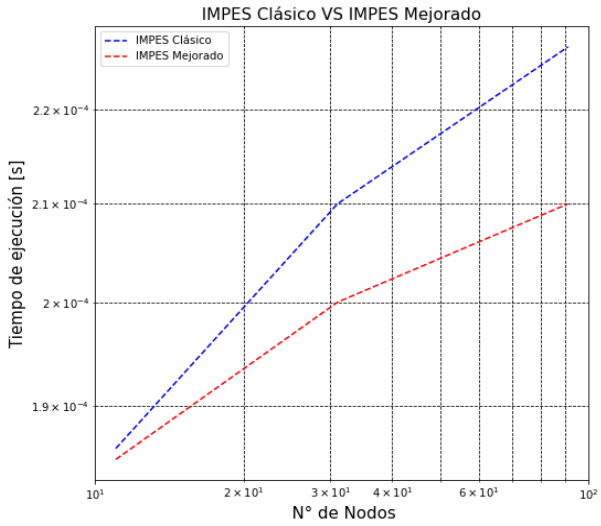


Figura 4.21 Gráfica de Rendimiento en el Tiempo de Ejecución del Solver entre los Algoritmos IMPES Clásico e IMPES Mejorado un Medio Anisotrópico.

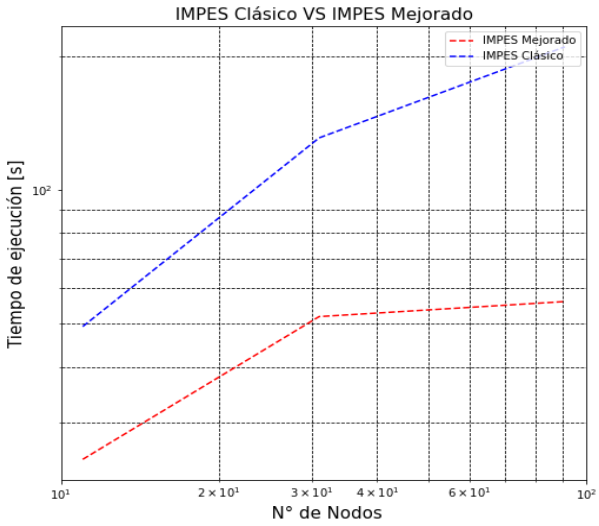


Figura 4.22 Gráfica del Rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Anisotrópico.

Cociente de Aceleración de los Algoritmos IMPES Clásico e IMPES Mejorado	
Tamaño de malla	**Algoritmo de Thomas**
100	**1.920**
300	**2.522**
500	**3.739**

Tabla 4.13 Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Anisotrópico.

.

4.3.1.2 Modelo Bidimensional: Medio Isotrópico

Ahora se analiza el comportamiento de los solvers descomposición LU optimizado (SPSOLVE), GMRES y BICGSTAB, de la biblioteca de *SciPy* de *Python*, con los cuales se resuelven los algoritmos para el modelo bidimensional, comenzando por el medio isotrópico y variando las mallas de 11 a 31 y a 91 nodos. Se empieza la comparación para ambos algoritmos con un paso de tiempo de 0.25 días, con el objetivo de comprobar que el comportamiento en cuanto a eficiencia de los solvers es similar para ambos algoritmos. La Tabla 4.14 y la Tabla 4.15 muestra los tiempos promedio de ejecución de los solvers SPSOLVE, GMRES y BICGSTAB con pasos de tiempo de 0.25 [días], en ambos algoritmos; IMPES Clásico y Mejorado respectivamente, de acuerdo con las dos tablas, los solvers se ejecutaron seis mil veces, sus valores son semejantes y su comportamiento es análogo. Si se observa con cuidado los valores en ambas tablas para la malla de 11 nodos, se capta que el orden de magnitud se encuentra en el rango de 10^{-3} y sus valores son idénticos; 1 milésima para el SPSOLVE, 2 milésimas para el GMRES y 3 milésimas para el BICGSTAB procediendo ascendentemente. Ahora, si se presta atención en la malla de 31 nodos, en ambas tablas los valores son muy parecidos; esta vez el valor para el SPSOLVE sigue siendo el más pequeño, el mayor valor es para el GMRES, y el BICGSTAB es el valor mediano. Con esta malla, a pesar de que el tiempo promedio de ejecución del SPSOLVE es una milésima más pequeña en el IMPES Clásico que en el Mejorado, los tiempos de ejecución para el GMRES y el BICGSTAB son ligeramente más grandes para el IMPES Clásico con respecto al Mejorado. La tendencia en el comportamiento de la malla de 31 nodos se repite cuando se utiliza una malla de 91 nodos, es decir, el tiempo promedio de ejecución con valor más corto es el del SPSOLVE, el del GMRES es el mayor y el del BICGSTAB es el mediano; sin embargo; eta vez todos los valores en el IMPES Clásico son ligeramente más grandes que en el IMPES Mejorado. La Figura 4.23 y la Figura 4.24 exhiben cómo es la conducta de los diferentes solvers, al tratarse de métodos directos e iterativos, parecen tomar una forma exponencial para ambos algoritmos, aunque ligeramente más pequeños para el IMPES Mejorado.

Si se comparan los tiempos totales de ejecución con el mismo paso de tiempo, se obtienen tiempos muy similares en la ejecución de ambos algoritmos y esto se debe a que la saturación para el caso del IMPES Clásico se resuelve una vez, mientras que en el Mejorado solo se resuelve un par de veces. En la Tabla 4.16 y en la Tabla 4.17 se puede examinar los diferentes tiempos de ejecución de acuerdo al mallado y los diferentes solvers que se utilizaron para la solución del sistema matricial, y al tener el mismo paso de tiempo para ambos casos, se percibe que para la malla pequeña en el IMPES Clásico todos los tiempos de ejecución son más cortos respecto del Mejorado. Para la malla de 31 x 31 se aprecia que los tiempos de ejecución son menores únicamente con el SPSOLVE, en tanto con GMRES y BICGSTAB son mayores en el IMPES Clásico respecto el IMPES Mejorado.

Esto se repite para la malla de 91 x 91, siendo el tiempo de ejecución menor con el SPSOLVE y mayor con el GMRES y BICGSTAB en el IMPES Clásico respecto el Mejorado.

Tiempo Promedio de Ejecución de los Solvers mediante el IMPES Clásico en [s]					**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	**Ejec.**	
11 x11	**0.001**	**0.002**	**0.003**	**6000**	**0.25**
31 x 31	**0.011**	**0.031**	**0.026**	**6000**	
91 x 91	**0.557**	**1.835**	**0.804**	**6000**	

Tabla 4.14 Tiempo de Ejecución Promedio de los Solvers mediante el IMPES Clásico para un Paso de Tiempo de 0.25 días en un Medio Isotrópico.

Tiempo Promedio de Ejecución de los Solvers mediante el IMPES Mejorado en [s]					**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	**Ejec.**	
11 x11	**0.001**	**0.002**	**0.003**	**6000**	**0.25**
31 x 31	**0.012**	**0.029**	**0.025**	**6000**	
91 x 91	**0.554**	**1.668**	**0.757**	**6000**	

Tabla 4.15 Tiempo de Ejecución Promedio de los Solvers mediante el IMPES Mejorado para un Paso de Tiempo de 0.25 días en un Medio Isotrópico.

Tiempo Total de Ejecución del Algoritmo IMPES Clásico en [s]				**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	
11 x11	**59.12**	**67.54**	**72.50**	**0.25**
31 x 31	**206.35**	**332.56**	**299.04**	
91 x 91	**4320.69**	**12135.42**	**5916.74**	

Tabla 4.16 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Isotrópico.

Tiempo Total de Ejecución del Algoritmo IMPES Mejorado en [s]				**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	
11 x11	**63.75**	**70.18**	**77.86**	**0.25**
31 x 31	**220.12**	**324.91**	**303.12**	
91 x 91	**4455.79**	**11230.53**	**5736.97**	

Tabla 4.17 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 0.25 días en un Medio Isotrópico.

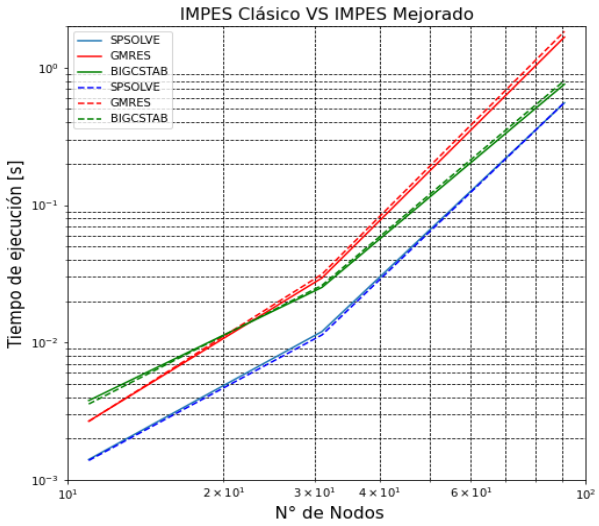


Figura 4.23 Gráfica del Rendimiento en el Tiempo de Ejecución de los Solvers entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico.

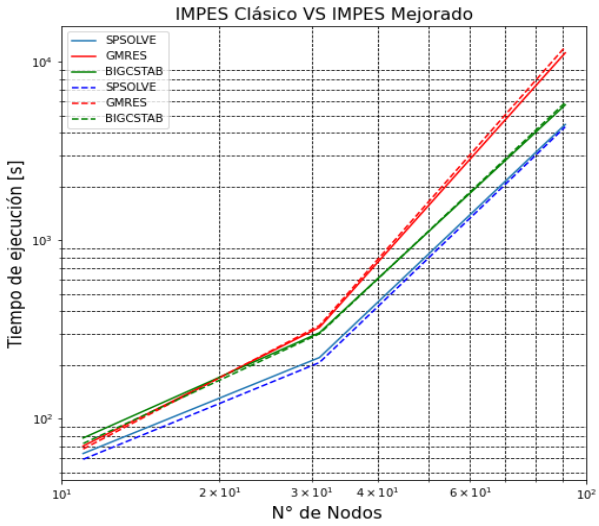


Figura 4.24 Gráfica del Rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Isotrópico.

La ventaja del IMPES Mejorado sale a relucir cuando se utilizan pasos de tiempo más grandes para la presión. Entonces, se observa un mejor desempeño para el IMPES Mejorado sin importar el solver que se ha empleado. En la Tabla 4.18 se registran los tiempos de ejecución para el IMPES Mejorado con un paso de tiempo de un día, y ahora se puede advertir que el desempeño para este algoritmo es mejor para las mallas más grandes, en este caso de igual manera se puede realizar un análisis en la aceleración de la ejecución de ambos algoritmos, ya que si se observan detenidamente los valores en la Tabla 4.19, se puede percatar que los solvers más eficientes son el SPSOLVE y el BICGSTAB. La Figura 4.25 muestra gráficamente esta diferencia en el tiempo de ejecución.

Tiempo Total de Ejecución del Algoritmo IMPES Mejorado en [s]				**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	
11 x11	**56.84**	**59.16**	**61.42**	**1**
31 x 31	**118.36**	**156.41**	**144.65**	
91 x 91	**3482.91**	**4650.2**	**3504.18**	

Tabla 4.18 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1 día en un Medio Isotrópico.

Cociente de Aceleración de los Algoritmos IMPES Clásico e IMPES Mejorado			
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**
11	**1.04**	**1.14**	**1.18**
31	**1.743**	**2.12**	**2.06**
91	**1.24**	**2.60**	**1.68**

Tabla 4.19 Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Isotrópico

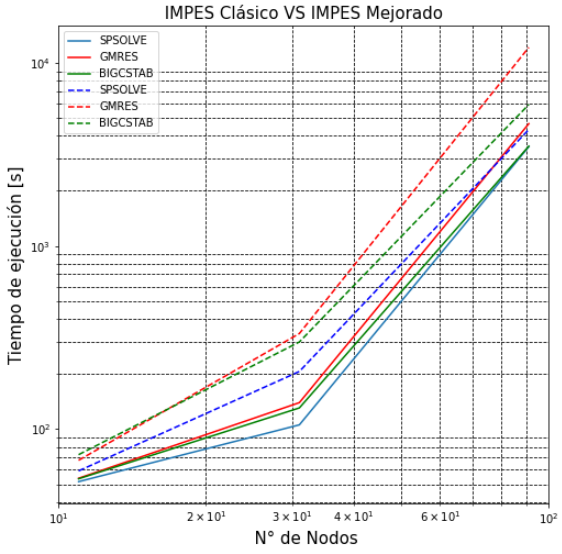


Figura 4.25 Gráfica del Rendimiento en el Tiempo de Ejecución entre los algoritmos IMPES Clásico e IMPES Mejorado un Medio Isotrópico con un Paso de Tiempo de 1 día.

4.3.1.3 Modelo Bidimensional: Medio Anisotrópico

Para finalizar este estudio se llega al escenario donde el medio y la permeabilidad se varía de forma aleatoria en un rango de 10 a 100 mD. En la Tabla 4.20 y en la Tabla 4.21 se observa el rendimiento de los algoritmos para las variaciones del paso de tiempo de 0.25 y 1 [días] para los IMPES Clásico y Mejorado, respectivamente, donde como se esperaba, los tiempos tienen un comportamiento similar cuantitativa y cualitativamente que en el caso isotrópico.

Tiempo Promedio de Ejecución de los Solvers mediante el IMPES Clásico en [s]					**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	**Ejec.**	
11 x11	**0.001**	**0.003**	**0.004**	**6000**	**0.25**
31 x 31	**0.011**	**0.041**	**0.033**	**6000**	
91 x 91	**0.561**	**2.175**	**0.832**	**6000**	

Tabla 4.20 Tiempo de Ejecución Promedio de los Solvers mediante el IMPES Clásico para un Paso de Tiempo de 0.25 días en un Medio Anisotrópico.

Tiempo Promedio de Ejecución de los Solvers mediante el IMPES Mejorado en [s]					**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	**Ejec.**	
11 x11	**0.001**	**0.003**	**0.005**	**1500**	**1**
31 x 31	**0.012**	**0.038**	**0.031**	**1500**	
91 x 91	**0.555**	**2.149**	**0.817**	**1500**	

Tabla 4.21 Tiempo de Ejecución Promedio de los Solver mediante el IMPES Mejorado para un Paso de Tiempo de 1 día en un Medio Anisotrópico.

Finalmente, para concluir este estudio se contempla el rendimiento de los algoritmos. Si se observa la Tabla 4.22 y la Tabla 4.23 en donde se vigila el tiempo de ejecución de estos, se verifica que son mucho más cortos para el IMPES Mejorado. Gráficamente, se nota que el desfase de las curvas es más notorio, y si se pone atención en la Tabla 4.24 de aceleración del solver se comprueba que es mucho más eficiente cuando las mallas son más finas, ya que se puede ver de manera contundente que para las mallas de 91x91 el IMPES Mejorado es hasta 3 veces más rápido que el Clásico.

Tiempo Total de Ejecución de los Solvers mediante el IMPES Clásico en [s]				**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	
11 x11	**60.95**	**77.48**	**82.71**	0.25
31 x 31	**218.47**	**399.51**	**344.48**	
91 x 91	**4333.09**	**14114.34**	**5999.60**	

Tabla 4.22 Tiempo Total de Ejecución del Algoritmo IMPES Clásico con un Paso de Tiempo de 0.25 días en un Medio Anisotrópico.

Tiempo Total de Ejecución de los Solvers mediante el IMPES Mejorado en [s]				**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	
11 x11	**51.71**	**53.45**	**58.03**	1
31 x 31	**108.75**	**145.18**	**135.59**	
91 x 91	**1376.24**	**3795.68**	**1923.86**	

Tabla 4.23 Tiempo Total de Ejecución del Algoritmo IMPES Mejorado con un Paso de Tiempo de 1día en un Medio Anisotrópico.

Cociente de Aceleración de los Algoritmos IMPES Clásico e IMPES Mejorado			
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**
11x11	**1.178**	**1.45**	**1.42**
31x31	**2.0**	**2.75**	**2.54**
91x91	**3.148**	**3.718**	**3.118**

Tabla 4.24 Cociente de Aceleración del Rendimiento de los Algoritmos IMPES Clásico vs IMPES Mejorado en un Medio Anisotrópico.

En la Figura 4.26 se percibe que el comportamiento de todos los solvers es idéntico, ya que la matriz de coeficientes del sistema continúa siendo diagonalmente dominante, no importa que el medio sea anisotrópico, ni la diferencia en el paso de tiempo. En la Figura 4.27 si se puede apreciar un desfase más marcado, el cual se podría explicar por la diferencia en el paso de tiempo.

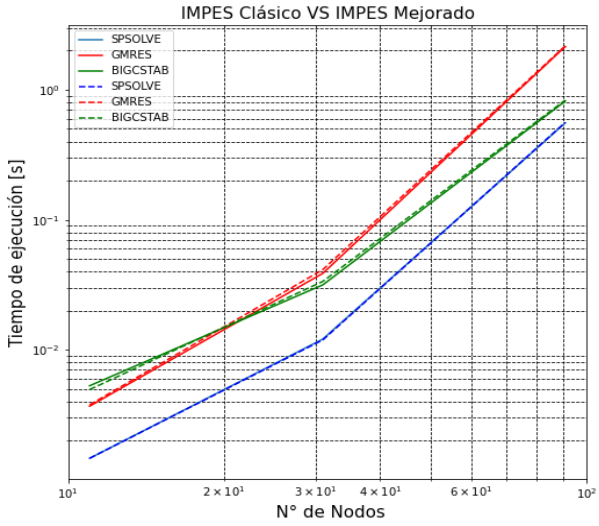


Figura 4.26 Gráfica del Rendimiento en el Tiempo de ejecución de los Solvers entre los Algoritmos IMPES Clásico e IMPES Mejorado en un Medio Anisotrópico con un Paso de Tiempo de 0.25 y [1 días].

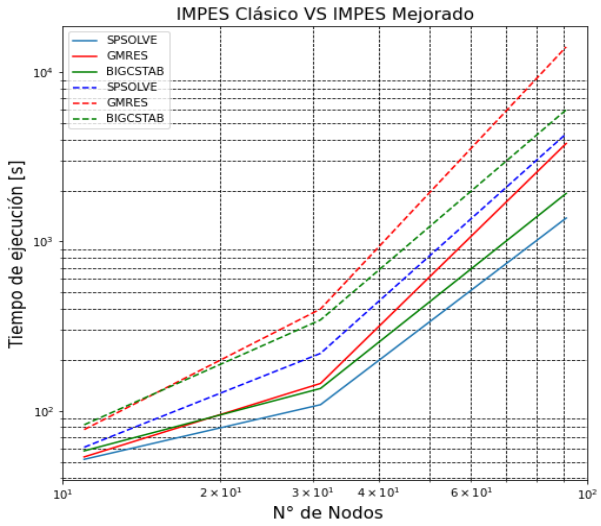


Figura 4.27 Gráfica del rendimiento en el Tiempo de Ejecución entre los Algoritmos IMPES Clásico e IMPES Mejorado un Medio Anisotrópico con un Paso de Tiempo de 0.25 y 1 [días].

4.3.1.4 Comparativa utilizando una Matriz Dispersa

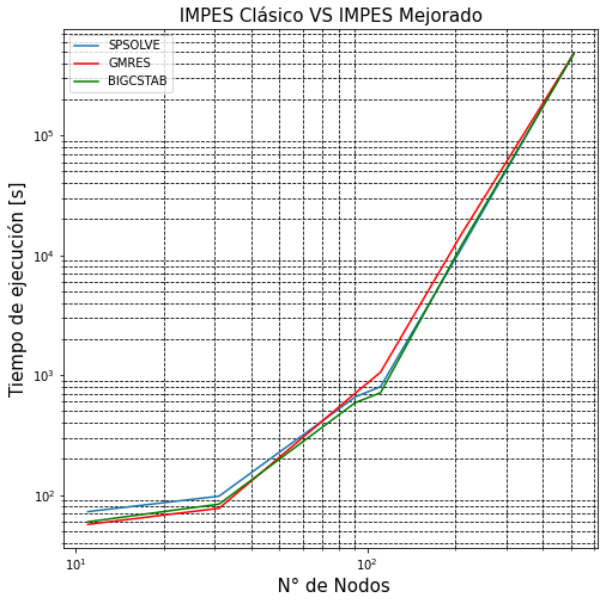
En esta sección se hace la última comparación en el rendimiento y eficiencia del IMPES Mejorado bajo la implementación CSR, que ayuda a resolver uno de los principales obstáculos en la simulación numérica debido principalmente a la memoria de los ordenadores. La observación fundamental arraiga en el almacenamiento de la matriz con la que el ordenador trabaja, el almacenamiento de una matriz densa ocupa mucha más memoria que el almacenamiento de una matriz vacía, esto generalmente ocasiona el desplome de la memoria de los equipos de cómputo y por esta razón se generan errores que vuelven imposible llevar a cabo la ejecución de los simuladores con mallas extremadamente finas. En seguida se muestra los resultados obtenidos tras ejecutar el algoritmo IMPES Mejorado utilizando una matriz dispersa vacía en donde se estudia su comportamiento para mallas de hasta 511 nodos y un medio isotrópico con el objetivo de manifestar que bajo el panorama de las matrices ralas es posible llevar a la práctica estas tareas, brincando dicho problema de almacenamiento. En la Tabla 4.25 se muestran los tiempos de ejecución son muy semejantes y que esta vez suele haber ligeras variaciones dependiendo del tamaño de la malla y del solver en cuestión, por ejemplo para la malla de once nodos el tiempo de ejecución es menor para el solver GMRES, le sigue el solver BICGSTAB y el tercer lugar es para el SPSOLVE, para la malla de 31 nodos la conducta sigue la misma inclinación; el primer lugar lo ocupa el GMRES, seguido del BICGSTAB y por último el SPSOLVE, y a pesar de ello para la malla de 91 nodos cambia, ahora es SPSOLVE seguido de BICGSTAB y en última instancia GMRES, a partir de aquí la orientación en los valores de los datos es la misma desde la malla de 111 nodos hasta la de 511 nodos, por lo que se puede

Tiempo Total de Ejecución del Algoritmo IMPES Mejorado en [s]				**Δt [días]**
Tamaño de malla	**SPSOLVE**	**GMRES**	**BICGSTAB**	
11 x 11	**72.55**	**56.8232**	**59.6520**	
31 x 31	**97.6523**	**77.33580**	**83.8**	
91 x 91	**658.2**	**708.1**	**589.7148**	**1**
111x111	**800.4574**	**1056.1**	**712.5101**	
211 x 211	**11552.9331**	**15222.3210**	**12257.3**	
511 x 511	**490128.3**	**482244.7124**	**479279.65**	

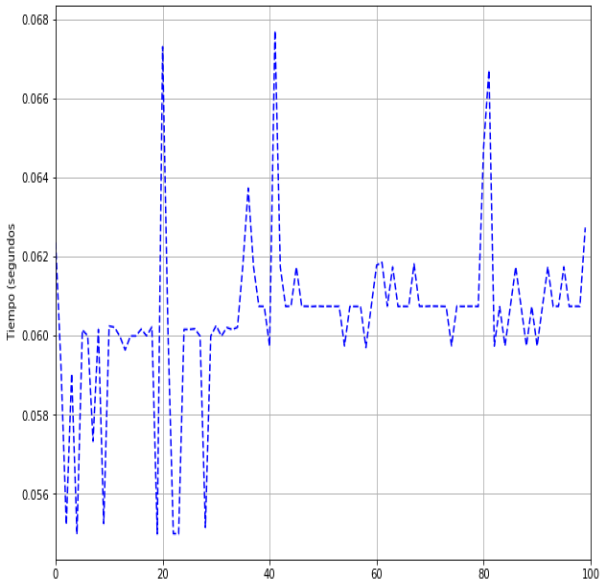
Tabla 4.25 Tiempo Total de Ejecución del IMPES Mejorado con Matriz Dispersa.

deducir que la potencialidad de los solvers también depende de su estabilidad, que de igual modo tendrán influencia directa en el desempeño del algoritmo. Es importante mencionar que la eficiencia de esta implementación se encuentra en el hecho de que, para una malla de 500 nodos o mayor, utilizar una matriz densa ya no fue funcional, mientras que, para esta implementación, si fue posible lograr la ejecución, también reside en el hecho de que el IMPES Clásico requería un paso de tiempo muy corto, lo que repercutía en un tiempo mucho mayor de ejecución, y, por lo tanto, ya no fue factible terminar su ejecución. En la Figura 4.28 a) se puede observar que el tiempo de ejecución del algoritmo, sea cual sea el solver, tiene una tendencia exponencial conforme se incrementa el número de nodos en la malla, esto se puede explicar debido a las oscilaciones que sufren los solvers al momento de su ejecución en cualquier momento del cálculo.

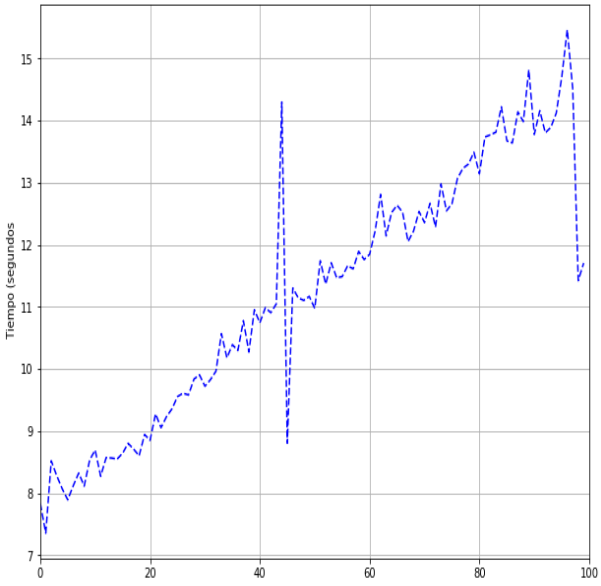
La Figura 4.28 b) muestra una oscilación en el tiempo de ejecución aproximadamente constante para el solver SPSOLVE, la Figura 4.28 c) exhibe una oscilación creciente para el GMRES y la Figura 4.28 d) demuestra una oscilación ligeramente decreciente para el BICGSTAB.



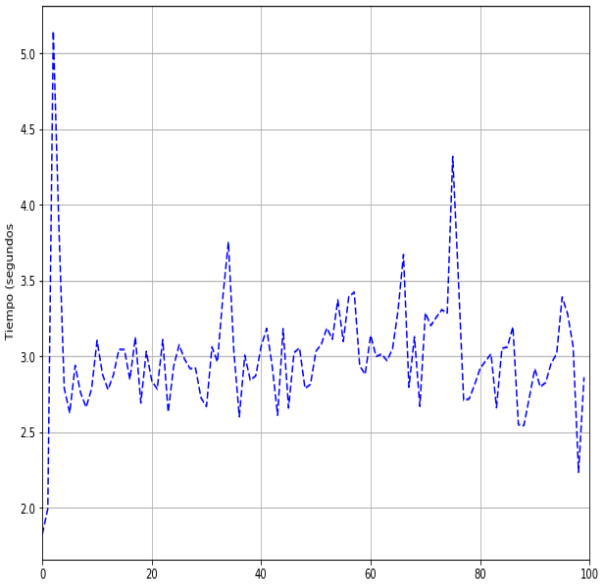
a) Gráfica del Tiempo de Ejecución del IMPES Mejorado.



b)Tiempo de Ejecución del solver SPSOLVE.



c) Tiempo de Ejecución del solver GMRES.



d) Tiempo de Ejecución del solver BICGSTAB.

Figura 4.28 Comportamiento del Tiempo de Ejecución.

CONCLUSIONES

Las metodologías numéricas y computacionales, el modelo matemático y numérico para simular el flujo de fluidos bifásico incompresible en medios porosos, son muy utilizados y ampliamente aceptados en la industria petrolera. Por lo que, en la presente tesis, se orienta nuestro empeño en realizar las comparativas de rendimiento y verificar dos de las implementaciones existentes en la literatura, para poder examinar su comportamiento y encontrar los factores que controlan su eficiencia, para poder buscar alternativas que ayuden a optimizar e impulsar el desarrollo de estos algoritmos. Para poder llevar a cabo mejores implementaciones futuras en este estudio se concluye que:

- Para mallas gruesas, por ejemplo, 11×11, los tiempos de ejecución entre el IMPES Clásico y mejorada son similares, por lo que se pudiera usar uno o el otro.
- Se puede concluir que en particular el solucionador SPSOLVE (método directo de descomposición LU optimizado) tarda menos tiempo en ejecutarse que los otros dos solucionadores, y que el GMRES y el BICGSTAB (métodos iterativos) tiene tiempos de ejecución muy similares. Esto pudiera deberse a que los sistemas resueltos son relativamente pequeños, donde un método directo es más eficiente que uno iterativo.
- La sobrecarga en la memoria principalmente se debe a un tamaño de malla muy grande debido a que está ligada con el número de operaciones que se invierten en el cálculo de las operaciones para la ecuación de presión.
- Se puede observar un mejor comportamiento y eficiencia de cómputo del IMPES Mejorado en respuesta a la refinación de la malla.
- El IMPES Mejorado es más confiable para tratar problemas con medios anisotrópicos.

RECOMENDACIONES

A cuenta de lo mencionado anteriormente, y para poder continuar estos estudios, se recomienda:

- Extender el código a 3D con la capacidad de manejar múltiples pozos.
- Realizar pruebas utilizando diferentes precondicionadores en los solvers.
- Implementar un paso de tiempo adaptativo general para la presión en el IMPES Mejorado.
- Desarrollar una interfaz gráfica para la utilización de los códigos.
- Implementar técnicas de cómputo de alto rendimiento para mejorar la eficiencia de los códigos.

5. Referencias

[1] T. A.-K. J. H. &. K. G. R. Ertekin, *Basic Applied Reservoir Simulation*, Texas: Society of petroleum engineer (SPE), 2001.

[2] G. H. Y. M. Zhangxin Chen, *Computational Methods for Multiphase Flow in Porous Media*, Texas: SIAM,Southern Methodist University Dallas, 2006.

[3] Z. Chen, *Reservoir Simulation Mathematical Techniques in Oil Recovery*, Alberta, Canada: SIAM, University of Calagary, 2007.

[4] K. Bjorlykke, *Petroleum Geoscience: From Sedimentary Enviroments to Rock Physics*, Springer, 2010.

[5] M. J. N. K. Economides, *Reservoir Stimulation*, 3er ed., Wiley,, 2010.

[6] K. Velten, *Mathematical Modeling and Simulation. Introduction for Scientists*, Wiley-VCH, 2009.

[7] A. S. Odeh, *An Overview of Mathematical Modeling of the Behavior of Hydrocarbon Reservoir*, SIAM, 1982.

[8] R. C. M. Carrada, *Simulación Numérica de Yacimientos Petrolero Empleando el Método de Líneas de Corriente*, México: UNAM, IGF, 2014.

[9] J. D. W. P. a. H. H. R. J. J. Douglas, *A method for calculating multi-dimensional immiscible displacement*, SPE AIME 216, 297–306., 1959.

[10] S. C. R. a. L. N. T. N. M. *Nachtigal, How Fast are Nonsymmetric Matrix Iterations?*, J. Matrix Anal. Appl 13 ed., SIAM, 1992, p. 778–795.

[11] H. K. V. a. W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, New York: Longman Scientific & Technical, 1995.

[12] R. I. V. Balderrama, *Métodos Numéricos*, vol. 2a ed, México: Trillas, 2007.

[13] S. C. C. &. R. P. Canale, *Métodos Numéricos para Ingenieros*, México, D. F.: McGraw-Hill, 2007.

[14] G. H. G. &. C. F. v. Loan, *Matrix Computations*, Baltimore, London.: Johns Hopkins University, 1996.

[15] C. Johnson, *Numerical Solutions of Partial Differential Equations by the Finite Element Method*, Cambridge, England: Cambridge University Press, 1994.

[16] Z. Chen, *Finite Element Methods and Their Applications*, Heidelberg and New York: Springer-Verlag, 2005.

[17] D. Peaceman, *Fundamentals of Numerical Reservoir Simulation*, New York: Elsevier, 1977.

[18] K. a. Settari, *Petroleum Reservoir Simulation*, London: Applied Science Publishers Ltd., 1979.

[19] M. R. H. a. E. L. Stiefel, *Methods of Conjugate Gradients for Solving Linear Systems*, J. Res. National Bureau of Standards, 1952.

[20] O. Axelsson, *Iterative Solution Methods, Cambridge*,: Cambridge University Press, 1994.

[21] C. Lanczos, *Solution of Systems of Linear Equations by Minimized Iterations*, J. Res. National Bureau of Standards, 1952.

[22] P. Sonneveld, CGS, *A Fast Lanczos-Type Solver for Nonsymmetric Linear Systems*, vol. J. Sci. Statist. Comput. 10, SIAM, 1989, p. 36–52.

[23] H. A. v. d. Vorst, *BI-CGSTAB:A Fast and Smoothly Converging Variant of BI-CG*, vol. Sci Statist. Comput. 13, SIAM J., 1992, p. 631–644.

[24] Yu.Kuznetsov, *Some Problems in the Theory and Applications of Iterative Methods*, Novosibirsk, Russia: Ph.D. Thesis, 1996.

[25] Y. S. a. M. H. Schultz, *GMRES:A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems*, vol. J. Sci. Statist. Comput. 7, SIAM J. Sci. Statist. Coput., 1986, p. 856–869.

[26] W. E. Arnoldi, *The Principle of Minimized Iteration in the Solution of the Matrix Eigenvalue Problem*, Quart, 1951.

[27] Y. Saad, *Iterative Methods for Sparse Linear Systems*, vol. 2nd ed., Philadelphia: SIAM, 2004.

[28] I. H. a. G. F. Pinder, *Mathematical Modeling in Science and Engineering: An Axiomatic Approach*, New Jersey, U. S.: John Wiley & Sons, 2012.

[29] J. a. E. Abou-Kassem, *An Efficient Algorithm for Removal of Inactive Blocks in Reservoir Simulation*, Can.Pet. Tech., 1992.

[30] A. E. C. a. R. T. McDonald, *Efficient Use of Mass Storage During Elimination for Sparse Sets of Simulation Equations.*, SPEJ17(4):300-16, 1977.

[31] P. T. S. J. R. a. S. G. Woo, *Application of Sparse Matrix Techniques in Reservoir Simulation*, Las Vegas: SPE,#4544 paper presented at the 48th Annual Fall Meeting, 30 Sept-3Oct, 1973.

[32] H. S. a. k. H. C. Price, *Direct Methods in Reservoir Simulation*, Vol 2J 14 (3) :295-308, SPE, 1974.

[33] V. L. T. Juárez, *Modelado Numérico de Procesos Térmicos de Recuperación Mejorada de Hidrocarburos*, Ciudad Universitaria,México, D.F.: UNAM, 2018.

[34] A. y. K. M. Datta-Gupta, *Streamline Simulation: Theory and Practice*, SPE, 2006.

[35] A. S. e. a. Al-Zawawi, *Using Streamline and Reservoir Simulation to Improve Waterflood Management*, Saudi Aramco Journal of Technology, 2011.

[36] T. K. A. I. Hantschel, *Fundamentals of Basin and Petroleum Systems Modeling*, Springer, 2009.

[37] A. R. T. Iske, *Mathematical Methods and Modelling in Hydrocarbon Exploration and Production*., Springer, 2000.

[38] D. W. Peaceman, *Interpretation of Well-Block Pressures in Numerical Reservoir Simulation, Denver*: SPE 6893, The 52nd Anual Fall Technical Conference and Exhibition, 1977.

[39] J. Arps, *Analisys of Decline Curves*, Houston: AIME, 1945.

[40] A. E. Arps J. J. & Smith, *Practical Use of Bottom-hole Pressure Build-up Curves*, Drill&Prod. Prac., 1949.

[41] S. E. a. L. M. C. Buckley, *Mechanism of Fluid Displacement in Sands*, Trans, AIME, 1942.

[42] «La factorización LU,» [En línea]. Available: https://www.uv.es/. [Último acceso: 18 mayo 2022].

[43] T. K. A. I. Hantschel, *Fundamentals of Basin and Petroleum Systems Modeling*, Springer, 2009.

ANEXO A. Nomenclatura

	Símbolo	*Descripción*	*Unidad*
1	$\boldsymbol{A}$	*Matriz de coeficientes de un sistema (matriz de rigidez)*	
2	AP	*Coeficientes de la presión en el puto* i, j	
3	AE	*Coeficientes de la presión en el puto* $i+1, j$	
4	AW	*Coeficientes de la presión en el puto* $i-1, j$	
5	AN	*Coeficientes de la presión en el puto* $i, j+1$	
6	AS	*Coeficientes de la presión en el puto* $i, j-1$	
7	$a_{i,j}$	*Elementos de la matriz* $\boldsymbol{A}$	
8	b	*Exponente en la ecuación de declinación*	
9	$\bar{B}$	*Vector del lado derecho de un sistema matricial*	
10	D	*Tasa de Declinación*	$[Dias^{-1}]$
11	DS_{max}	*Diferencial de saturación máximo Permitido*	
12	h	*Distancia entre dos puntos de malla*	$[L], [m\}, [ft]$
13	$f(x_i)$	*Función de x*	
14	f_α	*Flujo fraccional de la fase* $\boldsymbol{\alpha}$	
15	k	*Constante de la Tasa de Declinación*	
16	$\bar{\bar{k}}$	*Permeabilidad absoluta*	
17	J	*Intervalo del tiempo total de interés*	
18	$\kappa_{r\alpha}$	*Permeabilidad relativa de la fase* $\boldsymbol{\alpha}$	
19	$\mathcal{K}^k$	*k-ésimo espacio de Krylov*	
20	$P(x)$	*Polinomio de Taylor*	
21	P_i	*Relación de recurrencia en el algoritmo de Thomas*	
22	p_c	*Presión capilar*	$[Psi]$, $[Pa]$
23	p_{cow}	*Presión capilar de las fases aceite-agua*	$[Psi]$, $[Pa]$
24	p_{bh}	*Presión de fondo (inyección/produccion cte.)*	
25	p_o	*Presión del aceite*	$[Psi]$, $[Pa]$
26	p_w	*Presión del agua*	$[Psi]$, $[Pa]$
27	p_o^0	*Presión inicial*	$[Psi]$, $[Pa]$
28	q	*Gasto de producción*	$[L^3/d], [m^3/d], [RB/d]$
29	q_α	*Gasto de la fase* $\boldsymbol{\alpha}$	$[L^3/d], [m^3/d], [RB/d]$
30	$\tilde{q}_o$	*Gasto promedio del aceite*	$[L^3/d], [m^3/d], [RB/d]$
31	$\tilde{q}_w$	*Gasto promedio del agua*	$[L^3/d], [m^3/d], [RB/d]$
32	Q_i	*Relación de recurrencia en el algoritmo de Thomas*	
33	$\boldsymbol{r}^k$	*k-ésimo vector residual*	
34	r_e	*Radio equivalente*	$[ft], [in]$
35	r_{well}	*Radio de pozo*	$[ft], [in]$
36	$\mathbb{R}^d$	*Espacio euclidiano donde* $d = 1, 2, 3$	
37	$\mathbb{R}$	*Conjunto de números Reales*	

38	S_α	*Saturación de la fase*	
39	S_w	*Saturación del agua*	
40	S_o	*Saturación del aceite*	
41	S_w^0	*Saturación inicial*	
42	S_w^{n+1}	*Saturación en el tiempo actual*	
43	t	*tiempo*	$[s]$
44	T_{max}	*Tiempo de simulación*	$[días]$
45	$\mathbf{V}^k$	*Proyección ortogonal*	
46	$\vec{\boldsymbol{v}}$	*Velocidad de Darcy*	
47	$\vec{\boldsymbol{v}}_{\boldsymbol{\alpha}}$	*Velocidad de la fase* $\boldsymbol{\alpha}$	
48	x_i	*i-ésima iteración en el algoritmo de Thomas*	
49	$\overline{\boldsymbol{x}}$	*Vector incógnito en un sistema matricial*	
50	$\boldsymbol{x^0}$	*Estimación inicial en un sistema matricial*	
51	$\boldsymbol{x^k}$	*k-ésima iteración*	
52	$cond(\boldsymbol{A})$	*Número de condición de la matriz A*	
53	ΔV	*Diferencial de volumen*	
54	Δt	*Paso de tiempo*	
55	Δx	*Distancia entre puntos de una celda en la dirección x*	
56	Δy	*Distancia entre puntos de una celda en la dirección y*	
57	Δz	*Distancia entre puntos de una celda en la dirección z*	
58	$\Delta t_S^{n+1,m}$	*Paso de tiempo para la saturación en el IMPES Mejorado*	$[s]$
59	ρ_α	*Densidad de la fase* $\boldsymbol{\alpha}$	
60	λ	*Movilidad total*	
61	λ_α	*Movilidad de la fase* $\boldsymbol{\alpha}$	
62	μ_α	*Viscosidad de la fase* $\boldsymbol{\alpha}$	$[cp]$
63	$\wp$	*Constante gravitacional*	
64	$\|\cdot\|$	*Norma sobre un espacio*	
65	$\|\cdot\|_A$	*Norma sobre un espacio no conforme*	
66	$\langle \cdot, \cdot \rangle$	*Producto Interno*	
67	$\frac{\partial p_c}{\partial S_w}$	*Parcial de la presión capilar respecto la saturación*	
68	$\frac{\partial}{\partial x}$	*Derivada parcial respecto a x*	
69	$\frac{\partial}{\partial t}$	*Derivada parcial respecto a t*	
70	ϕ	*Porosidad*	
71	$\boldsymbol{\nabla}$	*Operador divergencia (nabla)*	
72	$\boldsymbol{\nabla p_C}$	*Gradiente de presión capilar*	
73	$\boldsymbol{\nabla S_w}$	*Gradiente de saturación*	
74	$\boldsymbol{\nabla z}$	*Gradiente de altura*	

ANEXO B. Modelado de Pozos de Peaceman

Para incorporar los pozos de inyección y de producción de una manera apropiada se considera el modelo de Peaceman extendido a mallas no uniformes, tomando en cuenta las fuerzas de gravedad, anisotropía del medio, efectos superficiales y dirección de pozo (vertical u horizontal). Por lo tanto, para un pozo vertical, ya sea de inyección o producción correspondiente a:

$$q_\alpha = \frac{2\pi\rho_\alpha k11\Delta z\sqrt{k_{11}k}_{22}}{\mu_\alpha(ln(r_e/r_{well}) + s}(p_{bh} + p_\alpha - \rho_\alpha\wp(z_{bh} - z)) \tag{5.1}$$

donde $\wp$ es la magnitud de la aceleración gravitacional, z es la profundidad, z_{bh} es la profundidad de referencia del pozo, r_{well} es el radio del pozo, ρ_α, $k_{r\alpha}$ y μ_α se refieren a la densidad, permeabilidad relativa y viscosidad de la fase α. El radio equivalente r_e de la ecuación (5.1) para un medio anisotrópico se define como:

$$r_e = \frac{0.14((k_{22}/k_{11})^{1/2}\Delta x^2 + ((k_{11}/k_{22})^{1/2}\Delta y^2)^{1/2}}{0.5((k_{22}/k_{11})^{1/4} + (k_{11}/k_{22})^{1/4})} \tag{5.2}$$

donde Δx y Δy se refieren al tamaño de celda en los ejes x y y que contiene el pozo vertical y k_{11} y k_{22} se refieren a las permeabilidades absolutas en las direcciones x y y. Otra magnitud muy importante en la ingeniería petrolera es el índice del pozo WI, el cual se expresa como sigue:

$$WI = \frac{2\pi\Delta z\sqrt{K_{11}K_{22}}}{ln(r_e/r_{well}) + s} \tag{5.3}$$

los efectos en la cercanía del pozo que pudieran afectar la productividad o los procesos asociados a la inyección, pueden ser incorporados usando el factor de daño s. Estos efectos se desprecian para efectos de este trabajo ($s = 0$). Para el caso de pozos perforados horizontalmente se tienen que llevar algunas modificaciones a la ecuación (5.1) para considerar la orientación correcta en los ejes coordenados. El índice de un pozo horizontal paralelo a la dirección del eje x es:

$$WI = \frac{2\pi\Delta x\sqrt{K_{22}K_{33}}}{ln(r_e/r_{well}) + s} \tag{5.4}$$

si el pozo es paralelo a la dirección y, se tiene:

$$WI = \frac{2\pi\Delta y\sqrt{K_{11}K_{33}}}{ln(r_e/r_{well}) + s} \tag{5.5}$$

para el caso abordado en esta tesis la ecuación (5.1), el pozo inyector queda:

$$q_w = \frac{2\pi\rho_w\lambda_w\Delta z\sqrt{k_{11}k_{22}}}{ln(r_e/r_{well})}\ (p_{bh} - p_o - p_c) \tag{5.6}$$

y el pozo productor queda:

$$q_o = \frac{2\pi\rho_o\lambda_o\Delta z\sqrt{k_{11}}k_{22}}{ln(r_e/r_{well})}\ (p_{bh} - p_o) \tag{5.7}$$

El acople de los pozos a las ecuaciones de balance o modelos matemáticos simplificados se Debe hacer siguiendo las restricciones que dicte el problema en particular. Por ejemplo, si se considera el caso en el que un fluido se inyecta a presión constante, es decir $p_{bh} = cte$, la variable a calcular será p_α, existen más restricciones como lo son flujo constante, máximo flujo permitido, etc., pero el alcance de esta tesis no abarca estas dos últimas consideraciones. La simulación numérica de los flujos de fluidos en los yacimientos de hidrocarburos debe tener en cuenta la presencia de pozos. La presión en una celda de malla que contiene un pozo es diferente de la presión promedio en ese bloque, y diferente de la presión de fondo que fluye hacia el pozo [38]. La dificultad de modelar pozos en una simulación numérica a escala de campo es que la región donde los gradientes de presión son mayores es la más cercana a un pozo y es mucho más pequeña que el tamaño espacial de las celdas en la malla (Figura 5.1). El uso del refinamiento de la malla alrededor del pozo puede aliviar este problema, pero puede conducir a una restricción poco práctica en los tamaños de paso de tiempo en la simulación numérica.

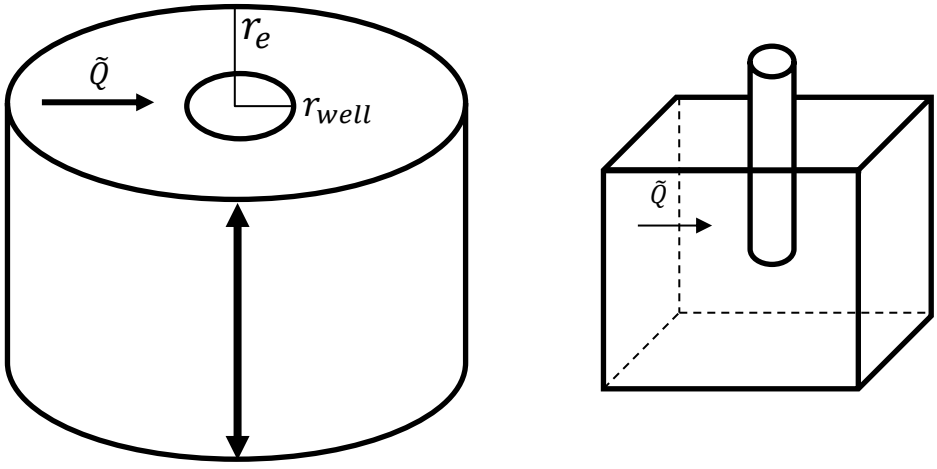


Figura 5.1 Representación de un pozo en una celda y a escala de yacimiento.

ANEXO C. Discretización del Problema.

Para poder acatar el problema es necesario comenzar con la discretización del medio, para este simulador desarrollado se utilizó un *ordenamiento de bloques natural* Figura 3.3 y Figura 3.4, por lo tanto, el orden de los nodos seria como muestra la Figura 5.2:

(n,0)	⋮	⋯	$n-1$	n
⋮	⋮	⋯	⋯	⋮
(2,0)	8	9	10	⋮
(1,0)	4	5	6	7
(0,0)	0	1	2	3
	(0,0)	(1,0)	(2,0) ⋯	(0,n)

Figura 5.2 Ordenamiento de bloques para una malla de n nodos.

Empezando el conteo de izquierda a derecha y de abajo hacia arriba para una malla de $M \times M$ para generar un total de n nodos y obtener una matriz pentadiagonal de dimensión $n \times n$, considerando además que el ordenamiento de los bloques en Python es de izquierda a derecha de arriba hacia abajo como se muestra en la Figura 5.3. Es importante tener en cuenta también la diferencia de los nodos internos y los nodos que pertenecen a las fronteras en la Figura 5.2 los nodos internos se representan sombreados de color gris mientras que las fronteras permanecen en blanco. Los nodos internos serán los que tengan mayor repercusión en los cálculos debido a que las condiciones de frontera serán de tipo Nauman, lo que significa que se tendrán condiciones de no flujo y cuyas condiciones iniciales S_w^0 y p_o^0 serán conocidas. La ventaja de tener este tipo de fronteras es que se pueden implementar más fácilmente, pues bastara con hacer cero los valores en esos puntos, pero se debe tener especial cuidado con los nodos internos, ya que solo en nodos específicos estos tendrán valor nulo, generalmente los nodos en las fronteras serán en los nodos que contengan las coordenadas $(0,j)$,$(i,0)$, (i,n),(n,j) para $0 < i < n$ y $0 < j < n$ y para los puntos en las esquinas de la malla $(0,0)$, $(0,n)$, $(n,0)$, (n,n). Al comenzar a aplicar el esquema de cinco puntos a partir del nodo cero de izquierda a derecha se forma la matriz de la Figura 5.3, en la que se puede observar en las diagonales AE y AW en algunos puntos tiene valor igual a cero que es donde corresponde a las fronteras este y oeste respectivamente, que es en el límite n de cada fila, entonces el conteo de coordenadas salta en el mismo orden a la siguiente columna para dar lugar a los índices de las coordenadas (i,j) como se muestra en la Figura 5.2. De la misma manera se puede observar

en la Figura 5.3 que las diagonales inferior y superior AS y AN aparecen en las coordenadas $(i, i+n)$ y $(i, j+n)$ que es donde se limita a la malla en las frontera norte y sur respectivamente, que es donde no existen los puntos cuando se aplica el esténcil de cinco puntos. Es importante mencionar que para su implementación se utiliza un nodo ficticio, que mantendrá las mismas propiedades que en los nodos internos, sin embargo, no se calcula ninguna presión en ese punto.

	0	1	2	3	⋯	n	6	7	8	⋯	n	11	12	13	⋯	⋯	⋯	n	p_i		B_i
0	AP	AE	0	0	0	AN	0	0	0	0	0	0	0	0	0	0	0	0	$p_{0,0}$		$B_{0,0}$
1	AW	AP	AE	0	0	0	AN	0	0	0	0	0	0	0	0	0	0	0	$p_{1,0}$		$B_{1,0}$
2	0	AW	AP	AE	0	0	0	AN	0	0	0	0	0	0	0	0	0	0	$p_{2,0}$		$B_{2,0}$
3	0	0	AW	AP	AE	0	0	0	AN	0	0	0	0	0	0	0	0	0	$p_{3,0}$		$B_{3,0}$
⋮	0	0	0	AW	AP	0	0	0	0	AN	0	0	0	0	0	0	0	0	⋮		⋮
n	AS	0	0	0	0	AP	AE	0	0	0	AN	0	0	0	0	0	0	0	$p_{n,0}$		$B_{n,0}$
6	0	AS	0	0	0	AW	AP	AE	0	0	0	AN	0	0	0	0	0	0	$p_{0,1}$		$B_{0,1}$
7	0	0	AS	0	0	0	AW	AP	AE	0	0	0	AN	0	0	0	0	0	$p_{1,1}$		$B_{1,1}$
8	0	0	0	AS	0	0	0	AW	AP	AE	0	0	0	AN	0	0	0	0	$p_{2,1}$		$B_{2,1}$
⋮	0	0	0	0	AS	0	0	0	AW	AP	0	0	0	0	AN	0	0	0	$p_{3,1}$	=	$B_{3,1}$
n	0	0	0	0	0	AS	0	0	0	0	AP	AE	0	0	0	AN	0	0	⋮		⋮
11	0	0	0	0	0	0	AS	0	0	0	AW	AP	AE	0	0	0	AN	0	$p_{n,1}$		$B_{n,1}$
12	0	0	0	0	0	0	0	AS	0	0	0	AW	AP	AE	0	0	0	AN	$p_{0,n}$		$B_{0,n}$
13	0	0	0	0	0	0	0	0	AS	0	0	0	AW	AP	AE	0	0	0	$p_{1,n}$		$B_{1,n}$
⋮	0	0	0	0	0	0	0	0	0	AS	0	0	0	AW	AP	0	0	0	$p_{2,n}$		$B_{2,n}$
⋮	0	0	0	0	0	0	0	0	0	0	AS	0	0	0	0	⋱	⋱	0	$p_{3,n}$		$B_{3,n}$
⋮	0	0	0	0	0	0	0	0	0	0	0	AS	0	0	0	⋱	⋱	AE	⋮		⋮
n	0	0	0	0	0	0	0	0	0	0	0	0	AS	0	0	0	AW	AP	$p_{n,n}$		$B_{n,n}$

Figura 5.3 Matriz resultante de la Discretización del Yacimiento.

Para obtener los coeficientes de la matriz Penta diagonal se procede a aplicar la técnica de diferencias finitas por bloques centrados y las consideraciones del modelo físico conceptual a las ecuaciones para la saturación y presión , es decir (3.31) y (3.34) respectivamente

$$-\nabla\cdot\left[\bar{\bar{\mathrm{k}}}\lambda\nabla\mathrm{p_o}\right]+\nabla\cdot\left[\bar{\bar{\mathrm{k}}}\lambda_{\mathrm{w}}\frac{\partial\mathrm{p_c}}{\partial\mathrm{S_w}}\nabla\mathrm{S_w}\right]=\frac{\mathrm{q_w}}{\rho_{\mathrm{w}}}+\frac{\mathrm{q_o}}{\rho_{\mathrm{o}}} \tag{5.8}$$

$$\frac{\partial\phi\mathrm{S_w}}{\partial\mathrm{t}}-\nabla\cdot\left[\bar{\bar{\mathrm{k}}}\lambda_{\mathrm{w}}\nabla\mathrm{p_o}\right]+\nabla\cdot\left[\bar{\bar{\mathrm{k}}}\lambda_{\mathrm{w}}\frac{\partial\mathrm{p_c}}{\partial\mathrm{S_w}}\nabla\mathrm{S_w}\right]=\frac{\mathrm{q_w}}{\rho_{\mathrm{w}}} \tag{5.9}$$

aplicando la ecuación (2.19) a (5.8) y (5.9) y multiplicando ambas ecuaciones por ΔxΔyΔz y aplicando (2.21) a (5.9) utilizando un esquema θ implícito para la presión y un esquema explícito para la saturación y omitiendo los superíndices n y $n+1$ por simplicidad se llega a :

$$\begin{aligned}
&-\left\{\bar{\bar{k}}_{xx}\lambda\big|_e\left(\frac{p_{o,E}-p_{o,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda\big|_w\left(\frac{p_{o,P}-p_{o,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&-\left\{\bar{\bar{k}}_{yy}\lambda\big|_n\left(\frac{p_{o,N}-p_{o,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda\big|_s\left(\frac{p_{o,P}-p_{o,S}}{\delta y_s}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_e\left(\frac{S_{w,E}-S_{w,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_w\left(\frac{S_{w,P}-S_{w,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_n\left(\frac{S_{w,N}-S_{w,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_s\left(\frac{S_{w,P}-S_{w,S}}{\delta y_s}\right)\right\}\Delta x\Delta z\\
&=\left(\frac{q_w}{\rho_w}+\frac{q_o}{\rho_o}\right)
\end{aligned} \tag{5.10}$$

para la presión y:

$$\begin{aligned}
&\phi\frac{{S_w}^{n+1}-{S_w}^{n}}{\Delta t}\Delta x\Delta y\Delta z-\left\{\bar{\bar{k}}_{xx}\lambda_w\big|_e\left(\frac{p_{o,E}-p_{o,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda_w\big|_w\left(\frac{p_{o,P}-p_{o,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&-\left\{\bar{\bar{k}}_{yy}\lambda_w\big|_n\left(\frac{p_{o,N}-p_{o,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda_w\big|_s\left(\frac{p_{o,P}-p_{o,S}}{\delta y_s}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_e\left(\frac{S_{w,E}-S_{w,P}}{\delta x_e}\right)-\bar{\bar{k}}_{xx}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_w\left(\frac{S_{w,P}-S_{w,W}}{\delta x_w}\right)\right\}\Delta y\Delta z\\
&+\left\{\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_n\left(\frac{S_{w,N}-S_{w,P}}{\delta y_n}\right)-\bar{\bar{k}}_{yy}\lambda_w\frac{\partial p_c}{\partial S_w}\Big|_s\left(\frac{S_{w,P}-S_{w,S}}{\delta y_s}\right)\right\}\Delta x\Delta z\\
&=\frac{q_w}{\rho_w}
\end{aligned} \tag{5.11}$$

para la saturación. A continuación, se presta especial atención en los términos del lado derecho de las ecuaciones (5.10) y (5.11) porque el termino correspondiente a los gastos del agua y el aceite serán sustituidos por las ecuaciones (5.6) y (5.7) respectivamente del siguiente modo:

$$q_w = \frac{2\pi\rho_w k_{rw}\Delta z\sqrt{\bar{\bar{k}}_{xx}\bar{\bar{k}}_{yy}}}{\mu_w\,(\ln(r_e/r_{well}))}\left(P_{bhp,inj} - P_o + P_c\right) + \frac{2\pi\rho_w k_{rw}\Delta z\sqrt{\bar{\bar{k}}_{xx}\bar{\bar{k}}_{yy}}}{\mu_w\,(\ln(r_e/r_{well}))}\left(P_{bh,prod} - P_o + P_c\right) \tag{5.12}$$

$$q_o = \frac{2\pi\rho_w k_{rw}\Delta z\sqrt{\bar{\bar{k}}_{xx}\bar{\bar{k}}_{yy}}}{\mu_w\,(\ln(r_e/r_{well}))}\left(P_{bh} - P_o\right) \tag{5.13}$$

y además recordando que :

$$\frac{k_{rw}}{\mu_w} = \lambda_w \tag{5.14}$$

$$\frac{k_{ro}}{\mu_o} = \lambda_o \tag{5.15}$$

$$\frac{2\pi\Delta z\sqrt{\bar{\bar{k}}_{xx}\bar{\bar{k}}_{yy}}}{(\ln(r_e/r_{well}))} = WI \tag{5.16}$$

luego se sustituyen las ecuaciones (5.12), (5.13), (5.14), (5.15) y (5.16) en el lado derecho de las ecuaciones (5.10) y (5.11) según correspondan se obtiene :

$$\begin{aligned}\lambda_w WI|_{w,inj}\left(P_{bh,inj} + P_c\right)\Delta x\Delta y\Delta z + \lambda_o WI|_{o,prod}P_{bh,prod}\Delta x\Delta y\Delta z \\ + \lambda_w WI|_{w,prod}\left(P_{bh,prod} + P_c\right)\Delta x\Delta y\Delta z - \lambda_w WI|_{w,inj}P_{o,p}\Delta x\Delta y\Delta z \\ - \lambda_o WI|_{o,prod}P_{o,p}\Delta x\Delta y\Delta z - \lambda_w WI|_{w,prod}P_{o,p}\Delta x\Delta y\Delta z\end{aligned} \tag{5.17}$$

para el lado derecho de (5.10) y:

$$\begin{aligned}\lambda_w WI|_{w,inj}\left(P_{bh,inj} + P_c\right)\Delta x\Delta y\Delta z + \lambda_o WI|_{o,prod}P_{bh,prod}\Delta x\Delta y\Delta z \\ + \lambda_w WI|_{w,prod}\left(P_{bh,prod} + P_c\right)\Delta x\Delta y\Delta z - \lambda_w WI|_{w,inj}P_{o,p}\Delta x\Delta y\Delta z\end{aligned} \tag{5.18}$$

para (5.11). Posteriormente se desarrolla el lado izquierdo de las ecuaciones (5.10) y (5.11) y se deben pasar los términos del lado derecho que contengan $P_{o,p}$ al lado izquierdo para llevarlas a la forma general:

$$AP_{i,j}p_{i,j} = AE_{i,j}p_{i+1,j} + AW_{i,j}p_{i-1,j} + AN_{i,j}p_{i,j+1} + AS_{i,j}p_{i,j-1} + B_{i,j} \tag{5.19}$$

de la siguiente manera:

$$\left(\left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta x_e}\right|_e \Delta y\Delta z + \left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta x_w}\right|_w \Delta y\Delta z + \left.\frac{\bar{\bar{k}}_{yy}\lambda}{\delta y_n}\right|_n \Delta y\Delta z + \left.\frac{\bar{\bar{k}}_{yy}\lambda}{\delta y_s}\right|_s \Delta x\Delta z + \lambda_w WI|_{w,inj}\Delta x\Delta y\Delta z + \lambda_o WI|_{o,prod}\Delta x\Delta y\Delta z + \lambda_w WI|_{w,prod}\Delta x\Delta y\Delta z\right) P_{o,p} = + \left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta x_e}\right|_e \Delta y\Delta z\, p_{o,E} + \left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta x_w}\right|_w \Delta y\Delta z\, p_{o,W} + \left.\frac{\bar{\bar{k}}_{yy}\lambda}{\delta y_n}\right|_n \Delta x\Delta z\, p_{o,N} + \left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta y_s}\right|_s \Delta y\Delta z\, p_{o,S} + \left(\left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\frac{\partial p_c}{\partial S_w}\right|_e \Delta y\Delta z + \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_w}\frac{\partial p_c}{\partial S_w}\right|_w \Delta y\Delta z + \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\frac{\partial p_c}{\partial S_w}\right|_n \Delta x\Delta z + \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_s}\frac{\partial p_c}{\partial S_w}\right|_s \Delta x\Delta z\right) S_{w,P} - \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta x_w}\frac{\partial p_c}{\partial S_w}\right|_w \Delta y\Delta z\, S_{w,W} - \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\frac{\partial p_c}{\partial S_w}\right|_n \Delta x\Delta z\, S_{w,N} - \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\frac{\partial p_c}{\partial S_w}\right|_e \Delta y\Delta z\, S_{w,E} - \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta y_s}\frac{\partial p_c}{\partial S_w}\right|_s \Delta y\Delta z\, S_{w,S} + \lambda_w WI|_{w,inj}\left(P_{bh,inj} + P_c\right)\Delta x\Delta y\Delta z + \lambda_o WI|_{o,prod}P_{bh,prod}\Delta x\Delta y\Delta z + \lambda_w WI|_{w,prod}\left(P_{bh,prod} + P_c\right)\Delta x\Delta y\Delta z$$

para la ecuación de presión y :

$$S_w{}^{n+1} = S_w{}^{n} - \left[\left\{\left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\right|_e + \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_w}\right|_w\right\}\frac{\Delta t}{\Delta x\phi} + \left\{\left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\right|_n + \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_s}\right|_s\right\}\frac{\Delta t}{\Delta y\phi} + \lambda_w WI|_{w,inj}\frac{\Delta t}{\phi} + \lambda_w WI|_{w,prod}\frac{\Delta t}{\phi}\right]p_{o,P} + \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\right|_e \frac{\Delta t}{\Delta x\phi}\, p_{o,E} + \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_w}\right|_w \frac{\Delta t}{\Delta x\phi}\, p_{o,W} + \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\right|_n \Delta x\Delta z\, p_{o,N} + \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta y_s}\right|_s \Delta y\Delta z\, p_{o,S} + \left[\left\{\left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\frac{\partial p_c}{\partial S_w}\right|_e + \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_w}\frac{\partial p_c}{\partial S_w}\right|_w\right\}\frac{\Delta t}{\Delta x\phi} + \left\{\left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\frac{\partial p_c}{\partial S_w}\right|_n + \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_s}\frac{\partial p_c}{\partial S_w}\right|_s\right\}\frac{\Delta t}{\Delta y\phi}\right] S_{w,P} - \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\frac{\partial p_c}{\partial S_w}\right|_e \frac{\Delta t}{\Delta x\phi}\, S_{w,E} - \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta x_w}\frac{\partial p_c}{\partial S_w}\right|_w \frac{\Delta t}{\Delta x\phi}\, S_{w,W} - \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\frac{\partial p_c}{\partial S_w}\right|_n \frac{\Delta t}{\Delta y\phi}\, S_{w,N} - \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta y_s}\frac{\partial p_c}{\partial S_w}\right|_s \frac{\Delta t}{\Delta y\phi} S_{w,S} + \lambda_w WI|_{w,inj}\,\frac{\Delta t}{\phi}\{P_{bhp,inj} + P_c\} + \lambda_w WI|_{w,prod}\frac{\Delta t}{\phi}\,\{P_{bh,prod} + P_c\}$$

para la saturación, además, se definen las transmisibilidades del agua y del aceite entre celdas como:

$$T_E = \left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta x_e}\right|_e \Delta y\Delta z; T_W = \left.\frac{\bar{\bar{k}}_{xx}\lambda}{\delta x_w}\right|_w \Delta y\Delta z; T_N = \left.\frac{\bar{\bar{k}}_{yy}\lambda}{\delta y_n}\right|_n \Delta x\Delta z; T_S = \left.\frac{\bar{\bar{k}}_{yy}\lambda}{\delta y_s}\right|_s \Delta x\Delta z\ ; T_{E,w} = \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_e}\right|_e \Delta y\Delta z;\ T_{W,w} = \left.\frac{\bar{\bar{k}}_{xx}\lambda_w}{\delta x_w}\right|_w \Delta y\Delta z; T_{N,w} = \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_n}\right|_n \Delta x\Delta z; T_{S,w} = \left.\frac{\bar{\bar{k}}_{yy}\lambda_w}{\delta y_s}\right|_s \Delta x\Delta z$$

y recalcando que $P = i, j, E = i + 1, j$, $W = i - 1, j$, $N = i, j + 1$ $S = i, j - 1$, $e = i + {}^1/_2, j$, $w = i - {}^1/_2, j$, $n = i, j + {}^1/_2$ $s = i, j - {}^1/_2$ y $\Delta x\Delta y\Delta z = \Delta V$.

Por lo que finalmente los coeficientes de la presión y la saturación se definen como sigue:

los coeficientes de la ecuación de presión quedan:

$$AP_{i,j} = T_E + T_W + T_N + T_S + \left(\lambda_w WI|_{w,inj} + \lambda_o WI|_{o,prod} + \lambda_w WI|_{w,prod}\right)\frac{\Delta t}{\phi} \quad (5.20)$$

$$AE_{i,j} = T_E \quad (5.21)$$

$$AW_{i,j} = T_W \quad (5.22)$$

$$AN_{i,j} = T_N \quad (5.23)$$

$$AS_{i,j} = T_S \quad (5.24)$$

$$\begin{aligned} Bij = &-\left[T_{E,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{e}}\frac{\Delta t}{\Delta x\phi}\left(\mathrm{S_{w,E}} - \mathrm{S_{w,P}}\right) - T_{W,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{w}}\frac{\Delta t}{\Delta x\phi}\left(\mathrm{S_{w,P}} - \mathrm{S_{w,W}}\right)\right] \\ &-\left[T_{N,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{n}}\frac{\Delta t}{\Delta y\phi}\left(\mathrm{S_{w,N}} - \mathrm{S_{w,P}}\right) - T_{S,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{s}}\frac{\Delta t}{\Delta y\phi}\left(\mathrm{S_{w,P}} - \mathrm{S_{w,S}}\right)\right] \\ &+\left(\lambda_w WI|_{w,inj}P_{bh,inj} + \lambda_w WI|_{w,prod}P_{bh,prod} + \lambda_o WI|_{o,prod}P_{bh,prod}\right. \\ &\left.+\lambda_w WI|_{w,inj}P_c + \lambda_w WI|_{w,prod}P_c\right)\frac{\Delta t}{\phi} \end{aligned} \quad (5.25)$$

y los coeficientes de la saturación quedan:

$$AP_{i,j} = \left(T_{E,w} + T_{W,w}\right)\frac{\Delta t}{\Delta x \phi} + \left(T_{N,w} + T_{S,w}\right)\frac{\Delta t}{\Delta y \phi} + \left(\lambda_w WI|_{w,inj} + \lambda_o WI|_{o,prod}\right)\frac{\Delta t}{\phi} \tag{5.26}$$

$$AE_{i,j} = T_{E,w}\frac{\Delta t}{\Delta x \phi} \tag{5.27}$$

$$AW_{i,j} = T_{W,w}\frac{\Delta t}{\Delta x \phi} \tag{5.28}$$

$$AN_{i,j} = T_{N,w}\frac{\Delta t}{\Delta y \phi} \tag{5.29}$$

$$AS_{i,j} = T_{S,w}\frac{\Delta t}{\Delta y \phi} \tag{5.30}$$

$$\begin{aligned} Bij = &-\left[T_{E,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{e}}\left(\mathrm{S_{w,E}} - \mathrm{S_{w,P}}\right) - T_{W,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{w}}\left(\mathrm{S_{w,P}} - \mathrm{S_{w,W}}\right)\right] \\ &-\left[T_{N,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{n}}\left(\mathrm{S_{w,N}} - \mathrm{S_{w,P}}\right) - T_{S,w}\frac{\partial \mathrm{p_c}}{\partial \mathrm{S_w}}\Big|_{\mathrm{s}}\left(\mathrm{S_{w,P}} - \mathrm{S_{w,S}}\right)\right] \\ &+\left(\lambda_w WI|_{w,inj}P_{bh,inj} + \lambda_w WI|_{w,prod}P_{bh,prod} + \lambda_o WI|_{o,prod}P_{bh,prod}\right. \\ &\left.+\lambda_w WI|_{w,inj}P_c + \lambda_w WI|_{w,prod}P_c\right)\frac{\Delta t}{\phi} \end{aligned} \tag{5.31}$$

ANEXO D. Implementación Computacional

Los Simuladores desarrollados en esta tesis se realizaron basados en los algoritmos IMPES Clásico e IMPES Mejorado. El simulador basado en el primer algoritmo está constituido por el menú principal denominado *MAIN_IMPES* en el que se definen las propiedades físicas del dominio y de los fluidos, para aplicar los factores de conversión correspondientes, después se define el tiempo de simulación y paso de tiempo, luego se genera la malla y sus centros, también se definen los valores iniciales para las variables primarias (presión inicial del yacimiento, saturación inicial, porosidad, el cambio de saturación, las permeabilidades en los ejes x y y, etc.) como Matrices, se definen los vectores vacíos, para guardar los gastos de producción y el flujo fraccional, por último se definen los parámetros que utilizan los *solvers* para su funcionamiento y los *frames* o listas para salvar las imágenes, igualmente una instrucción para la creación de un archivo y poder guardar los datos en un archivo de *Excel* . En las siguientes líneas comienza el algoritmo IMPES mediante un ciclo *while* de acuerdo con su estructura general (Figura 4.1) dentro del bucle se hacen llamar las bibliotecas integradas a *Python* como lo son *numpy*, *matplotlib.pyplot*, *PIL*, pandas, *openpyxl* , *pylab*, *Coefficients2D_IM* para calcular los coeficientes de la ecuación de presión que son las ecuaciones (5.20), (5.21), (5.22), (5.23), (5.24) y (5.25) descritos en el ANEXO C. La biblioteca *Coefficients2D_IM* se integran las funciones *upwind_2D*( ) que es la implementación de un esquema *upwin* para poder calcular las saturaciones en las caras de los bloques, *harmonic_average2D*( ) que calcula un promedio armónico para la permeabilidad absoluta en cualquier dirección del yacimiento, esta función sirve para los casos isotrópico y anisotrópico, *press_CoeffBL2D*( ) que como ya se mencionó es específicamente para calcular los coeficientes de la presión, *satw_CoeffBL2D*( ), que calcula los coeficientes para la ecuación de saturación definida por las ecuaciones (5.26), (5.27), (5.28), (5.29), (5.30) y (5.31), de la misma manera la función *characterizationData_2D*( ) que se utiliza para calcular los gastos de los fluidos producidos y el corte de agua dictados por las ecuaciones (5.6) para el agua, (5.7) para el aceite y (3.19) para el flujo fraccional del agua, y finalmente la función *csr_matrix_format*( ) que cumple el objetivo de crear la matriz dispersa cuya implementación se muestra en la Figura 5.6. *Coefficients2D_IM* de igual modo llama la biblioteca *fluidproperties* en la cual se calculan las propiedades de los fluidos determinadas por las ecuaciones (4.3) y (4.4) para las permeabilidades relativas y (4.5) para la presión capilar mediante las *funciones krw_2D*( ), *kro_2D*( ), *kro_centerPoint*( ), *pc_centerPoint*( ) y pc_*2D*( ) respectivamente, remarcando que las que se identifican con *centerPoint son para los nodos en los nodos internos y las otras en las caras de las celdas*. *MAIN_IMPES* llama las funciones *csr_matrix_format* para poder resolver la presión para posteriormente poder utilizar *satw_CoeffBL2D*( ) y calcular la saturación. Se actualiza el paso de tiempo y termina el bucle y se encuentran los comandos para las gráficas y almacenamiento de datos y finaliza el programa. Para crear el simulador con el segundo algoritmo se utilizan las mismas bibliotecas y se implementa la función *dSw_dt( )* la cual recalcula el paso de tiempo para la saturación atribuido a la ecuación (3.53). La Figura 5.4 y la Figura 5.5 muestran el algoritmo de las implementaciones basadas en los IMPES Clásico y Mejorado respectivamente.

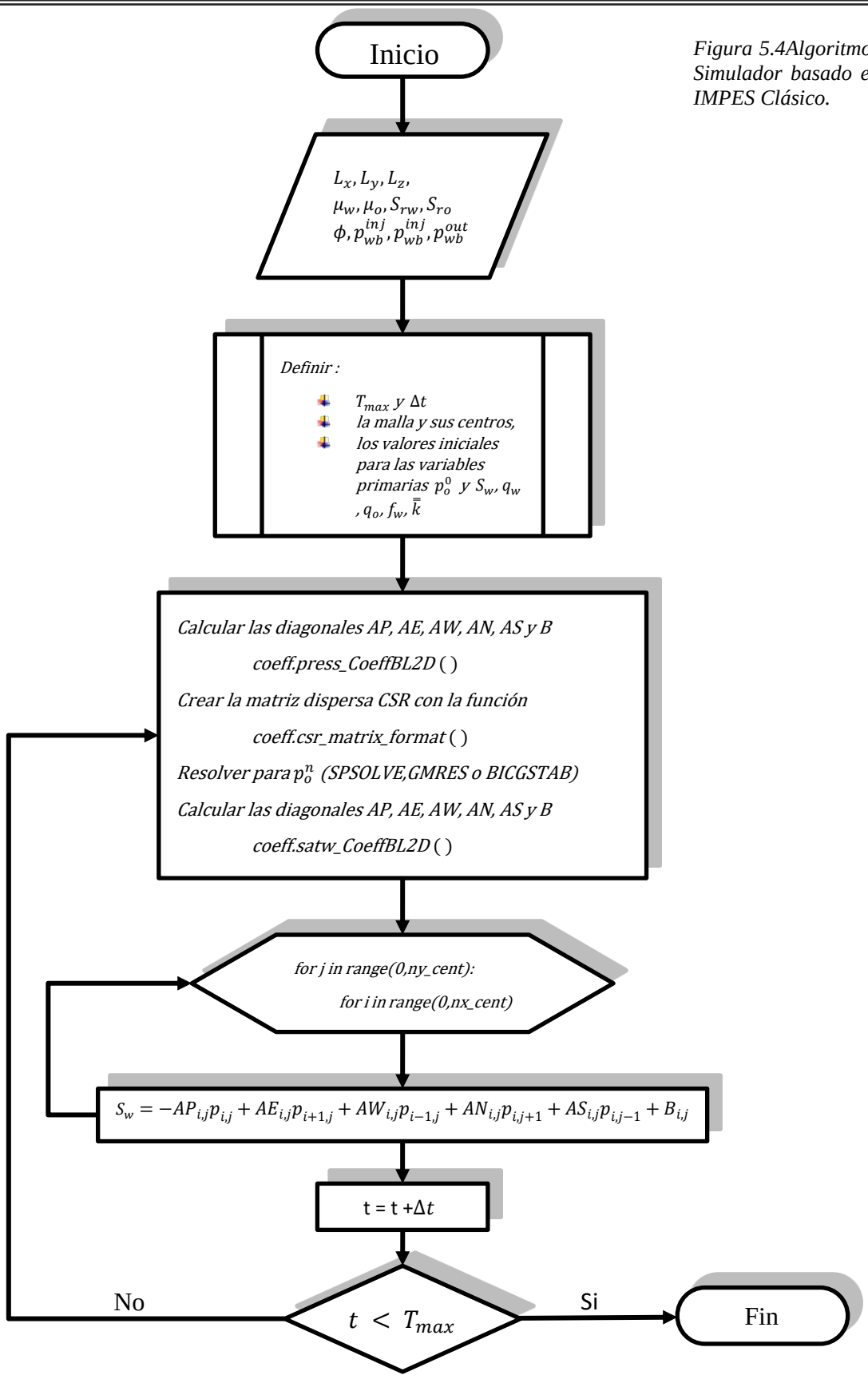


Figura 5.4Algoritmo del Simulador basado en el IMPES Clásico.

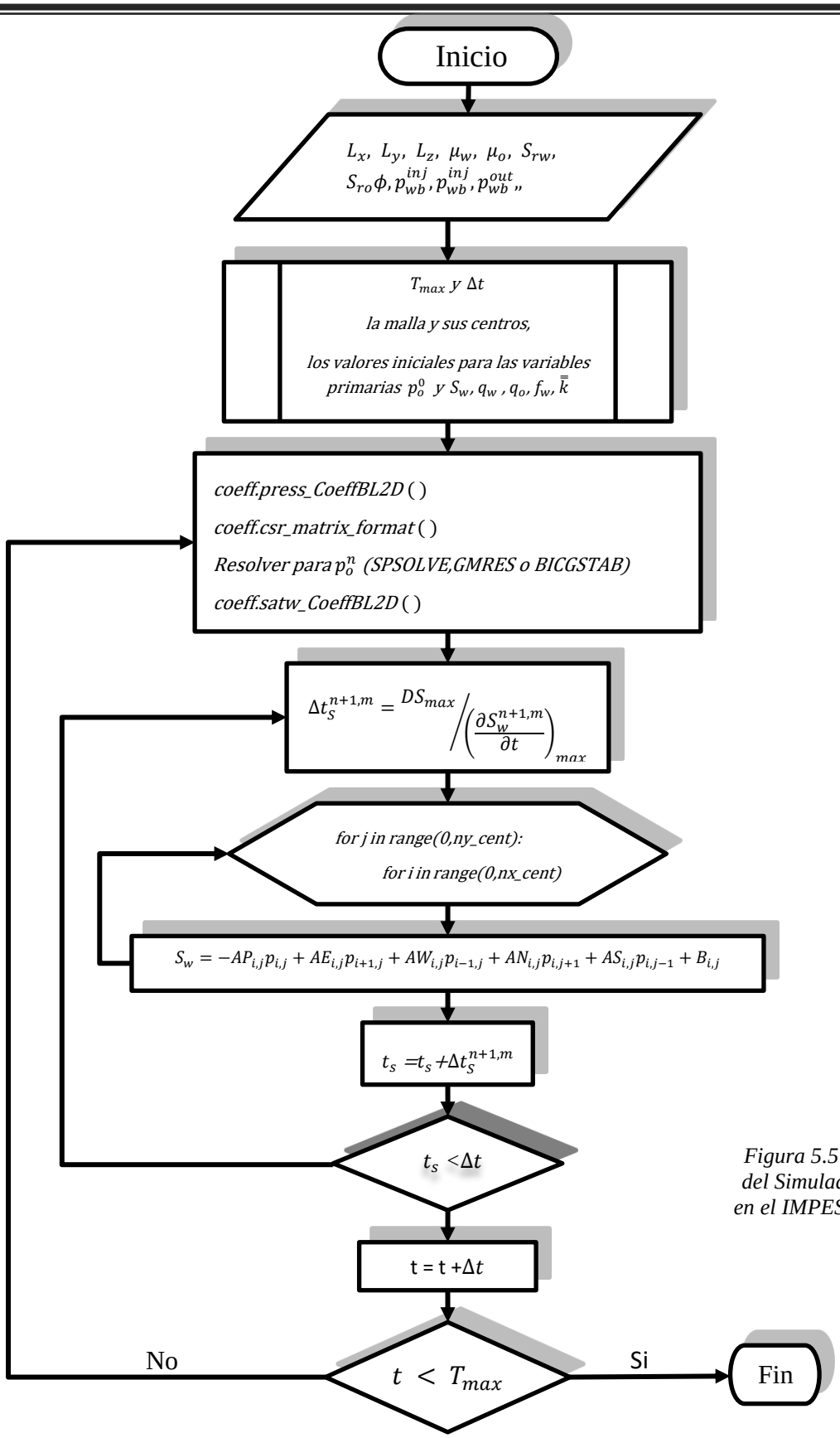


Figura 5.5 Algoritmo del Simulador basado en el IMPES Mejorado.

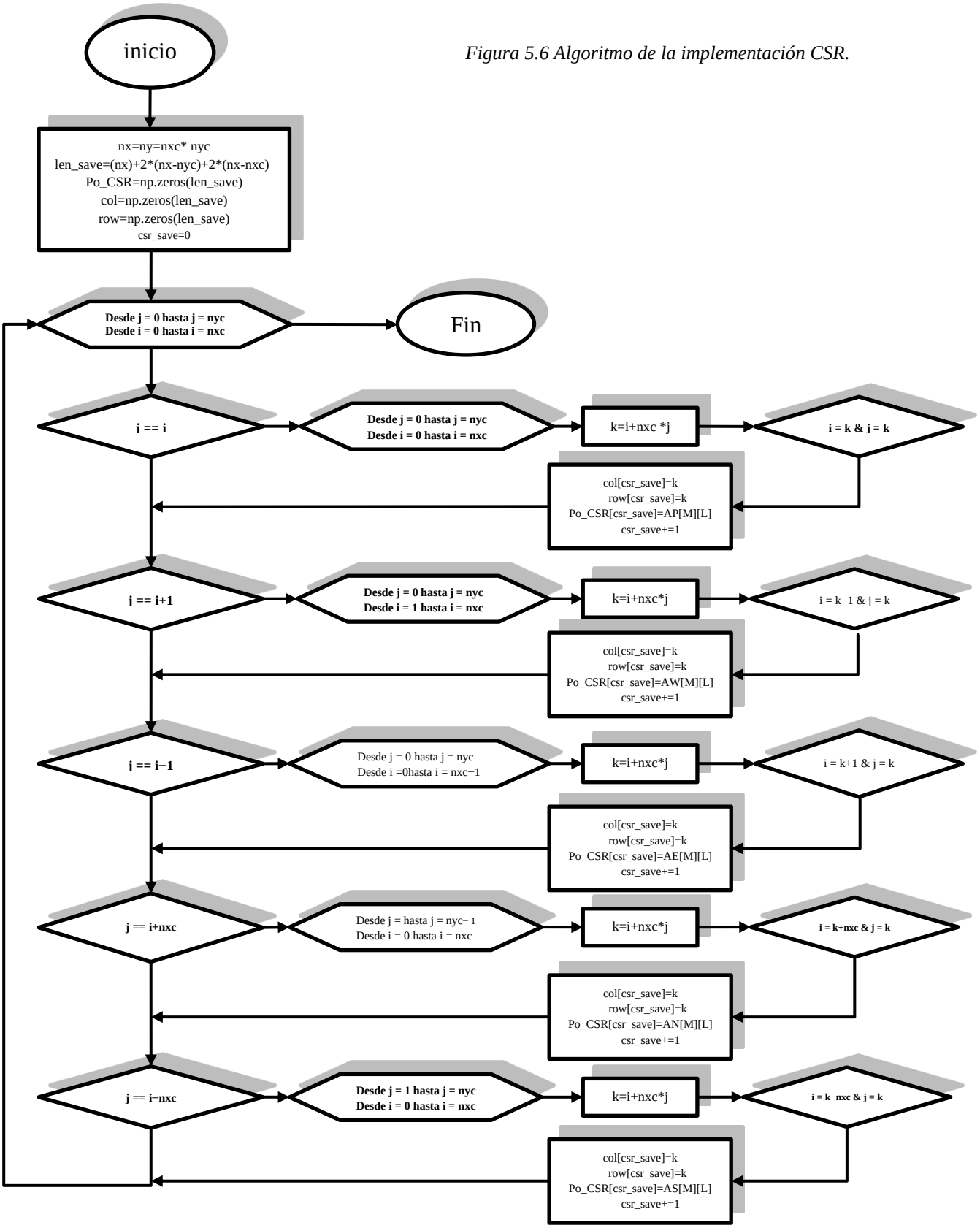


Figura 5.6 Algoritmo de la implementación CSR.