



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIAS DE LA TIERRA
INSTITUTO DE GEOFÍSICA

Modelado Numérico de Procesos Térmicos de Recuperación Mejorada de Hidrocarburos

TESIS

QUE PARA OPTAR POR EL GRADO DE:
DOCTOR EN CIENCIAS DE LA TIERRA

PRESENTA:

VÍCTOR LEONARDO TEJA JUÁREZ

DIRECTOR DE TESIS:

Dr. Luis Miguel de la Cruz Salas
Instituto de Geofísica

COMITÉ TUTOR:

Dr. Ismael Herrera Revilla
Instituto de Geofísica

Dr. Simón López Ramírez
Facultad de Ingeniería

Ciudad Universitaria, Cd. Mx., Diciembre de 2018



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Esta página se dejó en blanco intencionalmente.

Dedicatoria

*A mi esposa Yessi Espinosa
y a nuestro beboto que lleva en su vientre...*

Agradecimientos

Gracias a la entidad superior que llamamos “Dios” por hacer posible la existencia misma, quizá algún día conoceremos con certeza nuestro origen, aún si llega ese día, siempre estaré agradecido por habitar en este espacio y en este tiempo.

A mi esposa Yessica, por su apoyo incondicional y confianza para concluir estos estudios.

A mi Familia, por su apoyo moral y sus buenos consejos, ya sea de mis padres o mis hermanos.

Al Dr. Luis Miguel de la Cruz, porque más que un tutor, es un buen amigo ¡muchas gracias!

A mi comité tutor el Dr. Ismael Herrera y el Dr. Simón López, por sus enseñanzas, observaciones y por estar al pendiente en todo momento del desarrollo de este trabajo de tesis, ¡gracias por todo!

Al comité revisor: Dra. Leticia Flores Márquez, Dr. Eric Morales Casique y Dr. Josué Tago Pacheco, por sus comentarios e importantes sugerencias durante la revisión de la tesis.

A los investigadores del grupo de Modelación Matemática Computacional: Dra. Graciela Herrera, Dr. Guillermo Hernández y Dr. Norberto Vera, por los consejos y enseñanzas en los cursos del posgrado y durante el desarrollo de este trabajo.

A mis amigos del posgrado Marian, Mirce, Esther, Flor, Alberto, German e Imre, por los buenos momentos vividos durante esta etapa de mi vida.

Al Instituto de Geofísica de nuestra máxima casa de estudios: la Universidad Nacional Autónoma de México, por brindarme la oportunidad de realizar un sueño más.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el apoyo económico brindado durante los estudios de doctorado.

Índice general

Lista de figuras	VI
Lista de tablas	X
Abstract	XI
Resumen	XII
1. Introducción	1
1.1. Ubicación del problema a resolver	1
1.2. Arquitectura Unificada de Dispositivos de Cómputo (CUDA)	4
1.3. Objetivos de este trabajo	8
1.4. Aportaciones de este trabajo	9
1.5. Estructura de la tesis	10
2. Procesos térmicos para la Recuperación Mejorada de Hidrocarburos	11
2.1. Descripción de los procesos térmicos para la recuperación mejorada de hidrocarburos	11
2.1.1. Procesos térmicos por inyección de fluido caliente	12
2.1.2. Combustión <i>in situ</i>	16
2.2. Estado del arte de los modelos numéricos para simular los procesos térmicos de inyección de vapor	17
3. Flujo de fluidos en medios porosos	22
3.1. Sistemas multifásicos en medios porosos	22
3.1.1. Ecuación general de balance de masa	22
3.1.2. Ecuación general de balance de energía	25
4. Transformación de las ecuaciones	28
4.1. Conceptos básicos del mapeo de una región arbitraria a una región regular . .	28
4.2. Transformación de las ecuaciones gobernantes del flujo de fluidos en medios porosos	31
4.2.1. Sistemas de coordenadas curvilíneas	31

4.2.2.	Transformación de la ecuación general de balance de masa en medios porosos	34
4.2.3.	Transformación de la ecuación de balance de energía en medios porosos	38
5.	Modelado numérico del proceso de inyección de agua	40
5.1.	Formulación Presión-Saturación	40
5.2.	Modelo conceptual para flujo incompresible	43
5.3.	Modelo numérico en coordenadas cartesianas	44
5.3.1.	Discretización de la ecuación de saturación	44
5.3.2.	Método de Newton-Raphson para resolver sistemas de ecuaciones diferenciales parciales	48
5.3.3.	Solución simultánea de las ecuaciones de presión y saturación	49
5.3.4.	Implementación computacional	51
5.4.	Validación del código de inyección de agua	55
5.4.1.	Buckley-Leverett	55
5.4.2.	Problema de los 5 pozos	61
5.4.3.	Pozos horizontales: The Seventh SPE Project	67
5.5.	Modelo numérico en dominios no ortogonales	72
5.5.1.	Transformación de la ecuación de saturación de agua	72
5.5.2.	Transformación de la ecuación de presión	74
5.5.3.	Resultados de la simulación numérica en dominios no ortogonales	76
5.6.	Experimentos numéricos de desempeño	85
6.	Modelado numérico del proceso de inyección de vapor	87
6.1.	Modelo matemático del proceso térmico de inyección de vapor	87
6.2.	Modelo numérico del proceso de inyección de vapor en coordenadas cartesianas	89
6.2.1.	Planteamiento del algoritmo de solución	89
6.2.2.	Discretización de la ecuaciones	91
6.2.3.	Metodología de linealización y solución de los sistemas de ecuaciones diferenciales	94
6.2.4.	Implementación computacional	97
6.3.	Resultados en coordenadas cartesianas para la simulación del proceso de inyección de vapor	99
6.3.1.	Validación del código numérico de inyección de vapor	99
6.3.2.	Comparación del volumen de recuperación de aceite pesado mediante inyección de agua caliente versus inyección de vapor	105
6.4.	Modelo numérico en dominios no ortogonales	112
6.4.1.	Resultados de la simulación numérica en dominios no ortogonales	112
6.5.	Experimentos numéricos de desempeño	121
7.	Conclusiones	123

A. Discretización de las ecuaciones transformadas	126
A.1. Inyección de agua	126
A.1.1. Discretización de la ecuación de saturación en el plano computacional	126
A.1.2. Discretización de la ecuación de presión en el plano computacional . .	130
A.2. Inyección de vapor	131
A.2.1. Discretización de la ecuación general de balance de masa en el plano computacional	131
A.2.2. Discretización de la ecuación de balance de energía en el plano computacional	133
B. Ecuaciones constitutivas para el modelo de inyección de vapor	136
B.1. Propiedades de flujo	136
B.2. Propiedades del aceite	137
B.3. Propiedades del agua	138
B.4. Propiedades del vapor	139
B.5. Modelado de pozos	140
B.6. Término de condensación- evaporación M_g	141
C. Programación utilizando CUDA	142

Lista de figuras

1.1. Esquema de las etapas de recuperación de hidrocarburos.	2
2.1. Perfiles de saturación y temperatura en el proceso de inyección de agua caliente.	12
2.2. Saturaciones de aceite residual y agua versus la temperatura	13
2.3. Perfil de temperatura en el desplazamiento por inyección de agua caliente.	13
2.4. Perfil de saturación en el desplazamiento por vapor.	14
2.5. Proceso de drenado gravitacional asistido por inyección de vapor.	16
4.1. Mapeo de una región irregular simplemente conectada en la región computacional.	29
4.2. Mapeo de una región irregular doblemente conectada en una región rectangular.	30
4.3. Mapeo de una región doblemente conectada en una región simplemente conectada usando un corte divisorio.	31
4.4. Sistema de coordenadas curvilíneas.	32
5.1. Estencil tridimensional de volumen finito $\Delta V = \Delta x \Delta y \Delta z$	45
5.2. Sistema de ecuaciones lineales para las variables primarias p_o y S_w	50
5.3. Pseudocódigo general	51
5.4. Función para calcular el bloque del Jacobiano $\partial R_o / \partial p_o$ en la CPU	53
5.5. Función para calcular el bloque del Jacobiano $\partial R_o / \partial p_o$ en la GPU	53
5.6. Uso de la memoria de la GPU para el almacenamiento de variables	54
5.7. Procedimiento para envolver el Jacobiano utilizando la biblioteca Thrust.	55
5.8. Medio homogéneo de longitud L_x , inicialmente saturado con aceite y agua	56
5.9. Sistema de ecuaciones lineales para el problema de Buckley-Leverett.	56
5.10. a) Perfiles de saturación S_w y b) perfiles de presión p_o , para un tamaño de malla de 100 volúmenes.	58
5.11. a) Perfiles de saturación S_w y b) perfiles de presión p_o , para un tamaño de malla de 1,000 volúmenes.	59
5.12. a) Perfiles de saturación S_w y b) perfiles de presión p_o , para un tamaño de malla de 10,000 volúmenes.	59
5.13. Dominio físico para el problema "Five Spot".	61
5.14. Sistema de ecuaciones lineales para el problema "Five Spot".	62
5.15. Perfiles de saturación S_w para un tamaño de malla de 90x90 celdas.	64

5.16. Perfiles de presión de la fase aceite p_o (psi) para un tamaño de malla de 90x90 celdas.	65
5.17. Campo de velocidad de la fase agua para un tamaño de malla de 90x90 celdas.	66
5.18. a) Producción de fluidos y b) flujo fraccional F_w . Resultados reportados por Chen, línea punteada, y los obtenidos en este trabajo, línea continua.	67
5.19. Dominio físico para el problema de los pozos horizontales.	68
5.20. Perfiles de saturación de agua S_w en la capa de producción.	70
5.21. Perfiles de presión de la fase aceite p_o (psi) en la capa de producción.	71
5.22. a) Curvas de producción de aceite y b) producción de aceite acumulado. Resultados reportados por Chen, línea punteada, y los obtenidos en este trabajo, línea continua.	72
5.23. a) Curvas de producción de fluidos y b) flujo fraccional F_w	76
5.24. Centros de los volúmenes de control del dominio no ortogonal no. 1.	77
5.25. Perfiles de saturación S_w para a) 250 y b) 750 días de simulación.	77
5.26. Vectores de velocidad de la fase agua para a) 250 y b) 750 días de simulación.	78
5.27. Perfiles de presión p_o (psi) para a) 250 y b) 750 días de simulación.	78
5.28. a) Curvas de producción de fluidos y b) flujo fraccional F_w	79
5.29. Centros de los volúmenes de control del dominio no ortogonal no. 2.	79
5.30. Perfiles de saturación S_w para a) 100 y b) 500 días de simulación.	80
5.31. Vectores de velocidad de la fase agua para a) 100 y b) 500 días de simulación.	80
5.32. Perfiles de presión p_o (psi) para a) 100 y b) 500 días de simulación.	81
5.33. a) Curvas de producción de fluidos y b) flujo fraccional F_w	81
5.34. Centros de los volúmenes de control del dominio no ortogonal 3	82
5.35. Perfiles de saturación S_w para a) 100 y b) 500 días de simulación.	82
5.36. Vectores de velocidad de la fase agua para a) 100 y b) 500 días de simulación.	83
5.37. Perfiles de presión p_o (psi) para a) 100 y b) 500 días de simulación.	84
5.38. a) Curvas de producción de fluidos y b) flujo fraccional F_w	84
5.39. a) Tiempo de cómputo total de la CPU y la GPU, y b) aceleración lograda mediante el uso de la GPU.	86
6.1. Diagrama de flujo para la implementación del código numérico del inyección de vapor.	90
6.2. Sistema de ecuaciones lineales para las variables primarias p_o , S_w y S_g	95
6.3. Sistema de ecuaciones lineales para resolver el balance de energía con la variable primaria T	96
6.4. Pseudocódigo general del código numérico de inyección de vapor	97
6.5. Modelo físico para la simulación de inyección de vapor a escala de núcleo.	99
6.6. Comparación del perfil de temperatura a escala de núcleo.	101
6.7. Comparación del aceite producido en el núcleo en volúmenes porosos (VP).	102
6.8. Perfiles obtenidos para la simulación de inyección de vapor en un núcleo utilizando 10 volúmenes de control.	103
6.9. Perfiles obtenidos para la simulación de inyección de vapor en un núcleo utilizando 30 volúmenes de control.	104

6.10. Modelo físico para el problema de los nueve pozos.	105
6.11. Perfiles obtenidos en la simulación de inyección de agua caliente para un tiempo de 500 días.	107
6.12. Perfiles obtenidos en la simulación de inyección de vapor para un tiempo de 500 días.	108
6.13. Recuperación de aceite mediante inyección de agua caliente en los pozos más cercano (1) y más lejano (3).	109
6.14. Recuperación de aceite mediante la inyección de vapor en los pozos más cercano (1) y más lejano (3).	110
6.15. Flujo fraccional obtenido mediante inyección de agua caliente en los pozos más cercano (1) y más lejano (3).	110
6.16. Recuperación de aceite y corte de vapor para los pozos más cercano (1) y más lejano (3), en la inyección de vapor.	111
6.17. Curvas de aceite producido para los pozos más cercano (1) y más lejano (3).	112
6.18. Centros de los volúmenes de control del dominio no ortogonal no. 1.	113
6.19. Perfiles obtenidos para las variables primarias en un tiempo de simulación de 810 días después de iniciar la inyección de vapor en el dominio no ortogonal no. 1.	114
6.20. Perfiles obtenidos para las variables primarias en un tiempo de simulación de 1,350 días después de iniciar la inyección de vapor en el dominio no ortogonal no. 1.	115
6.21. Curvas de flujo fraccional F_w para los pozos (1) y (2) en el dominio no ortogonal no. 1.	116
6.22. Curvas de aceite producido para los pozos (1) y (2) en el dominio no ortogonal no. 1.	116
6.23. Centros de los volúmenes de control del dominio no ortogonal no. 2.	117
6.24. Perfiles obtenidos para las variables primarias en un tiempo de simulación de 300 días después de iniciar la inyección de vapor en el dominio no ortogonal no. 2.	118
6.25. Perfiles obtenidos para las variables primarias en un tiempo de simulación de 600 días después de iniciar la inyección de vapor en el dominio no ortogonal no. 2.	119
6.26. Curvas de flujo fraccional F_w para los pozos (1) y (2) en el dominio no ortogonal no. 2.	120
6.27. Curvas de aceite producido para los pozos (1) y (2) en el dominio no ortogonal no. 2.	120
6.28. a) Tiempo de cómputo total de la CPU y la GPU, y b) aceleración lograda mediante el uso de la GPU.	122
A.1. Volumen de control elemental en el plano transformado.	127
C.1. Función kernel para la suma de matrices en CUDA	142
C.2. Hilos de calculo (threads) en un kernel.	143

C.3. Estructura de un código serie-paralelo en CUDA	144
---	-----

Lista de tablas

5.1.	Parámetros para la simulación del problema de Buckley-Leverett.	57
5.2.	Permeabilidades relativas k_{rw} y k_{ro}	57
5.3.	Valores obtenidos de S_w para diferentes tiempos de simulación.	60
5.4.	Parámetros para la simulación del problema de los 5 pozos.	62
5.5.	Permeabilidades relativas k_{rw} , k_{ro} y presión capilar p_{cow}	69
5.6.	Parámetros para la simulación del problema <i>The Seventh SPE Project</i>	69
5.7.	Tiempo promedio obtenido para el cálculo del Jacobiano.	85
5.8.	Tiempo de cómputo promedio obtenido para la solución del sistema de ecuaciones lineales mediante el método BICGSTAB con el preconditionador de Jacobi.	86
6.1.	Parámetros para simular la inyección de vapor en un núcleo saturado inicialmente de agua y aceite.	100
6.2.	Permeabilidades relativas para los sistemas agua-aceite y gas-aceite.	100
6.3.	Parámetros para simular el problema simétrico de los 9 pozos.	106
6.4.	Tiempo promedio obtenido para el cálculo del Jacobiano del balance de masa.	121
6.5.	Tiempo de cómputo promedio obtenido para la solución del sistema de ecuaciones lineales mediante el método GMRES.	122

Abstract

In this work numerical simulations of isothermal water injection and steam injection processes for hydrocarbons recovery are presented. The mathematical model of water injection is based on the pressure-saturation formulation which simplifies the numerical solution of two phase flow in porous media. The mathematical model of steam injection, is obtained from the mass and energy balance equations. The numerical method used to transform both mathematical models into a set of discrete equations is the finite volume method (FVM). Due to the non-linearity of both models, the Newthou-Raphson (NR) technique is applied to linearize the partial differential equations. In the case of steam injection, a new method, based on the one described by Shutler [1], was proposed and implemented. The two-dimensional examples are solved in both box-shaped and irregular domains. The latter requires the structured mesh generation and the transformation of the equations by using curvilinear coordinates. In order to reduce the computing time for a large number of unknowns, an implementation on graphical processing units (GPUs) is presented. The main strategy is based on the construction of the Jacobian matrix directly in the GPU, which greatly reduces the exchange of information between the CPU and the GPU, resulting in better speedups. Also, the linear system is solved inside the GPU and only the final results are sent back to the CPU. In order to calibrate and validate the numerical results, a comparison against benchmarks reported by other authors was made obtaining very good agreement. On the other hand, the speedup obtained by using GPUs shows a good performance for examples of water and steam injection, in two dimensions. Additionally, the method proposed for the numerical solution of steam injection seems to be a good option for these kind of problems.

Resumen

En este trabajo se presentan los desarrollos para llevar a cabo las simulaciones numéricas de los procesos de inyección de agua isotérmica e inyección de vapor para la recuperación de hidrocarburos. El modelo matemático de inyección de agua se basa en la formulación de presión-saturación, que simplifica la solución numérica del flujo de dos fases en medios porosos. El modelo matemático de inyección de vapor es obtenido a partir de las ecuaciones de balance de masa y energía. El método numérico de volumen finito (MVF), se utiliza para transformar ambos modelos matemáticos en un conjunto de ecuaciones discretas. Debido a la no linealidad de ambos modelos, se aplica la técnica de Newton-Raphson (NR) para linealizar las ecuaciones diferenciales parciales. En el caso de la inyección de vapor, se propone e implementa un nuevo método, el cual está basado en el método descrito por Shutler [1]. Considerando los casos bidimensionales, los ejemplos se resuelven tanto en el plano cartesiano como en dominios irregulares. En éste último, se requiere la generación de mallas estructuradas y la transformación de las ecuaciones utilizando coordenadas curvilíneas. Con el objetivo de reducir el tiempo de cómputo para un gran número de incógnitas, se presenta la implementación de los códigos numéricos utilizando unidades de procesamiento gráfico (GPU's). La estrategia principal se basa en la construcción de la matriz jacobiana directamente en la GPU, para reducir en gran medida el intercambio de información entre la CPU y la GPU, lo cual se traduce en mejores aceleraciones. También, el sistema lineal se resuelve dentro de la GPU y sólo los resultados finales se envían de vuelta a la CPU. Con el fin de calibrar y validar los resultados numéricos, se llevaron a cabo comparaciones con los *benchmarks* reportados por otros autores, obteniendo muy buena concordancia con los mismos. Por otra parte, la aceleración obtenida utilizando las GPU's muestra un buen rendimiento para los ejemplos de inyección de agua y vapor, en dos dimensiones. Por último, el método propuesto para la solución numérica de inyección de vapor parece ser una buena opción para este tipo de problemas.

Capítulo 1

Introducción

1.1. Ubicación del problema a resolver

La recuperación mejorada de petróleo (*Enhanced Oil Recovery*, EOR) se define como la recuperación de hidrocarburos por inyección de materiales que no se encuentran naturalmente dentro de un yacimiento [2]. Esta definición cubre la mayoría de las formas de recuperación por inyección. En la vida productiva de un yacimiento existen tres etapas de extracción o recuperación de petróleo: primaria, secundaria y terciaria. En la etapa primaria el recobro de crudo se lleva a cabo mediante mecanismos naturales, por ejemplo, recuperación por la presión natural del yacimiento. En la recuperación secundaria se encuentran las técnicas de inyección de agua o gas, las cuales tienen el propósito de elevar o mantener la presión del yacimiento. Por último, la recuperación terciaria es cualquier técnica aplicada después de la recuperación secundaria. Dentro de este contexto se debe señalar que la definición de EOR abarca la recuperación terciaria, puesto que en la etapa terciaria se considera la inyección de surfactantes, gases miscibles e inmiscibles, aire o vapor (procesos térmicos) y otras sustancias que colaboren a mejorar la extracción de petróleo [2] (ver Figura 1.1).

En México es un hecho la declinación de producción de hidrocarburos por recuperación primaria y secundaria (véase [3]). Por este motivo, el estudio y análisis de las técnicas de recuperación mejorada es un área de oportunidad para colaborar en el desarrollo de tecnología para la extracción de hidrocarburos en los campos maduros de México a mediano y largo plazo. Las técnicas o métodos de recuperación mejorada se dividen en cuatro grupos:

- Métodos térmicos.
- Métodos de inyección de gases miscibles e inmiscibles.
- Métodos de inyección de químicos.
- Otros.

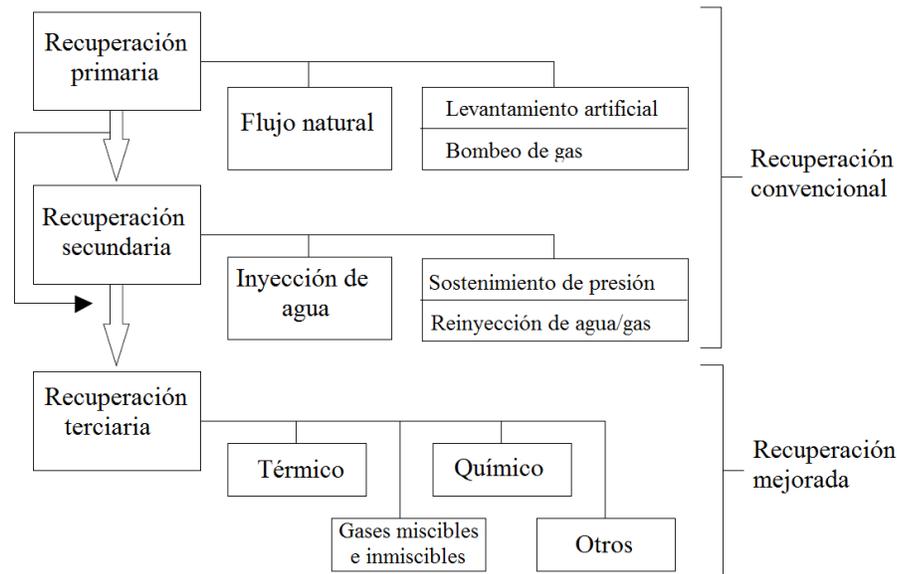


Figura 1.1: Esquema de las etapas de recuperación de hidrocarburos.
(Diagrama adaptado de [2].)

Métodos térmicos: estos métodos consisten de manera general en la inyección de vapor, agua caliente o aire (*combustion in situ*), tienen el objetivo de transferir energía térmica al yacimiento y mejorar la movilidad del aceite.

Métodos de inyección de gases miscibles e inmiscibles: se inyectan gases como gas natural, nitrógeno o dióxido de carbono y tienen por objetivo incrementar el número capilar. Esto significa que la tensión interfacial entre el fluido inyectado y el aceite se reduce [3].

Métodos de inyección de químicos: involucran la inyección de polímeros para mejorar la efectividad de la inundación de agua y desplazar aceite. También, en esta categoría se clasifica la inyección de surfactantes para ayudar a tener una baja tensión superficial y así prevenir goteo del crudo a través del yacimiento, lo cual colabora a tener un mejor desplazamiento del mismo [2], [4]. Por último, dentro de esta clasificación se encuentra la inyección de álcalis, en este método se reduce la tensión interfacial entre el aceite y la fase acuosa debido a que el álcali reacciona con los radicales en el aceite formando surfactantes. La mojabilidad de la roca es alterada por la interacción entre el álcali y las moléculas polarizadas adsorbidas por los granos de la roca; la cual tiende a ser mojada por agua. Finalmente, se forma una emulsión agua/aceite. En el caso de aceites pesados y viscosos, estas emulsiones tienen menor viscosidad que el aceite, mejorando el flujo a través del yacimiento [4].

Otros métodos: involucran métodos híbridos, la inyección de agentes microbianos y variadas técnicas en fase experimental.

Está claro que existen diversos grupos y variadas técnicas para la recuperación mejorada de hidrocarburos. Sin embargo, sería un trabajo muy extenso analizar y modelar numéricamente cada una de ellas, por esta razón en este trabajo de investigación se analizan solamente el proceso isotermico de inyección de agua y el proceso térmico de inyección de vapor, el cuál también incluye la inyección de agua caliente. El motivo de seleccionar este proceso térmico es que el recobro de hidrocarburos es superior (en algunos casos hasta el 50% del aceite residual), y por lo tanto son los que tienen una mayor aplicación, además se tiene un amplio rango de utilización de acuerdo con el tipo de aceite y las propiedades del medio poroso [3]. Adicionalmente, existen en México algunos campos donde ya se está aplicando inyección de vapor como técnica de EOR, un ejemplo de ello es la prueba piloto llevada a cabo en el campo Samaria ubicado al oeste Villahermosa. También, se están llevando a cabo estudios experimentales para conocer la viabilidad de inyectar vapor en los campos maduros Ébano-Pánuco.

El modelado matemático y numérico de la inyección de fluido caliente y la inyección de aire para producir *combustion in situ* ha sido estudiado por varios autores [1] - [20]. En estos trabajos se detalla la importancia de los procesos térmicos de recuperación mejorada y se enfatiza en la mejora de las técnicas de simulación numérica, además algunos trabajos reportan la importancia de validar el código numérico con resultados experimentales [13], [21], [19]. La diferencia en el modelado numérico de ambas técnicas de recuperación mejorada, radica en que en los modelos matemáticos conceptuales para la inyección de fluido caliente sólo se consideran las fases participantes (aceite, líquido y gas) y en los modelos para la combustión *in situ* o inyección de químicos se consideran las fases y sus componentes, dando origen a los modelos composicionales [17], [22] es decir, el modelado numérico de los procesos térmicos de inyección de combustión *in situ* es más complejo que el de inyección por fluido caliente. Esto implica que para los modelos composicionales de combustión *in situ*, se tiene que desarrollar un modelo con ecuaciones de estado con el objetivo de conocer el comportamiento termodinámico de cada componente, es decir, se debe de acoplar un modelo *PVT* (Presión, Volumen, Temperatura) [17], [22]. Se debe mencionar que desarrollar un modelo numérico composicional, implica un trabajo muy extenso y debe ser llevado a cabo por un equipo multidisciplinario de trabajo. Por lo tanto, en esta tesis, se desarrolla un código numérico capaz de modelar los procesos térmicos de recuperación mejorada por inyección de fluido caliente, poniendo especial énfasis en la recuperación de hidrocarburos por inyección de vapor.

Las técnicas numéricas utilizadas para resolver los procesos térmicos de EOR que reportan los autores, son básicamente tres: diferencias finitas, volumen finito y elemento finito. Para la aplicación de estas técnicas, se necesita discretizar el espacio para resolver las ecuaciones diferenciales sobre cada punto o volumen discreto, según sea el caso. Principalmente se utilizan dos maneras de discretización del espacio físico: la discretización estructurada y la no estructurada. En la primera, los volúmenes elementales son formados por líneas o superficies coordenadas que tienen forma cuadrangular y se encuentran rodeados por la misma cantidad de elementos de volumen, con excepción de los volúmenes frontera. Además, la discretización estructurada permite implementar los tres métodos numéricos antes mencionados de manera sencilla, puesto que la matriz de coeficientes resultante queda ordenada y en la mayoría de

los casos es diagonalmente dominante, de tal manera que pueden implementarse algoritmos numéricos para su rápida solución. En la discretización no estructurada, el dominio sometido a análisis se cubre con elementos triangulares o polígonos irregulares. Esta discretización se utiliza para ajustar mejor las fronteras del dominio de estudio, y se pueden implementar versiones un poco más complejas de diferencias finitas (diferencias finitas generalizadas) y de volumen finito (CVFEM por ejemplo), aunque para este tipo de mallado la técnica más utilizada es el método de elemento finito debido a la naturaleza de la discretización que surge al aplicar dicha técnica.

Por otra parte, casi todos los trabajos reportados en la literatura utilizan sistemas coordenados ortogonales (sistemas coordenados Cartesiano y cilíndrico), como representación del dominio físico del problema a resolver. En la vida real, la mayoría de los yacimientos o formaciones no tienen forma de un sistema coordenado ortogonal, siendo esta un área de oportunidad para su estudio. Por este motivo, se utiliza la técnica de volumen finito para resolver el proceso de inyección de vapor en dominios no ortogonales, para así modelar y analizar el fenómeno de flujo de fluidos en medios porosos de estos procesos en dominios físicos con geometrías irregulares.

Otro punto importante es el aprovechamiento de las nuevas tecnologías para realizar cómputo paralelo, puesto que este permite resolver más variedad de problemas en menos tiempo, o problemas con decenas de millones de puntos discretos en tiempos relativamente cortos. Se han llevado a cabo muchas investigaciones para disminuir el tiempo de ejecución de los simuladores numéricos de yacimientos [23] - [26]. La mayoría de estos trabajos utilizan cómputo distribuido, por ejemplo, en el año 2015, Wang reportó un código numérico escalable para resolver el modelo matemático conocido como *black oil*, con este código el autor logró resolver un problema con aproximadamente 10 millones de celdas [27]. Sin embargo, aún cuando el código numérico sea escalable, existe la limitación de que entre más aceleración se requiera, se necesita ampliar el hardware al menos en la misma cantidad; por ejemplo para obtener una aceleración del doble, se requiere, al menos, duplicar el hardware.

1.2. Arquitectura Unificada de Dispositivos de Cómputo (CUDA)

Hace tan solo 30 años, el cómputo en paralelo parecía una profesión exótica y típicamente se atribuía exclusivamente dentro del campo de las ciencias computacionales. En la actualidad esta percepción ha cambiado profundamente. El mundo del cómputo ha pasado del punto de la exclusividad, a que hoy prácticamente todo aspirante a programador o estudiante de ingeniería necesita entrenamiento en programación de cómputo paralelo para llevar a cabo la implementación exitosa de los modelos matemáticos aplicados a problemas de ingeniería.

En los últimos 10 años, en la industria del desarrollo de software se han generalizado las

aplicaciones en cómputo paralelo, puesto que la mayoría de los consumidores en la actualidad tienen dispositivos con múltiples núcleos o *cores*. Hoy en día los dispositivos de gama baja cuentan con procesadores *dual core* y las estaciones de trabajo (*workstations*) tienen de 8 a 16 núcleos, entonces esto significa que el cómputo paralelo ya no es tan exótico o exclusivo de las supercomputadoras. Incluso, los teléfonos móviles inteligentes (*smart phones*) y otros dispositivos electrónicos incorporan ya cómputo paralelo para ofrecer a los usuarios más funcionalidad que sus predecesores [28].

El cómputo científico es en gran medida el mayor impulsor de esta rama de las ciencias computacionales, puesto que los problemas científicos requieren ejecutar operaciones aritméticas a cientos de miles o millones de datos mediante algoritmos iterativos. Para ejecutar estos cálculos se requiere mucha capacidad de cómputo (múltiples núcleos), y también una alta capacidad de acceder a los datos mientras se ejecutan las operaciones (memoria de acceso aleatorio y buses de comunicación) y mucha capacidad de almacenamiento para guardar los datos obtenidos mediante los cálculos realizados (disco duro). Algunos de los problemas que requieren gran capacidad de cómputo son: simulación de partículas, simulación del flujo de fluidos, simulación del flujo de fluidos en yacimientos petroleros, simulación de fenómenos estelares, entre otros. La supercomputadora más poderosa del mundo actualmente es *Summit* que cuenta con 2,282,544 núcleos y se encuentra en el Laboratorio Nacional Oak Ridge en Tennessee, Estados Unidos. La supercomputadora Summit esta constituida por 4,356 nodos cada uno equipado con un arreglo dual de 22 procesadores Power9 de IBM y 6 unidades de procesamiento gráfico Tesla V100. Esta supercomputadora se utiliza por el Departamento de Energía de Estados Unidos para ejecutar cálculos de gran capacidad que colaboren a la seguridad nacional, así como a los desarrollos científicos en los campos de ciencias de materiales, medicina nuclear, ciencias químicas, exploración de petróleo, el cosmos, ciencias biológicas, entre otros.

Por 30 años uno de los métodos más importantes para mejorar el desempeño de los dispositivos de cómputo ha sido incrementar la velocidad de operación del reloj del procesador. Empezando con las primeras computadoras personales a principios de los 80's, las cuales tenían una frecuencia de reloj de alrededor de 1 MHz en su unidad central de procesamiento (*Central Processing Unit, CPU*). Después de 30 años, la mayoría de los equipos de cómputo tiene frecuencias de reloj entre 1 GHz hasta 4 GHz , prácticamente, 1,000 veces más rápidos que sus predecesoras. Aunque el incremento de la frecuencia de reloj no es el único método por el cual se puede incrementar el desempeño de una computadora, siempre se ha considerado que ésta, es una fuente confiable para medir el desempeño. Sin embargo, en la actualidad las industrias manufactureras están buscando alternativas para incrementar la potencia computacional de sus dispositivos, debido a que existen limitaciones físicas en la fabricación de los circuitos integrados, la frecuencia del reloj y las restricciones del consumo de energía y generación de calor. Por este motivo, habría que mirar hacia otros horizontes para aumentar el rendimiento de las computadoras personales [28].

Por otra parte, los procesadores gráficos han experimentado una revolución. Para princi-

pios de la década de los 90's creció la popularidad de los sistemas operativos para integrar controles gráficos, esto ayudó a crear un mercado para un nuevo tipo de procesador. Entonces, en 1990 los usuarios comenzaron a comprar aceleradores gráficos bidimensionales para sus computadoras personales. Estos aceleradores gráficos contaban con asistencia de hardware que realizaba operaciones de mapeo de bits para asistir al ambiente gráfico de la pantalla de los sistemas operativos.

Al comienzo del año 2,000 las GPU's (*Graphics Processing Unit, GPU*) fueron diseñadas para producir el color para cada pixel en la pantalla utilizando unidades aritméticas conocidas como sombreadores de pixel (*pixel shaders*). En general, un sombreador de pixel utiliza su posición (x, y) en la pantalla así como información adicional, después combina ambas entradas y calcula el color final. La información adicional pueden ser los colores de entrada, la textura, u otros atributos que tienen que pasarse al matizador cuando éste se muestra en la pantalla. Los investigadores observaron que los colores de entrada pueden ser “cualquier dato”, puesto que el programador controla completamente la aritmética y las texturas de los colores. Entonces, si los datos de entrada fueran datos numéricos que signifiquen algo más que otros colores, los programadores podrían programar los sombreadores de pixel para ejecutar cualquier otro tipo de cálculos con estos datos. Los resultados serían devueltos a la GPU como el “color final del pixel”, aunque los colores simplemente serían el resultado del listado de instrucciones que el programador ha hecho para ejecutarse con los datos de entrada. En esencia, la GPU esta siendo “engañada”, para ejecutar tareas de procesamiento no gráfico haciendo que estas tareas se parezcan a la tarea estándar del procesamiento gráfico [28].

Las GPU's tendrían un futuro prometedor, ya que mostraron un alto rendimiento en los experimentos hechos con los sombreadores de pixel. Sin embargo, el modelo de programación todavía estaba lejos de los desarrolladores en masa. Había restricciones de recursos, debido a que los programadores podrían recibir solo algunos datos de entrada útiles de los colores y de las texturas. También, existían serias limitaciones sobre cómo y cuándo el programador podía escribir resultados a la memoria, entonces algunos algoritmos requerían la habilidad de escribir de manera arbitraria en la memoria que no podría ejecutarse sobre una GPU. Además, era casi imposible predecir cómo una GPU en particular trataba con datos de punto flotante, entonces la mayoría de los cálculos científicos serían incapaces de utilizar una GPU. Finalmente, cuando el programa inevitablemente calculaba datos incorrectos, fallaba en la ejecución, o simplemente trababa la máquina, no había un método razonablemente bueno para corregir los códigos que eran ejecutados en la GPU.

Como si las limitaciones no fueran suficientes, cualquiera que quisiera utilizar una GPU para ejecutar cálculos de propósito general necesitaría aprender *OpenGL* o *DirectX*, debido a que estos softwares mantenían el único enlace para interactuar con una GPU. Esto no sólo significaba almacenar los datos en texturas gráficas y ejecutar cálculos llamando funciones de OpenGL o DirectX, también significaba escribir cálculos en lenguajes de programación especiales para gráficos conocidos como lenguajes de sombreado (*shading languages*). No sería hasta unos años después en las versiones de la tarjetas NVIDIA-GeForce 3, que el cómputo

para las GPU's estaría listo por primera vez mediante un nuevo lenguaje de programación llamado *CUDA*.

La Arquitectura Unificada de Dispositivos de Cómputo (*Compute Unified Device Architecture*, *CUDA*), es una arquitectura de cálculo paralelo desarrollada por *NVIDIA*, que aprovecha la potencia de la GPU para proporcionar un incremento en el rendimiento del sistema y proporciona un lenguaje estándar para el programador, sin tener que aprender el lenguaje de los ambientes gráficos como *OpenGL*. La plataforma de cálculo paralelo *CUDA* proporciona extensiones de *C* y *C++* que permiten implementar el paralelismo en el procesamiento de tareas y datos. El programador puede expresar ese paralelismo mediante funciones *kernel* con diferentes lenguajes de alto nivel como *C*, *C++* y *Fortran*, o mediante estándares abiertos como las directivas de *OpenACC*. Un ejemplo de cómo se usa *CUDA* para cómputo en paralelo se describe en el apéndice C.

Siendo *CUDA* un lenguaje relativamente nuevo, puesto que surgió en el año 2007, existen muchas investigaciones en desarrollo para aprovechar esta tecnología. En el área de los modelos numéricos para petróleo se han desarrollado varios trabajos desde paralelizar la forma de solución de los sistemas de ecuaciones lineales [29], [30], [31], hasta el desarrollo de bibliotecas para realizar cómputo paralelo masivo en clusters híbridos (CPU-GPU) [32], [33].

1.3. Objetivos de este trabajo

Haciendo una síntesis de lo mencionado anteriormente se tienen los siguientes objetivos:

Objetivo general:

Desarrollar un software que permita predecir el comportamiento físico del flujo de fluidos en medios porosos, para analizar los fenómenos que ocurren en los procesos térmicos de inyección de vapor, el cual es una de las principales técnicas de recuperación mejorada de hidrocarburos por inyección de fluido caliente.

Objetivos específicos:

1. Desarrollar un código numérico computacional capaz de modelar el flujo de fluidos trifásico térmico en medios porosos, el cual se aplique en dominios ortogonales y no ortogonales.
2. Paralelizar el código numérico computacional utilizando unidades de procesamiento gráfico (GPU's).
3. Comparar el código numérico desarrollado con problemas de referencia y datos petrofísicos reportados en la literatura.

1.4. Aportaciones de este trabajo

Hasta el día de hoy, como se explicará en el siguiente capítulo, existen muchas investigaciones encaminadas a mejorar la simulación numérica de la recuperación de aceite en yacimientos petroleros mediante la inyección de vapor. Prácticamente todos los modelos que se han revisado parten de las ecuaciones de conservación de masa y energía, los cuales han evolucionado en la forma de calcular las propiedades de flujo, términos fuente (modelado de pozos) y modificaciones en los algoritmos de solución.

Sin embargo, como el modelo de inyección de vapor tiene un alto grado de complejidad en su implementación numérica, primero se estudia el modelo “bien conocido” de flujo bifásico incompresible, el cual se utiliza para simular el proceso isotérmico de inyección de agua. El estudio de este modelo tiene como objetivo llevar a cabo las adaptaciones necesarias en el código numérico para su utilización en dominios no ortogonales y aprovechar la potencia del cómputo paralelo mediante el uso de GPU’s, para posteriormente utilizar estas adaptaciones en la implementación numérica del simulador de inyección de vapor. Además, como se verá en los resultados, se obtuvieron valores numéricos muy precisos que permiten ser usados como datos de comparación para otros autores.

En el caso particular del modelo numérico para simular el proceso de inyección de vapor, se seleccionó el modelo trifásico térmico propuesto por Shutler en el año de 1970 [1]. Aún cuando el modelo seleccionado es antiguo, actualmente es uno de los más utilizados como modelo de referencia (véase referencias [7], [10], [34]). En este trabajo se proponen varias modificaciones al algoritmo original, una de estas modificaciones consiste en el uso de una discretización temporal completamente implícita para darle mayor estabilidad, ya que el autor original utiliza una discretización semi-implícita; adicionalmente, en este trabajo se seleccionaron como variables primarias la presión de la fase aceite p_o y las saturaciones de las fases agua y vapor S_w y S_g , respectivamente, mientras que Shutler seleccionó la presión de la fase aceite p_o y las presiones capilares de los sistemas agua-aceite p_{cow} y aceite-gas p_{cog} ; y finalmente, para hacer la simulación en dominios irregulares, se utilizan coordenadas curvilíneas para transformar el plano físico al plano computacional. Todo ello con la meta de obtener mayor generalidad y que los cálculos puedan adaptarse a las complejidades de los yacimientos reales. También se estudia la aplicación del cómputo paralelo utilizando GPU’s para este modelo numérico.

Los resultados obtenidos en este trabajo, demuestran que estas adaptaciones pueden ser de gran utilidad para la predicción del comportamiento físico de campos petroleros cuando se desplaza hidrocarburo mediante la inyección de agua o vapor, reduciendo los tiempos de cálculo, y mejorando en algunos casos la precisión de los resultados.

1.5. Estructura de la tesis

Este documento está ordenado de la siguiente manera: en el Capítulo 2, se presenta la definición y descripción de los procesos térmicos para la recuperación mejorada de hidrocarburos. También, en ese capítulo se detalla el estado del arte en el que se encuentran los modelos numéricos y computacionales para la simulación de estos procesos de EOR. En el Capítulo 3, se utiliza la formulación axiomática para deducir las ecuaciones de balance de masa y energía de los sistemas multifásicos en medios porosos. En el contenido del Capítulo 4, se tratan los aspectos matemáticos relacionados con los sistemas de coordenadas curvilíneas y su relación con el sistema coordenado Cartesiano, para su aplicación en la transformación de las ecuaciones de balance de masa y energía. En los Capítulos 5 y 6, se presentan las técnicas numéricas y los algoritmos computacionales con los cuales se resuelven los modelos matemáticos para llevar a cabo la simulación del proceso isotérmico de inyección de agua y el proceso térmico de inyección de vapor, para simular la recuperación de hidrocarburos. Por último, en el Capítulo 7, se presentan las conclusiones de este trabajo, también se describen brevemente los puntos de mejora y áreas de oportunidad para continuar con este trabajo de investigación. Además de los capítulos mencionados, se incluyen tres apéndices: el Apéndice A describe la discretización de las ecuaciones de balance en el dominio computacional; el Apéndice B contiene información sobre las ecuaciones de estado y correlaciones empíricas utilizadas para el desarrollo del modelo numérico de inyección de vapor, y el Apéndice C, en donde se explica brevemente como se hace uso de CUDA para realizar cómputo paralelo.

Capítulo 2

Procesos térmicos para la Recuperación Mejorada de Hidrocarburos

En este capítulo, se da una breve descripción de los procesos térmicos para la recuperación mejorada de hidrocarburos. Además, se aborda el estado del arte para el modelado numérico de estos.

2.1. Descripción de los procesos térmicos para la recuperación mejorada de hidrocarburos

La diferencia entre los métodos térmicos, con respecto a otros métodos de EOR, recae en el hecho de que el fluido inyectado proporciona energía térmica al yacimiento. Existen dos categorías de métodos térmicos: aquellos en los que el calor se produce en la superficie (inyección de fluido caliente) y aquellos en los cuales el calor se crea en la formación del yacimiento (combustión *in situ*). En el primer caso, el fluido inyectado provee calor al yacimiento, mientras que en el segundo caso, el fluido es uno de los reactantes involucrados en una reacción exotérmica que ocurre dentro del yacimiento. Hay muchas diferencias básicas entre estos dos métodos, debido a que el calor aportado tiende a dispersarse lejos de la zona calentada, entonces, es evidente que la pérdida de calor es mucho más grande en el primer caso que en el segundo [4].

En la inyección de fluido caliente, los fluidos inyectados en su máxima temperatura, inicialmente entran en contacto con la zona de barrido y a medida que avanzan por el yacimiento existe una pérdida de calor. En la combustión *in situ*, el calor sólo se produce donde éste se requiere, por ejemplo, donde el aceite será desplazado. De esta manera, la aplicación de la inyección del fluido caliente depende esencialmente de su eficiencia térmica. Esto depende de dos factores: de la pérdida de calor en el punto de inyección hacia las formaciones de los

alrededores; y de las características del yacimiento: capas y rocas base, por las cuales también fluye el calor. Haciendo una comparación, la pérdida de calor no es un factor limitante en la combustión *in situ*.

La eficiencia térmica en varios de los métodos podría mejorar, si se recupera alguna cantidad de calor atrapado en la formación o en las rocas de los alrededores, por ejemplo, el proceso de combustión mediante la inyección simultánea de aire y agua. Otras posibilidades, incluyen la aplicación de diferentes métodos de manera alternada, por ejemplo, inyección de agua fría después de un barrido parcial de vapor o combustión *in situ*.

El factor común en todos los métodos térmicos, es el aumento en la temperatura de una parte del yacimiento. Esto involucra mecanismos específicos, los cuales ayudan a mejorar la eficiencia del barrido y del desplazamiento, lo que a su vez ayuda a incrementar la tasa de producción de hidrocarburos. Por lo tanto, para el modelado numérico de los procesos térmicos, se deben de considerar la transferencia de calor hacia la roca o formación y las propiedades de los fluidos que dependen de la temperatura para conocer el comportamiento dinámico de los fluidos.

2.1.1. Procesos térmicos por inyección de fluido caliente

Inyección de agua caliente

Para entender este proceso de recuperación considere la figura 2.1 , la cual muestra los perfiles de saturación de agua y temperatura en un yacimiento homogéneo bajo condiciones de desplazamiento unidimensional, en éste se inyecta agua caliente por el extremo izquierdo y se producen los fluidos por el extremo derecho, despreciando la pérdida de calor en los alrededores.

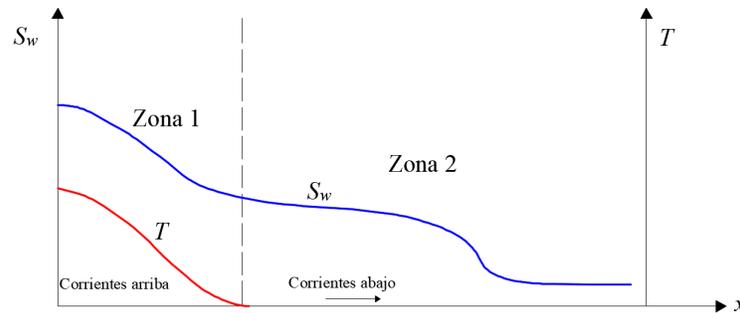


Figura 2.1: Perfiles de saturación y temperatura en el proceso de inyección de agua caliente. (Ilustración adaptada de [4].)

En la figura 2.1, se pueden distinguir dos zonas principales: corrientes arriba y corrientes abajo, las cuales se describen como sigue:

Zona 1: Siguiendo la trayectoria corrientes arriba, la saturación de agua y temperatura van en aumento, mientras que la saturación de aceite residual disminuye, lo que indica que el barrido de aceite hecho por el agua es más eficiente cuando la temperatura aumenta. Esto puede observarse de mejor manera en la figura 2.2. Cabe señalar que si el aceite contiene fracciones volátiles de hidrocarburo, ciertas fracciones pueden ser desplazadas por vaporización-condensación.

Zona 2: La zona de corrientes abajo, indica que el aceite en el lugar es desplazado por agua a la misma temperatura del yacimiento, la cual se enfría por pérdida de calor.

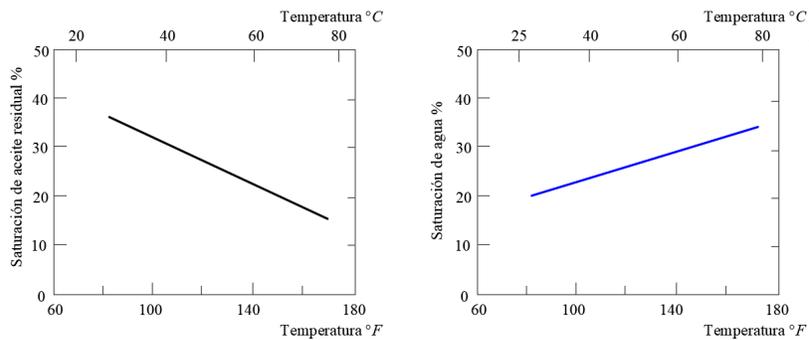


Figura 2.2: Saturaciones de aceite residual y agua versus la temperatura (Gráficos adaptados de [4].)

En la práctica, la pérdida de calor de la zona caliente hacia la formación resulta en una disipación de energía más pronunciada y con ello la disminución de la temperatura en la dirección del flujo, este fenómeno se puede observar en la figura 2.3. En esta figura se aprecian 2 perfiles de temperatura, el perfil superior corresponde a la curva de temperatura en el cual se ignoran las pérdidas térmicas, ello se refiere a que no se considera la disipación de energía en las fronteras del medio poroso, mientras que en el perfil inferior, en el cual se incluyen las pérdidas térmicas, la curva de temperatura declina más rápidamente.

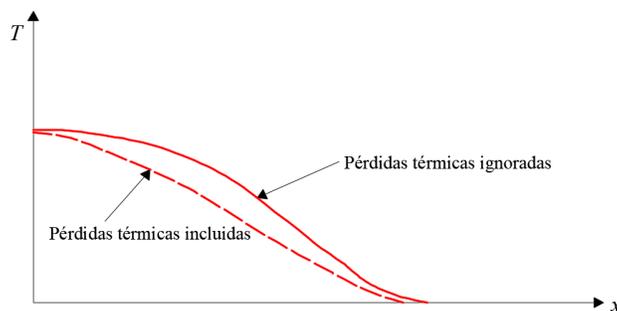


Figura 2.3: Perfil de temperatura en el desplazamiento por inyección de agua caliente. (Ilustración adaptada de [4].)

Inyección de vapor

Este proceso, consiste en la inyección continua de vapor al yacimiento, el cual forma una zona de vapor que avanza lentamente hacia las zonas productoras. En otras palabras, se trata de un desplazamiento de aceite por vapor y agua condensada, bajo condiciones inmiscibles. El aceite es movilizado principalmente por la reducción de su viscosidad, que resulta de la elevación de la temperatura del yacimiento. La saturación de aceite en las zonas barridas por el vapor puede reducirse aproximadamente hasta en un 10 por ciento, resultando en factores de recuperación que pueden estar en el rango de 50 a 60 por ciento del volumen original [3].

Con la misma hipótesis del yacimiento homogéneo y unidimensional, se pueden identificar tres zonas principales, las cuales se muestran en la figura 2.4.

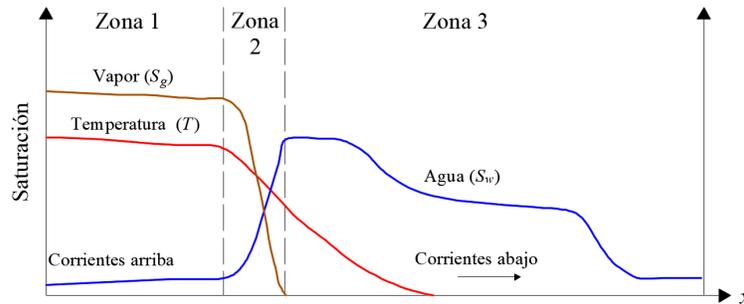


Figura 2.4: Perfil de saturación en el desplazamiento por vapor.
(Ilustración adaptada de [4].)

Zona 1: Es la zona de condensación corrientes arriba, donde la temperatura es alta y cae suavemente de acuerdo a la temperatura de condensación del vapor. En ésta, existen las tres fases de los fluidos (vapor, agua y aceite) que se mantienen aproximadamente constantes, de las tres fases sólo se mueven el vapor y el agua. También, pueden vaporizarse las fracciones más ligeras del hidrocarburo, las cuales son movilizadas por la fase vapor. La temperatura de la formación es prácticamente igual a la temperatura de inyección del vapor.

Zona 2: El vapor entra en contacto con la formación fría y se condensa. La temperatura promedio disminuye durante la condensación, y las fracciones de hidrocarburo previamente vaporizadas se condensan al mismo tiempo que el vapor.

Zona 3: En esta zona el desplazamiento es por agua caliente, eso significa que la saturación de vapor S_g disminuye a cero. Sin embargo, debido a que el volumen específico del vapor presente en las zonas previas es mucho mayor que el del agua, la velocidad del agua es más alta que si el agua líquida fuera inyectada a la misma cantidad de flujo másico, mejorando la tasa de recuperación de hidrocarburo.

La importancia relativa de la vaporización y condensación (destilación) de las componentes más volátiles del aceite, depende precisamente del tipo de aceite y la composición química de sus componentes. Puede estimarse que el incremento debido al fenómeno de destilación, puede alcanzar el 20 % para un crudo del cual se destila por lo menos el 50 % a 170°C. De esta manera, en el caso de los aceites ligeros, la destilación del crudo es la mayor contribución a la eficiencia del desplazamiento por fluido caliente, para crudos pesados, pasa lo opuesto. Si se toma en cuenta la pérdida de calor en los alrededores de la formación, se encuentra que el vapor se condensa continuamente corrientes arriba, es decir, la zona 1 se reduce y la zona de condensación (zona 2) comienza a corta distancia del pozo inyector, la cual aumenta de tamaño con el tiempo.

Inyección cíclica de vapor

Es un método de recuperación que se realiza en un pozo que opera alternadamente como inyector y productor. Este proceso considera ciclos que básicamente consisten en tres etapas. En la etapa inicial, el vapor se inyecta continuamente por un tiempo. Posteriormente, el pozo es cerrado para que el calor inyectado se transfiera y se distribuya en la vecindad del pozo. Finalmente, el pozo es abierto a producción, donde inicialmente se observa un incremento de producción, por un periodo corto y posteriormente declina lentamente durante un periodo que puede ir desde varios meses hasta un año aproximadamente [3].

Drenado gravitacional asistido por inyección de vapor

El drenado gravitacional asistido por inyección de vapor SAGD (*Steam Assisted Gravity Drainage*), es un proceso térmico de recuperación mejorada desarrollado a principios de los 80's para los yacimientos de areniscas Canadienses [20]. El proceso considera un par de pozos perforados horizontalmente uno sobre el otro, separados por un espacio vertical teniendo en cuenta el espesor de la capa productora del yacimiento. El vapor se inyecta en el pozo superior, llamado pozo inyector, y el pozo inferior (pozo productor) recolecta el aceite.

El proceso de SAGD, se utiliza comúnmente cuando el yacimiento de hidrocarburo conformado de arenisca es profundo y tiene bajas probabilidades de ser explotado de manera convencional. La configuración del proceso de SAGD se muestra en la figura 2.5. El concepto fundamental de este proceso, es que el vapor inyectado transfiere calor dentro del yacimiento y el vapor se condensa en su fase líquida. La transferencia de calor movilizará el aceite encurado en la formación arenosa por disminución de la viscosidad, y después éste se drenará por gravedad al pozo productor localizado en la parte inferior. Este proceso ocurre generalmente en una región alrededor del pozo inyector influenciado por la temperatura de inyección de vapor. La región afectada se le conoce como cámara de vapor (*steam chamber*), como el aceite drenado es producido, la cámara de vapor se expande hacia arriba y hacia los lados (véase figura 2.5).

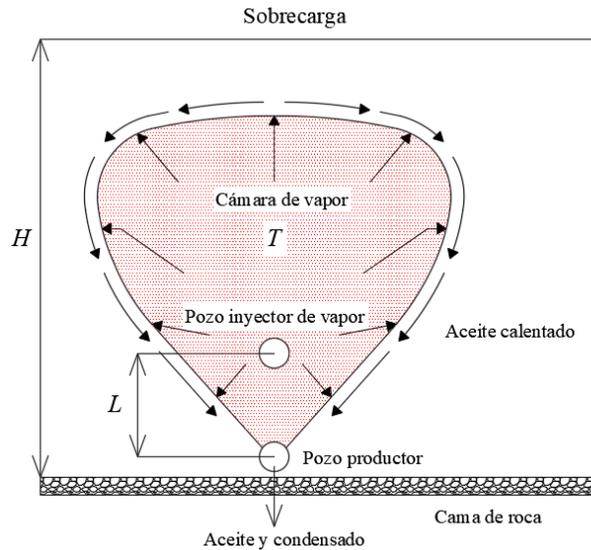


Figura 2.5: Proceso de drenado gravitacional asistido por inyección de vapor.
(Ilustración adaptada de [20].)

2.1.2. Combustión *in situ*

El principio básico de la combustión *in situ* es lograr la combustión dentro de los poros de un yacimiento productor de hidrocarburos, quemar una parte del aceite en el lugar con objeto de mejorar el flujo y desplazamiento de la parte de aceite no quemada. La combustión se logra con la inyección de aire dentro del yacimiento en uno o más pozos. El calor que se genera durante la combustión es suficiente para elevar la temperatura de la roca, el cual permite la propagación de un frente de combustión después de la ignición inicial.

Si se considera el caso más simple, es decir, la inyección de aire en un pozo y otro pozo para la producción del hidrocarburo, puede apreciarse la dirección de propagación del frente de combustión. Si en los alrededores de la zona del pozo inyector se logra una temperatura suficientemente alta, la ignición tiene lugar en la vecindad de pozo inyector y el frente de combustión viaja en dirección del pozo productor, en la dirección del flujo de fluidos. Esto se conoce como combustión hacia adelante (*forward combustion*). Por otra parte, si se calientan los alrededores del pozo productor, la ignición tiene lugar cerca de este pozo y el frente de combustión viaja hacia el pozo inyector, en dirección opuesta del flujo de fluidos. A esto se le llama combustión en reversa (*reverse combustion*).

Debido a que el proceso térmico de combustión no se aborda en este trabajo, sólo se hace mención del mismo. Si el lector desea profundizar en este tema puede leer las referencias [2] y [4].

2.2. Estado del arte de los modelos numéricos para simular los procesos térmicos de inyección de vapor

En esta sección se presentan de manera cronológica y general, los avances en cuestiones numéricas para simular procesos térmicos de inyección de vapor para la recuperación mejorada de hidrocarburos.

A principios de la década de los 70's, Shutler presentó un modelo trifásico térmico unidimensional para simular numéricamente el proceso de inyección de vapor [1]. En el algoritmo reportado, se resuelven secuencialmente las ecuaciones de balance de masa y energía por el método de Newton-Raphson. Además, se proporciona una expresión matemática para el término de transferencia de masa en las fases vapor y agua, al cual se le denominó, término de condensación- evaporación. El autor utilizó diferencias finitas considerando una discretización semi-implícita en el tiempo. Shutler validó su modelo numérico con los resultados reportados experimentalmente por Willman [35]. Poco después, el modelo fue extendido por el mismo autor para considerar dos dimensiones [5]. Posteriormente, Abdalla y Coats desarrollaron un modelo numérico bidimensional para simular el proceso de inyección de vapor mediante la técnica de diferencias finitas, en el cual utilizaron una discretización implícita en presión y una discretización explícita en saturación (IMPES) [6]. Los autores también presentaron una solución analítica para considerar la pérdida de calor por las fronteras superior e inferior del yacimiento. El modelo fue validado experimentalmente por los autores, ellos reportaron una concordancia superior al 90 % entre la simulación numérica versus los datos experimentales. A mediados de la década de los 70's Coats *et al.*, desarrollaron un modelo tridimensional para la simulación numérica del proceso de inyección de vapor. El modelo describe el flujo trifásico de las fases agua, aceite y vapor mediante las ecuaciones de balance de masa, la transferencia de energía al yacimiento se incorpora mediante la ecuación de balance de energía y una ecuación de equilibrio térmico [7]. La discretización de estas ecuaciones se lleva a cabo mediante diferencias finitas. En este método se resuelve simultáneamente los balances de masa y energía, las cuales se acoplan mediante una ecuación de equilibrio térmico, eliminando así la necesidad de calcular el término de transferencia masa entre agua y vapor (término de condensación- evaporación). En 1979 Grabowski *et al.*, presentaron un modelo numérico utilizando una discretización en el tiempo completamente implícita para simular la combustión *in situ* o procesos de inyección de vapor [8]. El modelo incorpora cuatro fases, un número variable de componentes para el aceite, un número variable de reacciones químicas y toma en cuenta los términos de gravedad y presión capilar. Los autores estudiaron la sensibilidad de su modelo a los cambios en el número de celdas de malla y el tamaño del paso de tiempo. Por último, se concluyó que su modelo era estable numéricamente, sin embargo que requería mucho tiempo de cómputo debido a la discretización completamente implícita. A mediados de los 80's Rubin y Buchanan, publicaron un modelo numérico completamente implícito a cuatro fases para simular la inyección de agua caliente y de vapor en un yacimiento, así como la combustión seca y la combustión húmeda, considerando las tres dimensiones en un sistema coordenado cartesiano o curvilíneo [9]. Ellos acoplaron un modelo implícito

de pozo a su simulador, donde la variable primaria es la presión del pozo. Para validar su modelo, ellos compararon los resultados obtenidos con experimentos reportados previamente en la literatura. De este trabajo, los autores consideraron que su simulador daba buenos resultados para el problema de extracción de aceite muerto, así como para procesos térmicos de recuperación. Para el año de 1986 la SPE (*Society of Petroleum Engineers*) decidió convocar a distintas empresas internacionales, esto con la finalidad de que probaran sus simuladores de inyección de vapor con varios problemas propuestos a escala de campo; puesto que hasta la fecha no había ningún tipo de benchmark de esta índole, aun cuando ya se habían hecho experimentos a escala de laboratorio. Los problemas propuestos fueron: inyección cíclica de vapor, el problema simétrico de los 9 pozos, y el problema simétrico de los 9 pozos considerando componentes volátiles y no volátiles en el aceite. Algunas de las empresas participantes fueron: ARCO, Chevron, SSI y CMG. En 1987 la SPE publicó los resultados, en los cuáles los distintos simuladores proporcionaban diferencias entre sí de hasta el 25 % en los valores de producción, corte de vapor, y otras variables para los mismos problemas, sin embargo la forma cualitativa se mantenía en todos los casos [36]. Los miembros de la SPE concluyeron que las diferencias en los resultados se debían a las diferencias entre los algoritmos numéricos utilizados, ya que los parámetros de convergencia e iteraciones fueron determinados empíricamente por cada participante. Además, las ecuaciones de balance de masa y energía, los valores petrofísicos y el tamaño de malla fueron iguales para todos los simuladores por lo que no había diferencias en el modelo matemático principal.

Para la década de los 90's, los esfuerzos en el desarrollo numérico para simular el proceso de inyección de vapor se concentraron en validar los modelos matemáticos y numéricos, teniendo como objetivo su aplicación a yacimientos reales o generar nuevas técnicas de EOR, como es el caso de la técnica de drenado gravitacional asistido por inyección de vapor (SAGD). En 1992 Chan *et al.*, se percataron que los simuladores de yacimientos térmicos convencionales carecían del acople de los fenómenos de transporte que ocurren en las fracturas por la inyección cíclica de vapor, específicamente en los yacimientos de aceite localizados en Alberta, Canadá. Los autores plantearon un procedimiento por el cual las características de fractura pueden ser, al menos parcialmente, incorporadas en dichos estudios de simulación, ya sea en simuladores comerciales o simuladores de desarrollo propio [37]. Al utilizar un modelo de fractura, las características de la fractura se trasladan a las distribuciones de presión, temperatura y saturación, que el simulador de yacimiento térmico puede utilizar para realizar la simulación. El procedimiento se probó con éxito en varios pozos simulados con inyección cíclica de vapor en la formación McMurray del yacimiento Athabasca en Alberta, Canadá. Los autores concluyeron que su metodología podía ser aplicada para otro tipo de formaciones ajustando los parámetros de la fracturas. En el año de 1995 Kisman *et al.*, realizaron un estudio numérico para predecir el rendimiento de recuperación del proceso de SAGD en un yacimiento de Burnt Lake, Canadá. Se preparó una descripción geológica y petrofísica del yacimiento y un modelo numérico, el cual se basó en los datos recopilados de una prueba piloto de SAGD que realizó el Departamento de Energía de Pruebas Subterráneas de Alberta (UTF, por sus siglas en inglés). En el caso base se calculó un desempeño favorable para un par de pozos horizontales de 1000 m de longitud, con un volumen de recuperación de aceite de 70.1

m^3/d por más de 15 años. Los autores se dieron cuenta que para optimizar las operaciones del proceso SAGD, influyen factores como el espaciado entre los pares de pozos y el gasto de inyección, los cuales pueden mejorar el rendimiento previsto. En 1996 Nasr *et al.*, llevaron a cabo experimentos bidimensionales para representar el proceso de SAGD, los cuales estaban diseñados para representar depósitos de aceite pesado y extra pesado, para calibrar el simulador numérico de yacimientos STARS [38]. Se realizaron observaciones visuales durante el desarrollo experimental en la cámara de vapor (celda de gravedad) las cuales fueron comparadas con las predicciones del modelo numérico. Los autores realizaron un estudio detallado sobre el efecto de la permeabilidad en la simulación fuente/sumidero del proceso SAGD, con el cual se demostró la importancia de la fase de inicialización o transitoria del proceso. La duración de la fase de inicialización aumentó a medida que disminuía la permeabilidad, lo que indica que esta fase ocupaba una parte cada vez más importante de todo el proceso de drenado a medida que disminuía la permeabilidad. Con el simulador calibrado experimentalmente, los autores reportaron los parámetros de operación del SAGD para la recuperación de aceite pesado en yacimientos de areniscas. En ese mismo año Sumnu *et al.* reportaron el ensamble de un equipo experimental, el cual fue diseñado con el objetivo de comprender los mecanismos involucrados en la transferencia de fluidos matriz-fractura [39], esto debido a la incertidumbre que existía en los procesos de transferencia de fluidos matriz-fractura durante la inyección de vapor en yacimientos fracturados. Ellos llevaron a cabo las inyecciones continua y cíclica de vapor en su desarrollo experimental para calibrar su simulador numérico. Las saturaciones se determinaron *in situ* tanto en la fractura como en la matriz mediante un escaner de rayos x . Los resultados indicaron que no había saturación de vapor en la matriz, y que la conducción de calor es el mecanismo dominante de transferencia de calor. Se usaron simulaciones numéricas para modelar experimentos de inyección de vapor continuo y cíclico. Para modelar las pérdidas de calor, los modelos numéricos en el simulador tuvieron que ajustarse basándose en el análisis de las pérdidas de calor observadas experimentalmente con modelos analíticos. Después de este ajuste, los resultados de las simulaciones coincidieron bien con los experimentos. Los autores encontraron que la recuperación de petróleo fue causada principalmente por la imbibición de agua en la matriz y la conducción de calor. Ellos concluyeron que sus resultados pueden ser útiles para modelar la transferencia de fluidos matriz-fractura en modelos térmicos de doble porosidad.

En los últimos 17 años, se han hecho modificaciones a los algoritmos principales de solución para mejorar su desempeño o aplicarlos a problemas específicos, tales como: la aparición de caminos de flujo preferenciales, acoplamiento de efectos geomecánicos, técnicas de recuperación híbridas, entre otros. En el 2001 Sasaki *et al.*, publicaron un estudio experimental y numérico para el proceso de SAGD [13]. Uno de los objetivos de la investigación numérica fue determinar la relación entre los resultados numéricos y los datos generados mediante experimentos escalados. Se utilizó el simulador térmico desarrollado por *Computer Modelling Group's* (CMG, STARS) para analizar la cámara de vapor y su distribución de temperatura. Adicionalmente, se estudiaron los efectos en el desempeño del proceso de SAGD para diferentes parámetros físicos, como son: la presión de inyección de vapor, la separación vertical entre el pozo productor y el pozo inyector, y espesor del yacimiento. Los resultados de la simu-

lación numérica concordaron con aquellos obtenidos por los experimentos para el caudal de aceite producido y la producción acumulativa. Los autores concluyeron que los resultados de la simulación utilizando funciones de permeabilidad relativa lineal con puntos de saturación final no cero, concuerdan con el perfil de la cámara de vapor y los resultados experimentales observados. En el año 2003, Sheorey y Muralidhar reportaron un estudio numérico del flujo inmiscible de aceite pesado en un medio poroso [14]. La región física tomada en cuenta es un dominio cuadrado en un plano horizontal con puntos de baja y alta presión en esquinas opuestas. El agua es el fluido de invasión, cuando se introduce en alta presión desplaza el aceite a una zona de baja presión. El objetivo de la investigación fue estudiar la extensión del desplazamiento de aceite por agua a través del medio poroso en determinada cantidad de tiempo y la aparición de caminos preferenciales de flujo (dedos). Se emplea un modelo matemático que puede tratar las presiones de flujo individualmente, efectos capilares y la transferencia de calor. Se desarrolló un modelo completamente implícito bidimensional para calcular la presión y los campos de temperatura. Se adopta la técnica de descomposición del dominio en la solución numérica debido a que el problema es de gran escala y requiere grandes recursos de cómputo. Los autores reportaron que cuando el cociente de viscosidad del aceite y agua es grande, las fuerzas viscosas del aceite se vuelven dominantes y constituyen un factor mayor para el control de las distorsiones de flujo en un medio poroso. Los cálculos numéricos muestran que los efectos no isotérmicos así como las capas heterogéneas promueven la formación de dedos viscosos y consecuentemente se afecta la eficiencia del barrido por inyección de agua a alta presión.

A principios de la década actual en el año 2010, Yin *et al.*, se percataron que la simulación tradicional de yacimientos falla en predecir los transitorios asociados con los movimientos de la superficie, porque generalmente no se acoplan los efectos geomecánicos [18]. También que en la recuperación mejorada de hidrocarburos, específicamente el proceso de inyección de vapor, se involucran esfuerzos considerables y presiones muy altas, además de cambios de volumen y temperatura. Ellos presentaron un modelo geomecánico-térmico acoplado utilizando una técnica híbrida conocida como DDFEM, en la cual se utiliza la técnica de elemento finito para la solución del flujo de fluidos en el yacimiento considerando los efectos térmicos en los alrededores de la zona afectada, y el método de discontinuidad de desplazamiento (Displacement Discontinuity, DD) se utiliza para resolver la zona elástica e isotérmica. Los autores concluyeron que su modelo considera los movimientos transitorios de la superficie del suelo de una manera natural y concuerda con los resultados reportados previamente en la literatura. En el año 2013, Mozaffari y Nikookarb presentaron un modelo numérico tridimensional a tres fases de la inyección de vapor en yacimientos de aceite pesado [34]. El modelo incluye los efectos de permeabilidad relativa de las tres fases, presión capilar y las propiedades de los fluidos dependientes de la presión y temperatura. En este modelo se permite la transferencia de masa en la interfaz agua-vapor, pero se considera que el aceite es no volátil y el gas de hidrocarburo no es soluble en las fases líquidas. Las ecuaciones de balance de masa y balance de energía se resuelven mediante la aplicación del algoritmo propuesto por Coats *et al.*, se utilizan algunos datos de inyección de vapor obtenidos en el laboratorio para estudiar la exactitud del modelo. Además, se muestran estudios comparativos de un yacimiento Iraní

junto con el análisis de sensibilidad de varios parámetros, los cuales pueden utilizar para mejorar el entendimiento del flujo de fluidos en los yacimientos de aceite pesado. Los autores concluyeron que la inyección de vapor puede mejorar el recobro de aceite de cero a casi 60 % durante un periodo fijo de tiempo. Los resultados demuestran que existe un tiempo óptimo para la producción de aceite, éste se determina de acuerdo a la razón del flujo de producción de aceite y el vapor. En el 2017, Yang *et al.* realizaron un estudio mediante una simulación numérica con el fin de investigar el rendimiento de la co-inyección de un proceso de aire y vapor a escala experimental [40]. Se desarrolló un modelo numérico radial tridimensional utilizando el software comercial STARS, para simular una prueba de inyección de vapor y una prueba de un tubo de combustión. También se llevó a cabo experimentalmente la inyección conjunta de aire enriquecido y vapor, la cual se realizó después de un período de inundación con agua caliente en la prueba del tubo de combustión. Con el fin de comprender los mecanismos de desplazamiento, se compararon y analizaron los resultados de la simulación y experimentación obtenidos en ambas pruebas, incluidos los perfiles de temperatura, la velocidad frontal del vapor, la saturación de aceite residual y la recuperación de aceite. Los autores encontraron que la co-inyección de vapor y aire enriquecido tiene el potencial de mejorar la recuperación de aceite. Ellos lograron un factor de recuperación final de alrededor del 90 % para la inyección conjunta del proceso de vapor y aire enriquecido, mientras que el factor de recuperación es de alrededor del 60 % para la prueba de inundación de vapor. Los autores concluyeron que estos porcentajes se deben a que la *combustion in situ* puede recuperar el aceite residual dejado por el barrido con vapor. Sin embargo, el vapor todavía juega un papel dominante en el desplazamiento del aceite.

Como se puede apreciar, hasta el día de hoy existen muchos y muy variados trabajos reportados para simular numéricamente la recuperación de aceite en yacimientos petroleros mediante la inyección de vapor. Cabe aclarar que todos los modelos numéricos parten de las ecuaciones de conservación de masa y energía, estos modelos han ido evolucionando en la forma de calcular las propiedades de flujo, términos fuente, y por su puesto modificaciones en los algoritmos de solución. Por esta razón, en este trabajo de tesis se seleccionó un modelo numérico trifásico de inyección de vapor, en el cual se propone la modificación de su algoritmo de solución para obtener más generalidad y su adaptación a las nuevas tecnologías de cómputo paralelo mediante el uso de GPU's.

Capítulo 3

Flujo de fluidos en medios porosos

En este capítulo, se introducen las ecuaciones de balance de masa y energía para el flujo de fluidos multifásicos en medios porosos. Estos se realizan con base en la formulación axiomática, la cual se explica ampliamente en el texto de Herrera I. y Pinder G. [41], y en la referencia [42].

3.1. Sistemas multifásicos en medios porosos

3.1.1. Ecuación general de balance de masa

La ecuación de balance local de una propiedad intensiva ψ , está dada por [41]:

$$\frac{\partial \psi}{\partial t} + \nabla \cdot (\vec{v}\psi) = \nabla \cdot \vec{\kappa} + g, \quad (3.1)$$

donde t representa el tiempo, \vec{v}_α es la velocidad del fluido, $\vec{\kappa}$ es la difusión y g 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 la misma (o advección). Del lado derecho observamos como primer término a la difusión y como segundo a la fuente (o sumidero).

En un sistema multifásico dentro de un medio poroso la masa del fluido de la fase α es una propiedad extensiva, la cual puede escribirse como [41]:

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

donde ϕ , es la porosidad del medio, ρ_α , es la densidad de la fase α , y S_α , se refiere a la saturación de la fase α . La correspondiente propiedad intensiva para la ecuación (3.2), es:

$$\psi_\alpha = \phi \rho_\alpha S_\alpha. \quad (3.3)$$

Sustituyendo ψ_α en la ecuación (3.1) y considerando que no existe difusión ($\kappa = 0$), se obtiene:

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

donde además de las sustituciones antes mencionadas se reemplazó la variable g por q , para estandarizar la notación que se tiene en el flujo de fluidos en medios porosos en problemas de ingeniería petrolera.

En el siglo XVIII el ingeniero francés Henry Darcy derivó mediante un experimento de drenado por gravedad en arena compactada una ecuación para calcular el caudal del flujo de fluidos en medios porosos en una fase. Esta ecuación se escribe como sigue:

$$Q = Ac \frac{h_2 - h_1}{L}, \quad (3.5)$$

donde A representa la sección transversal, $h_2 - h_1$ la diferencia de altura, c una constante de proporcionalidad y L la longitud del medio poroso. La constante c involucra las propiedades del medio poroso y del fluido. Pasando el área (A) del lado izquierdo de la ecuación (3.5) y sustituyendo las propiedades del medio y del fluido en la constante c para considerar una diferencia de presiones, se obtiene una forma más general de la ecuación (3.5), la cual se expresa de la siguiente manera:

$$\vec{u} = -\frac{\bar{k}}{\mu} (\nabla p - \rho\varphi\nabla z), \quad (3.6)$$

donde el tensor \bar{k} representa la permeabilidad del medio, μ se refiere a la viscosidad del fluido, ∇p es el gradiente de presión, ρ es la densidad del fluido, φ representa la magnitud de la gravedad y ∇z el gradiente de altura en la dirección z . La ecuación (3.6) se le conoce como el flujo de Darcy en medios porosos. El flujo de Darcy considerando más de un fluido, se define como [43]:

$$\vec{u}_\alpha = -\frac{\bar{k}k_{r\alpha}}{\mu_\alpha} (\nabla p_\alpha - \rho_\alpha\varphi\nabla z), \quad (3.7)$$

donde el subíndice α representa la fase. La nueva propiedad que aparece es la permeabilidad relativa de la fase α ($k_{r\alpha}$), la cual es adimensional puesto que se define como el cociente de la permeabilidad efectiva de la fase α (k_α) sobre una permeabilidad absoluta o base (k):

$$k_{r\alpha} = \frac{k_\alpha}{k}. \quad (3.8)$$

aquí k es un escalar y representa el valor de la permeabilidad absoluta. Por otra parte, la permeabilidad efectiva de la fase k_α depende de la saturación de la fase, es decir:

$$k_\alpha = k_\alpha(S_\alpha), \quad (3.9)$$

por lo tanto:

$$k_{r\alpha} = k_{r\alpha}(S_\alpha). \quad (3.10)$$

Tomando en cuenta el flujo de Darcy de la fase α se puede hacer la igualdad $\vec{u}_\alpha = \vec{v}_\alpha \phi S_\alpha$, puesto que el flujo de Darcy es función de la permeabilidad, la cual es la propiedad de flujo del medio poroso que considera los poros interconectados (porosidad efectiva) y también es función de la saturación de la fase mediante la consideración de la permeabilidad relativa. Sustituyendo \vec{u}_α por $\vec{v}_\alpha \phi S_\alpha$ en la ecuación (3.4), se obtiene:

$$\frac{\partial \phi \rho_\alpha S_\alpha}{\partial t} + \nabla \cdot (\rho_\alpha \vec{u}_\alpha) = q_\alpha. \quad (3.11)$$

Sustituyendo la ecuación (3.7) en la ecuación (3.11), se llega a:

$$\frac{\partial \phi \rho_\alpha S_\alpha}{\partial t} - \nabla \cdot \left(\rho_\alpha \frac{\bar{k} k_{r\alpha}}{\mu} (\nabla p_\alpha - \rho_\alpha \wp \nabla z) \right) = q_\alpha. \quad (3.12)$$

Si definimos el gradiente del potencial de la fase α como $\nabla \Phi_\alpha = \nabla p_\alpha - \rho_\alpha \wp \nabla z$, entonces la ecuación (3.12) queda de la siguiente forma:

$$\frac{\partial \phi \rho_\alpha S_\alpha}{\partial t} - \nabla \cdot \left(\rho_\alpha \frac{\bar{k} k_{r\alpha}}{\mu} \nabla \Phi_\alpha \right) = q_\alpha. \quad (3.13)$$

La ecuación (3.13) representa un número de ecuaciones para la conservación de masa igual al número de fases que se encuentran presentes en el medio poroso. Además la ecuación (3.13) se utiliza para llevar a cabo desarrollos posteriores en este trabajo de tesis.

Para completar el sistema de ecuaciones se tienen relaciones constitutivas las cuales involucran la dependencia de una variable respecto de la otra y estas se obtienen mediante correlaciones o por vía experimental. En un sistema multifásico se tienen las siguientes ecuaciones constitutivas:

$$\sum_{\alpha=1,n} S_\alpha = 1, \quad (3.14)$$

$$p_c = p_c(S_{mojanate}), \quad (3.15)$$

donde n es el número de fases, p_c se refiere a la presión capilar la cual es función de la saturación de la fase mojanante ($S_{mojanante}$). Además existen expresiones empíricas las cuales relacionan la permeabilidad relativa como función de la saturación $k_{r\alpha} = k_{r\alpha}(S_\alpha)$, la porosidad como función de la presión $\phi = \phi(p)$, etc. Estas relaciones se detallan en los siguientes capítulos del presente trabajo.

3.1.2. Ecuación general de balance de energía

Para modelar los procesos térmicos de EOR en medios porosos además de las ecuaciones de balance de masa, se necesita también una ecuación del balance total de energía, puesto que esta ecuación permite describir el transporte del calor en el medio poroso. Teniendo esto en cuenta, en un sistema multifásico dentro de un medio poroso la energía del fluido de la fase α es una propiedad extensiva, la cual puede escribirse como [42]:

$$E_\alpha = \int_{B(t)} \phi \rho_\alpha U_\alpha S_\alpha d\vec{x}, \quad (3.16)$$

donde U_α es la energía interna de la fase α dada en unidades de energía por unidad de masa, por ejemplo kJ/kg . Cabe señalar que en la ecuación (3.16) se desprecia la energía cinética debido a que la velocidad del fluido dentro de los poros es muy pequeña (véase [42]). La correspondiente propiedad intensiva para la ecuación (3.16), es:

$$\psi_\alpha = \phi \rho_\alpha U_\alpha S_\alpha. \quad (3.17)$$

Debido a que se está tratando con la energía total, el término fuente de la ecuación de balance local puede descomponerse en dos partes: fuentes de calor y fuentes de energía mecánica [41]. Esto se entiende mejor realizando un análisis integral, de tal forma que la ecuación de balance en la propiedad extensiva de la fase α , es:

$$\frac{dE_\alpha}{dt} = \int_{B(t)} \left\{ \rho_\alpha \left[\phi S_\alpha h_\alpha - \gamma (\vec{u}_\alpha)_z \right] - q_{L,\alpha} + g_{I,\alpha}^E \right\} d\vec{x} + \int_{\partial B(t)} \left(\phi S_\alpha \bar{\sigma}_\alpha \vec{v}_\alpha + \vec{q}_\alpha \right) \cdot \vec{n} d\vec{x}, \quad (3.18)$$

aquí los términos fuente son $\phi S_\alpha \rho_\alpha h_\alpha$ y $q_{L,\alpha}$, el primero representa el flujo por unidad de volumen del espacio en el que la energía interna es suministrada a la fase α por fuentes distribuidas en el interior del cuerpo (por ejemplo, reacciones exotérmicas), mientras que $q_{L,\alpha}$ es el calor de la fase α que se pierde por las fronteras de sobrecarga (*overburden*) y de bajocarga (*underburden*). El término $\gamma \rho_\alpha (\vec{u}_\alpha)_z$ representa el trabajo mecánico hecho por la gravedad (γ) en la fase α , donde $(\vec{u}_\alpha)_z$ se refiere a la componente de la velocidad en dirección z . El término $g_{I,\alpha}^E$ representa la energía total del intercambio proveniente de otras fases distintas a la fase α . Los flujos de energía provenientes a través de las fronteras están dados por $\phi S_\alpha \bar{\sigma}_\alpha \vec{v}_\alpha$ y \vec{q}_α , el primero representa el trabajo mecánico hecho en la frontera y el segundo el flujo de calor. Finalmente, se observa que:

$$\sum_{\alpha=w,o,g,s} g_{I,\alpha}^E = 0. \quad (3.19)$$

Para el caso particular de la ecuación de balance de energía aquí deducida, se desprecian el trabajo mecánico hecho por la gravedad, también se desprecian las fuerzas viscosas, por lo tanto, el tensor de esfuerzos que actúa sobre la fase α queda $\bar{\sigma}_\alpha = -p_\alpha \bar{I}$. De esta manera, se puede escribir:

$$\int_{\partial B(t)} \left(\phi S_\alpha \bar{\sigma}_\alpha \vec{v}_\alpha \right) \cdot \vec{n} d\vec{x} = - \int_{\partial B(t)} \left(p_\alpha \vec{u}_\alpha \right) \cdot \vec{n} d\vec{x}. \quad (3.20)$$

Sustituyendo (3.20) en (3.18) e incluyendo las consideraciones antes mencionadas, se obtiene:

$$\frac{dE_\alpha}{dt} = \int_{B(t)} \left\{ \rho_\alpha \phi S_\alpha h_\alpha - q_{L,\alpha} + g_{I,\alpha}^E \right\} d\vec{x} + \int_{\partial B(t)} \left(p_\alpha \vec{u}_\alpha + \vec{q}_\alpha \right) \cdot \vec{n} d\vec{x}. \quad (3.21)$$

Además, sustituyendo la propiedad intensiva definida previamente en la ecuación de balance global definida por Herrera [41], se llega a:

$$\frac{dE_\alpha}{dt} = \int_{B(t)} \frac{\partial \psi}{\partial t} + \nabla \cdot (\vec{v} \psi) d\vec{x} = \int_{B(t)} \frac{\partial \phi \rho_\alpha U_\alpha S_\alpha}{\partial t} + \nabla \cdot (\vec{v}_\alpha \phi \rho_\alpha U_\alpha S_\alpha) d\vec{x}. \quad (3.22)$$

Igualando las ecuaciones (3.22) y (3.21), aplicando el Teorema de la Divergencia al segundo término del lado derecho de la ecuación (3.21) y considerando que se cumplen para cualquier volumen, se obtiene la ecuación de balance de energía en su forma diferencial:

$$\frac{\partial \phi \rho_\alpha U_\alpha S_\alpha}{\partial t} + \nabla \cdot (\rho_\alpha U_\alpha \vec{u}_\alpha) = \nabla \cdot \left(\vec{q}_\alpha + p_\alpha \vec{u}_\alpha \right) + \rho_\alpha \phi S_\alpha h_\alpha - q_{L,\alpha} + g_{I,\alpha}^E. \quad (3.23)$$

Considerando la definición de entalpía $h_\alpha = U_\alpha + p_\alpha (\rho_\alpha)^{-1}$ y reordenando términos, la ecuación (3.23), puede escribirse como:

$$\frac{\partial \phi \rho_\alpha U_\alpha S_\alpha}{\partial t} + \nabla \cdot (\rho_\alpha h_\alpha \vec{u}_\alpha) = \nabla \cdot \vec{q}_\alpha + \rho_\alpha \phi S_\alpha h_\alpha - q_{L,\alpha} + g_{I,\alpha}^E. \quad (3.24)$$

Por otra parte, la ecuación de balance de energía en su forma diferencial para un sólido, es (véase [42]):

$$\frac{\partial ((1 - \phi) \rho_s U_s)}{\partial t} = \nabla \cdot \vec{q}_s - q_{L,s} + g_{I,s}^E. \quad (3.25)$$

Si se considera que el equilibrio térmico entre la roca y las fases es alcanzado instantáneamente, es decir, entre el medio poroso y los fluidos se tiene la misma temperatura T , aplicando la ley de Fourier para determinar el flujo de calor $\vec{q} = \bar{k} \nabla T$ y sumando las ecuaciones del

balance de energía de las fase de fluidos representadas por (3.24) y la ecuación de la fase del sólido (3.25), se llega a:

$$\frac{\partial}{\partial t} \left(\sum_{\alpha}^n (\phi U_{\alpha} \rho_{\alpha} S_{\alpha}) + (1 - \phi) U_s \rho_r \right) + \nabla \cdot \left(\sum_{\alpha}^n \rho_{\alpha} h_{\alpha} \vec{u}_{\alpha} \right) - \nabla \cdot \bar{\bar{k}}_T \nabla T = \sum_{\alpha}^n q_{\alpha} - q_L, \quad (3.26)$$

donde el término $\sum_{\alpha}^n q_{\alpha} = \sum_{\alpha}^n \rho_{\alpha} \phi S_{\alpha} h_{\alpha}$ representa el calor total transferido debido a los términos fuente, $\bar{\bar{k}}_T$ se refiere a la conductividad térmica total considerando los fluidos y el medio poroso. Se debe señalar que el transporte por dispersión hidrodinámica no se ha tomado en cuenta para deducir la ecuación (3.26). Por otra parte, la conductividad térmica total para la ecuación de balance de energía es muy difícil de determinar, pues ésta, es función de las propiedades del medio poroso, la saturación de los fluidos, y de las conductividades térmicas de los fluidos en cuestión ($\bar{\bar{k}}_T = \bar{\bar{k}}_T(\sum S_{\alpha} \bar{\bar{k}}_{term,\alpha}, \phi \bar{\bar{k}}_{term,s}, T)$). En la literatura existen varias correlaciones empíricas para determinar la conductividad térmica promedio de acuerdo a los tipos de fluidos y el tipo de medio poroso que los contiene [44]. Sin embargo, para los problemas numéricos planteados en este documento, $\bar{\bar{k}}_T$ se considera como un escalar de valor conocido.

Para un sistema de fluidos trifásico $n = 3$, entonces α puede reemplazarse por w, o y g donde estas literales representan la fase agua w , la fase aceite o y la fase gas g , además si se considera la energía interna del sólido como función lineal de su calor específico $U_s = C_{p,s} T$, la ecuación de balance de energía puede escribirse de la siguiente manera:

$$\begin{aligned} & \frac{\partial}{\partial t} [\phi (S_o u_o \rho_o + S_w u_w \rho_w + S_g u_g \rho_g) + (1 - \phi) (C_{p,s} \rho_s) T] = \\ & -\nabla \cdot [u_o \rho_o h_o + u_w \rho_w h_w + u_g \rho_g h_g] + \nabla \cdot \bar{\bar{k}}_T \nabla T + \sum_{\alpha} q_{\alpha} - q_L. \end{aligned} \quad (3.27)$$

La ecuación (3.27) es la base para calcular el transporte de energía en simuladores numéricos térmicos de flujo de fluidos en medios porosos.

Capítulo 4

Transformación de las ecuaciones

En este capítulo, se tratan los aspectos matemáticos relacionados con los sistemas de coordenadas curvilíneas y su relación con el sistema coordenado cartesiano, para su aplicación en la transformación de las ecuaciones de balance de masa y energía.

4.1. Conceptos básicos del mapeo de una región arbitraria a una región regular

Generalmente, los fenómenos físicos de flujo de fluidos tienen lugar en geometrías irregulares que no pueden ser representados por los sistemas coordenados ortogonales, entonces, para resolver numéricamente estos fenómenos es necesario realizar una transformación que permita mapear la región física arbitraria a una región computacional regular. La principal ventaja del mapeo de una región arbitraria a una región regular es la sencillez con que son aplicadas las condiciones de frontera en la región regular, lo cual evita el uso de interpolaciones u otras aproximaciones para la aplicación de las mismas.

Para ilustrar los conceptos básicos del mapeo se considera un dominio físico bidimensional en coordenadas cartesianas x, y y un dominio computacional en coordenadas cartesianas ξ, η . La transformación entre las coordenadas x, y y ξ, η se hace tal que las fronteras del dominio físico coinciden con las coordenadas curvilíneas, de esta manera no se necesita una interpolación para el nodo frontera. Las regiones físicas que son transformadas en el dominio computacional pueden ser identificadas en las dos categorías siguientes: región simplemente conectada y región múltiplemente conectada [45].

Región simplemente conectada: considere una región irregular $ABCD$ en el plano físico x, y en coordenadas cartesianas como se ilustra en la figura 4.1. La región es llamada simplemente conectada porque no contiene obstáculos dentro de la región. Esta región se mapea al dominio computacional en coordenadas cartesianas ξ, η de tal manera que la región mapeada tendrá una forma rectangular y permitirá la construcción de una malla cuadrada

sobre ella. Además, las fronteras del dominio físico coincidirán con las líneas coordenadas de las fronteras ξ, η de la región transformada en el dominio computacional. Una forma de realizar el mapeo es seleccionar los valores de ξ, η en dirección de las fronteras de la región física de la siguiente manera:

agrupar, $\eta = \text{constante}$, $\xi = \text{variante}$ en dirección de los segmentos frontera AB y CD en la región física y

agrupar, $\xi = \text{constante}$, $\eta = \text{variante}$ en dirección de los segmentos frontera BC y DA en la región física.

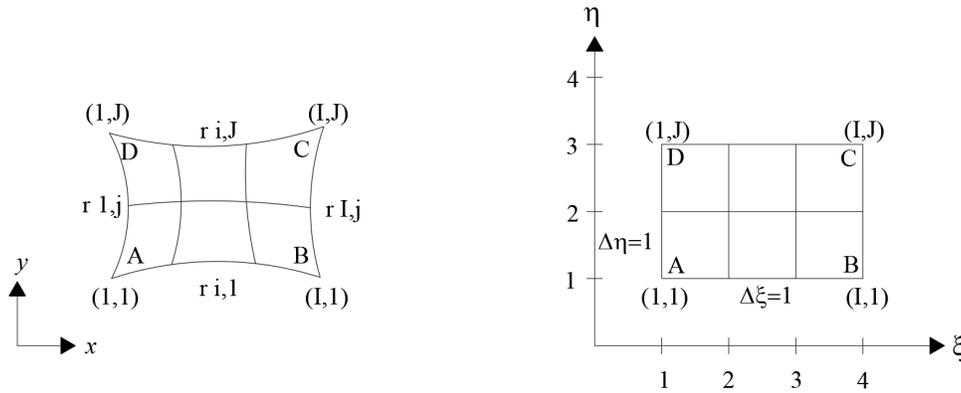


Figura 4.1: Mapeo de una región irregular simplemente conectada en la región computacional. (Ilustración adaptada de [45].)

Con tales requerimientos sobre los valores de ξ y η en dirección de las fronteras de la región física, los segmentos AB y CD en la región física son mapeados en el dominio computacional como líneas horizontales, mientras que los segmentos BC y DA son mapeados como líneas verticales como se ilustra en la figura 4.1. Se nota que, cada segmento frontera de la región irregular en el dominio físico es mapeado en los lados de una región rectangular en el dominio computacional.

Región múltiplemente conectada: ahora se considera una región irregular en el dominio físico con un obstáculo en la parte interior como se ilustra en la figura 4.2. La región mostrada en la figura 4.2 es una región doblemente conectada, porque hay un sólo obstáculo dentro de la región. Cuando hay más de un obstáculo dentro de la región, la región se llama: región múltiplemente conectada.

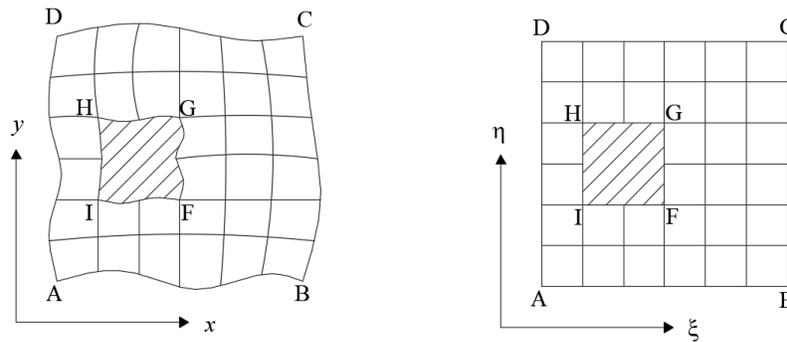


Figura 4.2: Mapeo de una región irregular doblemente conectada en una región rectangular. (Ilustración adaptada de [45].)

Se consideran las siguientes dos posibilidades para el mapeo de una región doblemente conectada:

1. En la primera aproximación, la región física irregular doblemente conectada se mapea en el dominio computacional como una región rectangular doblemente conectada con una ventana rectangular como se ilustra en la 4.2. Esto se lleva a cabo seleccionando los valores de ξ, η en dirección de las fronteras del dominio físico en la siguiente manera:

agrupar, $\eta = \text{constante}$, $\xi = \text{variante}$ en dirección de las fronteras AB y DC , EF y HG en la región física y

agrupar, $\xi = \text{constante}$, $\eta = \text{variante}$ en dirección de las fronteras AD , BC , EH y FG en la región física.

2. En la segunda aproximación, la región irregular doblemente conectada se mapea en el dominio computacional como una región simplemente conectada rectangular por un corte divisorio como se ilustra en la figura 4.3, el bosquejo superior de esta figura muestra como las pseudofronteras BC y AD se generan por el corte divisorio. El segundo bosquejo ilustra el procedimiento de alargamiento y el inferior muestra el mapeo final en la forma de una región rectangular simplemente conectada.

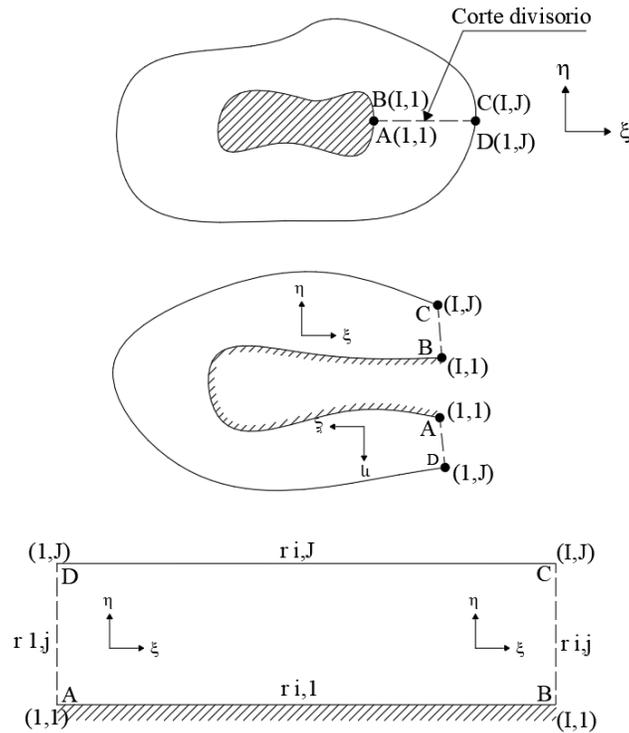


Figura 4.3: Mapeo de una región doblemente conectada en una región simplemente conectada usando un corte divisorio. (Ilustración adaptada de [45].)

Como se muestra en este apartado existen diferentes maneras de mapear una región física irregular a una región computacional regular, esto indica la importancia de identificar la región física a resolver.

4.2. Transformación de las ecuaciones gobernantes del flujo de fluidos en medios porosos

En esta sección se tratan los aspectos matemáticos relacionados con el sistema de coordenadas curvilíneas y su relación con el sistema coordenado cartesiano [46], para su aplicación posterior en la transformación de las ecuaciones al plano computacional.

4.2.1. Sistemas de coordenadas curvilíneas

En la figura 4.4 se muestra un sistema de coordenadas curvilíneas (ξ, η, ζ) referidas al sistema cartesiano (x, y, z) .

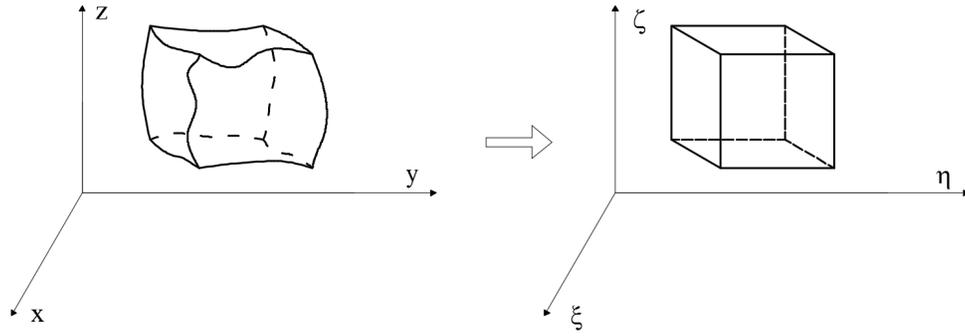


Figura 4.4: Sistema de coordenadas curvilíneas.
(Ilustración adaptada de [46].)

Las coordenadas curvilíneas de un punto, son referidas al sistema cartesiano con las siguientes ecuaciones:

$$\xi = \xi(x, y, z), \tag{4.1}$$

$$\eta = \eta(x, y, z), \tag{4.2}$$

$$\zeta = \zeta(x, y, z). \tag{4.3}$$

Existe la posibilidad de representar un nuevo sistema coordenado en movimiento con el tiempo, lo que altera en forma funcional las ecuaciones (4.1) a (4.3) para involucrar la variable tiempo.

Las métricas de esta transformación pueden obtenerse a través de la función inversa. Las diferenciales de cada eje coordenado, están dadas por:

$$d\xi = \xi_x dx + \xi_y dy + \xi_z dz, \tag{4.4}$$

$$d\eta = \eta_x dx + \eta_y dy + \eta_z dz, \tag{4.5}$$

$$d\zeta = \zeta_x dx + \zeta_y dy + \zeta_z dz, \tag{4.6}$$

donde ξ_i, η_i, ζ_i , son las derivadas parciales respecto al eje coordenado indicado en el subíndice, escribiendo las ecuaciones (4.4) a (4.6) en forma matricial:

$$\begin{pmatrix} d\xi \\ d\eta \\ d\zeta \end{pmatrix} = \begin{pmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{pmatrix} \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix}, \tag{4.7}$$

en forma compacta:

$$(d^T) = (A) (d^F), \tag{4.8}$$

donde d^T y d^F son las diferenciales en el dominio transformado y en el dominio físico respectivamente. Las diferenciales en el plano físico son:

$$\begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} = \begin{pmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{pmatrix} \begin{pmatrix} d\xi \\ d\eta \\ d\zeta \end{pmatrix}, \quad (4.9)$$

ó:

$$(d^F) = (B) (d^T). \quad (4.10)$$

Utilizando las ecuaciones (4.8) a (4.10), se llega a:

$$A = B^{-1} = J \begin{pmatrix} y_\eta z_\zeta - y_\zeta z_\eta & -(x_\eta z_\zeta - x_\zeta z_\eta) & x_\eta y_\zeta - x_\zeta y_\eta \\ -(y_\xi z_\zeta - y_\zeta z_\xi) & x_\xi z_\zeta - x_\zeta z_\xi & x_\xi y_\zeta - x_\zeta y_\xi \\ y_\xi z_\eta - y_\eta z_\xi & -(x_\xi z_\eta - x_\eta z_\xi) & x_\xi y_\eta - x_\eta y_\xi \end{pmatrix}, \quad (4.11)$$

luego, comparando (A) con (B^{-1}) , elemento por elemento, las métricas están dadas por:

$$\begin{aligned} \xi_x &= J (y_\eta z_\zeta - y_\zeta z_\eta), & \xi_y &= -J (x_\eta z_\zeta - x_\zeta z_\eta), & \xi_z &= J (x_\eta y_\zeta - x_\zeta y_\eta), \\ \eta_x &= -J (y_\xi z_\zeta - y_\zeta z_\xi), & \eta_y &= J (x_\xi z_\zeta - x_\zeta z_\xi), & \eta_z &= J (x_\xi y_\zeta - x_\zeta y_\xi), \\ \zeta_x &= J (y_\xi z_\eta - y_\eta z_\xi), & \zeta_y &= -J (x_\xi z_\eta - x_\eta z_\xi), & \zeta_z &= J (x_\xi y_\eta - x_\eta y_\xi), \end{aligned} \quad (4.12)$$

donde la literal J es conocida como el Jacobiano de la transformación, el cual está dado por:

$$J = \det(A) = \frac{1}{\det(B)}, \quad (4.13)$$

ó:

$$J = \{x_\xi (y_\eta z_\zeta - y_\zeta z_\eta) - x_\eta (y_\xi z_\zeta - y_\zeta z_\xi) + x_\zeta (y_\xi z_\eta - y_\eta z_\xi)\}. \quad (4.14)$$

Las ecuaciones (4.1) a (4.3) representan una transformación de un sistema (x, y, z) a un sistema (ξ, η, ζ) . La inversa de la transformación es:

$$x = x(\xi, \eta, \zeta), \quad (4.15)$$

$$y = y(\xi, \eta, \zeta), \quad (4.16)$$

$$z = z(\xi, \eta, \zeta), \quad (4.17)$$

donde las métricas de cada función inversa son:

$$\begin{aligned} x_\xi &= \frac{1}{J} (\eta_y \zeta_z - \eta_z \zeta_y), & x_\eta &= -\frac{1}{J} (\xi_y \zeta_z - \xi_z \zeta_y), & x_\zeta &= \frac{1}{J} (\eta_z \xi_y - \eta_y \xi_z), \\ y_\xi &= \frac{1}{J} (\eta_x \zeta_z - \eta_z \zeta_x), & y_\eta &= \frac{1}{J} (\xi_x \zeta_z - \xi_z \zeta_x), & y_\zeta &= -\frac{1}{J} (\xi_x \eta_z - \xi_z \eta_x), \\ z_\xi &= \frac{1}{J} (\eta_x \zeta_y - \eta_y \zeta_x), & z_\eta &= -\frac{1}{J} (\xi_x \zeta_y - \xi_y \zeta_x), & z_\zeta &= \frac{1}{J} (\xi_x \eta_y - \xi_y \eta_x). \end{aligned} \quad (4.18)$$

4.2.2. Transformación de la ecuación general de balance de masa en medios porosos

Se presenta el procedimiento para transformar la ecuación general de conservación de masa. La metodología presentada es similar a la reportada por Maliska, 1995 [46].

La ecuación de balance de masa de la fase α en medios porosos es:

$$\frac{\partial \phi \rho_\alpha S_\alpha}{\partial t} - \nabla \cdot \left(\rho_\alpha \frac{\bar{k} k_{r\alpha}}{\mu} (\nabla p_\alpha - \rho_\alpha \wp \nabla z) \right) = q_\alpha, \quad (4.19)$$

la cual puede escribirse en forma vectorial como:

$$\frac{\partial Q}{\partial t} - \nabla \cdot \vec{F} = S, \quad (4.20)$$

ó en la forma:

$$\frac{\partial Q}{\partial t} - \frac{\partial E}{\partial x} - \frac{\partial F}{\partial y} - \frac{\partial G}{\partial z} = S, \quad (4.21)$$

donde:

$$\vec{F} = E\vec{i} + F\vec{j} + G\vec{k}, \quad (4.22)$$

$$Q = \phi \rho_\alpha S_\alpha, \quad (4.23)$$

$$E = a_{11}^\alpha \frac{\partial p_\alpha}{\partial x} - b_{11}^\alpha \frac{\partial z}{\partial x} = \rho_\alpha k_{11} \frac{k_{r\alpha}}{\mu_\alpha} \frac{\partial p_\alpha}{\partial x} - \rho_\alpha k_{11} \frac{k_{r\alpha}}{\mu_\alpha} \rho_\alpha \wp \frac{\partial z}{\partial x}, \quad (4.24)$$

$$F = a_{22}^\alpha \frac{\partial p_\alpha}{\partial y} - b_{22}^\alpha \frac{\partial z}{\partial y} = \rho_\alpha k_{22} \frac{k_{r\alpha}}{\mu_\alpha} \frac{\partial p_\alpha}{\partial y} - \rho_\alpha k_{22} \frac{k_{r\alpha}}{\mu_\alpha} \rho_\alpha \wp \frac{\partial z}{\partial y}, \quad (4.25)$$

$$G = a_{33}^\alpha \frac{\partial p_\alpha}{\partial x} - b_{33}^\alpha \frac{\partial z}{\partial x} = \rho_\alpha k_{11} \frac{k_{r\alpha}}{\mu_\alpha} \frac{\partial p_\alpha}{\partial z} - \rho_\alpha k_{33} \frac{k_{r\alpha}}{\mu_\alpha} \rho_\alpha \wp \frac{\partial z}{\partial z}, \quad (4.26)$$

$$S = q_\alpha. \quad (4.27)$$

La transformación necesaria para la solución de los problemas transitorios tridimensionales, está dada por:

$$\xi = \xi(x, y, z), \quad (4.28)$$

$$\eta = \eta(x, y, z), \quad (4.29)$$

$$\zeta = \zeta(x, y, z), \quad (4.30)$$

$$\tau = t, \quad (4.31)$$

donde el Jacobiano y las métricas de la transformación son los mismos que en la sección anterior de este documento, además la letra griega τ representa el tiempo en el sistema coordenado transformado.

Usando la regla de la cadena, las derivadas de las componentes de flujo son:

$$\frac{\partial E}{\partial x} = \frac{\partial E}{\partial \xi} \xi_x + \frac{\partial E}{\partial \eta} \eta_x + \frac{\partial E}{\partial \zeta} \zeta_x, \quad (4.32)$$

$$\frac{\partial F}{\partial y} = \frac{\partial F}{\partial \xi} \xi_y + \frac{\partial F}{\partial \eta} \eta_y + \frac{\partial F}{\partial \zeta} \zeta_y, \quad (4.33)$$

$$\frac{\partial G}{\partial z} = \frac{\partial G}{\partial \xi} \xi_z + \frac{\partial G}{\partial \eta} \eta_z + \frac{\partial G}{\partial \zeta} \zeta_z, \quad (4.34)$$

$$\frac{\partial Q}{\partial t} = \frac{\partial Q}{\partial \tau}. \quad (4.35)$$

Introduciendo las ecuaciones anteriores en la ecuación (4.21) y dividiendo por J , se obtiene:

$$\begin{aligned} \frac{\partial Q}{\partial t} \left(\frac{1}{J} \right) - \left[\frac{\partial E}{\partial \xi} \frac{\xi_x}{J} + \frac{\partial E}{\partial \eta} \frac{\eta_x}{J} + \frac{\partial E}{\partial \zeta} \frac{\zeta_x}{J} \right] - \left[\frac{\partial F}{\partial \xi} \frac{\xi_y}{J} + \frac{\partial F}{\partial \eta} \frac{\eta_y}{J} + \frac{\partial F}{\partial \zeta} \frac{\zeta_y}{J} \right] \\ - \left[\frac{\partial G}{\partial \xi} \frac{\xi_z}{J} + \frac{\partial G}{\partial \eta} \frac{\eta_z}{J} + \frac{\partial G}{\partial \zeta} \frac{\zeta_z}{J} \right] = \frac{S}{J}. \end{aligned} \quad (4.36)$$

Para que las métricas y el Jacobiano queden dentro de la derivada, se aplica la regla del producto de las derivadas:

$$\begin{aligned} \frac{\partial}{\partial \xi} \left(\frac{E \xi_x}{J} \right) &= E \frac{\partial}{\partial \xi} \left(\frac{\xi_x}{J} \right) + \left(\frac{\xi_x}{J} \right) \frac{\partial E}{\partial \xi}, \\ \frac{\partial}{\partial \eta} \left(\frac{E \eta_x}{J} \right) &= E \frac{\partial}{\partial \eta} \left(\frac{\eta_x}{J} \right) + \left(\frac{\eta_x}{J} \right) \frac{\partial E}{\partial \eta}, \\ \frac{\partial}{\partial \zeta} \left(\frac{E \zeta_x}{J} \right) &= E \frac{\partial}{\partial \zeta} \left(\frac{\zeta_x}{J} \right) + \left(\frac{\zeta_x}{J} \right) \frac{\partial E}{\partial \zeta}, \\ \frac{\partial}{\partial \xi} \left(\frac{F \xi_y}{J} \right) &= F \frac{\partial}{\partial \xi} \left(\frac{\xi_y}{J} \right) + \left(\frac{\xi_y}{J} \right) \frac{\partial F}{\partial \xi}, \\ \frac{\partial}{\partial \eta} \left(\frac{F \eta_y}{J} \right) &= F \frac{\partial}{\partial \eta} \left(\frac{\eta_y}{J} \right) + \left(\frac{\eta_y}{J} \right) \frac{\partial F}{\partial \eta}, \\ \frac{\partial}{\partial \zeta} \left(\frac{F \zeta_y}{J} \right) &= F \frac{\partial}{\partial \zeta} \left(\frac{\zeta_y}{J} \right) + \left(\frac{\zeta_y}{J} \right) \frac{\partial F}{\partial \zeta}, \\ \frac{\partial}{\partial \xi} \left(\frac{G \xi_z}{J} \right) &= G \frac{\partial}{\partial \xi} \left(\frac{\xi_z}{J} \right) + \left(\frac{\xi_z}{J} \right) \frac{\partial G}{\partial \xi}, \\ \frac{\partial}{\partial \eta} \left(\frac{G \eta_z}{J} \right) &= G \frac{\partial}{\partial \eta} \left(\frac{\eta_z}{J} \right) + \left(\frac{\eta_z}{J} \right) \frac{\partial G}{\partial \eta}, \\ \frac{\partial}{\partial \zeta} \left(\frac{G \zeta_z}{J} \right) &= G \frac{\partial}{\partial \zeta} \left(\frac{\zeta_z}{J} \right) + \left(\frac{\zeta_z}{J} \right) \frac{\partial G}{\partial \zeta}, \\ \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) &= Q \frac{\partial}{\partial \tau} \left(\frac{1}{J} \right) + \left(\frac{1}{J} \right) \frac{\partial Q}{\partial \tau}. \end{aligned} \quad (4.37)$$

Despejando los términos de las ecuaciones (4.37) $\left(\frac{\xi_x}{J} \right) \frac{\partial E}{\partial \xi} = \frac{\partial}{\partial \xi} \left(\frac{E \xi_x}{J} \right) - E \frac{\partial}{\partial \xi} \left(\frac{\xi_x}{J} \right)$ y sustituyéndolos en la ecuación (4.36), resulta:

$$\begin{aligned}
& \left[\frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) - Q \frac{\partial}{\partial \tau} \left(\frac{1}{J} \right) \right] - \frac{\partial}{\partial \xi} \left(\frac{\xi_x E + \xi_y F + \xi_z G}{J} \right) - \frac{\partial}{\partial \eta} \left(\frac{\eta_x E + \eta_y F + \eta_z G}{J} \right) \\
& - \frac{\partial}{\partial \zeta} \left(\frac{\zeta_x E + \zeta_y F + \zeta_z G}{J} \right) + E \left[\frac{\partial}{\partial \xi} \left(\frac{\xi_x}{J} \right) + \frac{\partial}{\partial \eta} \left(\frac{\eta_x}{J} \right) + \frac{\partial}{\partial \zeta} \left(\frac{\zeta_x}{J} \right) \right] \\
& + F \left[\frac{\partial}{\partial \xi} \left(\frac{\xi_y}{J} \right) + \frac{\partial}{\partial \eta} \left(\frac{\eta_y}{J} \right) + \frac{\partial}{\partial \zeta} \left(\frac{\zeta_y}{J} \right) \right] \\
& + G \left[\frac{\partial}{\partial \xi} \left(\frac{\xi_z}{J} \right) + \frac{\partial}{\partial \eta} \left(\frac{\eta_z}{J} \right) + \frac{\partial}{\partial \zeta} \left(\frac{\zeta_z}{J} \right) \right] = \frac{S}{J}.
\end{aligned} \tag{4.38}$$

Con ayuda de las ecuaciones (4.12) y (4.18) se puede mostrar que, los últimos 3 términos entre corchetes en el lado izquierdo de la ecuación (4.38) son iguales a cero. Además la derivada temporal del Jacobiano del primer término entre corchetes es igual a cero, entonces la ecuación transformada, se reduce a:

$$\frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) - \frac{\partial}{\partial \xi} \left(\frac{\xi_x E + \xi_y F + \xi_z G}{J} \right) - \frac{\partial}{\partial \eta} \left(\frac{\eta_x E + \eta_y F + \eta_z G}{J} \right) - \frac{\partial}{\partial \zeta} \left(\frac{\zeta_x E + \zeta_y F + \zeta_z G}{J} \right) = \frac{S}{J}. \tag{4.39}$$

Aplicando la regla de la cadena para los términos E , F y G , se obtiene:

$$\begin{aligned}
E &= a_{11}^\alpha \frac{\partial p_\alpha}{\partial x} - b_{11}^\alpha \frac{\partial z}{\partial x} = a_{11}^\alpha \left(\xi_x \frac{\partial p_\alpha}{\partial \xi} + \eta_x \frac{\partial p_\alpha}{\partial \eta} + \zeta_x \frac{\partial p_\alpha}{\partial \zeta} \right) - b_{11}^\alpha \left(\xi_x \frac{\partial z}{\partial \xi} + \eta_x \frac{\partial z}{\partial \eta} + \zeta_x \frac{\partial z}{\partial \zeta} \right), \\
F &= a_{22}^\alpha \frac{\partial p_\alpha}{\partial y} - b_{22}^\alpha \frac{\partial z}{\partial y} = a_{22}^\alpha \left(\xi_y \frac{\partial p_\alpha}{\partial \xi} + \eta_y \frac{\partial p_\alpha}{\partial \eta} + \zeta_y \frac{\partial p_\alpha}{\partial \zeta} \right) - b_{22}^\alpha \left(\xi_y \frac{\partial z}{\partial \xi} + \eta_y \frac{\partial z}{\partial \eta} + \zeta_y \frac{\partial z}{\partial \zeta} \right), \\
G &= a_{33}^\alpha \frac{\partial p_\alpha}{\partial z} - b_{33}^\alpha \frac{\partial z}{\partial z} = a_{33}^\alpha \left(\xi_z \frac{\partial p_\alpha}{\partial \xi} + \eta_z \frac{\partial p_\alpha}{\partial \eta} + \zeta_z \frac{\partial p_\alpha}{\partial \zeta} \right) - b_{33}^\alpha \left(\xi_z \frac{\partial z}{\partial \xi} + \eta_z \frac{\partial z}{\partial \eta} + \zeta_z \frac{\partial z}{\partial \zeta} \right),
\end{aligned} \tag{4.40}$$

sustituyendo las ecuaciones (4.40) en (4.39), se llega a:

$$\begin{aligned}
& \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) - \frac{\partial}{\partial \xi} \left\{ \frac{1}{J} \left[\xi_x a_{11}^\alpha \left(\xi_x \frac{\partial p_\alpha}{\partial \xi} + \eta_x \frac{\partial p_\alpha}{\partial \eta} + \zeta_x \frac{\partial p_\alpha}{\partial \zeta} \right) - \xi_x b_{11}^\alpha \left(\xi_x \frac{\partial z}{\partial \xi} + \eta_x \frac{\partial z}{\partial \eta} + \zeta_x \frac{\partial z}{\partial \zeta} \right) \right. \right. \\
& + \xi_y a_{22}^\alpha \left(\xi_y \frac{\partial p_\alpha}{\partial \xi} + \eta_y \frac{\partial p_\alpha}{\partial \eta} + \zeta_y \frac{\partial p_\alpha}{\partial \zeta} \right) - \xi_y b_{22}^\alpha \left(\xi_y \frac{\partial z}{\partial \xi} + \eta_y \frac{\partial z}{\partial \eta} + \zeta_y \frac{\partial z}{\partial \zeta} \right) \\
& + \left. \left. \xi_z a_{33}^\alpha \left(\xi_z \frac{\partial p_\alpha}{\partial \xi} + \eta_z \frac{\partial p_\alpha}{\partial \eta} + \zeta_z \frac{\partial p_\alpha}{\partial \zeta} \right) - \xi_z b_{33}^\alpha \left(\xi_z \frac{\partial z}{\partial \xi} + \eta_z \frac{\partial z}{\partial \eta} + \zeta_z \frac{\partial z}{\partial \zeta} \right) \right] \right\} \\
& - \frac{\partial}{\partial \eta} \left\{ \frac{1}{J} \left[\eta_x a_{11}^\alpha \left(\xi_x \frac{\partial p_\alpha}{\partial \xi} + \eta_x \frac{\partial p_\alpha}{\partial \eta} + \zeta_x \frac{\partial p_\alpha}{\partial \zeta} \right) - \eta_x b_{11}^\alpha \left(\xi_x \frac{\partial z}{\partial \xi} + \eta_x \frac{\partial z}{\partial \eta} + \zeta_x \frac{\partial z}{\partial \zeta} \right) \right. \right. \\
& + \eta_y a_{22}^\alpha \left(\xi_y \frac{\partial p_\alpha}{\partial \xi} + \eta_y \frac{\partial p_\alpha}{\partial \eta} + \zeta_y \frac{\partial p_\alpha}{\partial \zeta} \right) - \eta_y b_{22}^\alpha \left(\xi_y \frac{\partial z}{\partial \xi} + \eta_y \frac{\partial z}{\partial \eta} + \zeta_y \frac{\partial z}{\partial \zeta} \right) \\
& + \left. \left. \eta_z a_{33}^\alpha \left(\xi_z \frac{\partial p_\alpha}{\partial \xi} + \eta_z \frac{\partial p_\alpha}{\partial \eta} + \zeta_z \frac{\partial p_\alpha}{\partial \zeta} \right) - \eta_z b_{33}^\alpha \left(\xi_z \frac{\partial z}{\partial \xi} + \eta_z \frac{\partial z}{\partial \eta} + \zeta_z \frac{\partial z}{\partial \zeta} \right) \right] \right\} \\
& - \frac{\partial}{\partial \zeta} \left\{ \frac{1}{J} \left[\zeta_x a_{11}^\alpha \left(\xi_x \frac{\partial p_\alpha}{\partial \xi} + \eta_x \frac{\partial p_\alpha}{\partial \eta} + \zeta_x \frac{\partial p_\alpha}{\partial \zeta} \right) - \zeta_x b_{11}^\alpha \left(\xi_x \frac{\partial z}{\partial \xi} + \eta_x \frac{\partial z}{\partial \eta} + \zeta_x \frac{\partial z}{\partial \zeta} \right) \right. \right. \\
& + \zeta_y a_{22}^\alpha \left(\xi_y \frac{\partial p_\alpha}{\partial \xi} + \eta_y \frac{\partial p_\alpha}{\partial \eta} + \zeta_y \frac{\partial p_\alpha}{\partial \zeta} \right) - \zeta_y b_{22}^\alpha \left(\xi_y \frac{\partial z}{\partial \xi} + \eta_y \frac{\partial z}{\partial \eta} + \zeta_y \frac{\partial z}{\partial \zeta} \right) \\
& + \left. \left. \zeta_z a_{33}^\alpha \left(\xi_z \frac{\partial p_\alpha}{\partial \xi} + \eta_z \frac{\partial p_\alpha}{\partial \eta} + \zeta_z \frac{\partial p_\alpha}{\partial \zeta} \right) - \zeta_z b_{33}^\alpha \left(\xi_z \frac{\partial z}{\partial \xi} + \eta_z \frac{\partial z}{\partial \eta} + \zeta_z \frac{\partial z}{\partial \zeta} \right) \right] \right\} = \frac{S}{J},
\end{aligned} \tag{4.41}$$

rearrreglando términos la ecuación resultante es:

$$\begin{aligned} \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) &= \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_\alpha}{\partial \xi} + A_{12} \frac{\partial p_\alpha}{\partial \eta} + A_{13} \frac{\partial p_\alpha}{\partial \zeta} - B_{11} \frac{\partial z}{\partial \xi} - B_{12} \frac{\partial z}{\partial \eta} - B_{13} \frac{\partial z}{\partial \zeta} \right] \\ &+ \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_\alpha}{\partial \xi} + A_{22} \frac{\partial p_\alpha}{\partial \eta} + A_{23} \frac{\partial p_\alpha}{\partial \zeta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} - B_{23} \frac{\partial z}{\partial \zeta} \right] \\ &+ \frac{\partial}{\partial \zeta} \left[A_{31} \frac{\partial p_\alpha}{\partial \xi} + A_{32} \frac{\partial p_\alpha}{\partial \eta} + A_{33} \frac{\partial p_\alpha}{\partial \zeta} - B_{31} \frac{\partial z}{\partial \xi} - B_{32} \frac{\partial z}{\partial \eta} - B_{33} \frac{\partial z}{\partial \zeta} \right] + \frac{S}{J}, \end{aligned} \quad (4.42)$$

donde las componentes del tensor $A_{i,j}$ y $B_{i,j}$, están dadas por:

$$\begin{aligned} A_{11} &= \frac{1}{J} (a_{11}^\alpha \xi_x^2 + a_{22}^\alpha \xi_y^2 + a_{33}^\alpha \xi_z^2), & B_{11} &= \frac{1}{J} (b_{11}^\alpha \xi_x^2 + b_{22}^\alpha \xi_y^2 + b_{33}^\alpha \xi_z^2), \\ A_{12} &= \frac{1}{J} (a_{11}^\alpha \xi_x \eta_x + a_{22}^\alpha \xi_y \eta_y + a_{33}^\alpha \xi_z \eta_z), & B_{12} &= \frac{1}{J} (b_{11}^\alpha \xi_x \eta_x + b_{22}^\alpha \xi_y \eta_y + b_{33}^\alpha \xi_z \eta_z), \\ A_{13} &= \frac{1}{J} (a_{11}^\alpha \xi_x \zeta_x + a_{22}^\alpha \xi_y \zeta_y + a_{33}^\alpha \xi_z \zeta_z), & B_{13} &= \frac{1}{J} (b_{11}^\alpha \xi_x \zeta_x + b_{22}^\alpha \xi_y \zeta_y + b_{33}^\alpha \xi_z \zeta_z), \\ A_{21} &= \frac{1}{J} (a_{11}^\alpha \xi_x \eta_x + a_{22}^\alpha \xi_y \eta_y + a_{33}^\alpha \xi_z \eta_z), & B_{21} &= \frac{1}{J} (b_{11}^\alpha \xi_x \eta_x + b_{22}^\alpha \xi_y \eta_y + b_{33}^\alpha \xi_z \eta_z), \\ A_{22} &= \frac{1}{J} (a_{11}^\alpha \eta_x^2 + a_{22}^\alpha \eta_y^2 + a_{33}^\alpha \eta_z^2), & B_{22} &= \frac{1}{J} (b_{11}^\alpha \eta_x^2 + b_{22}^\alpha \eta_y^2 + b_{33}^\alpha \eta_z^2), \\ A_{23} &= \frac{1}{J} (a_{11}^\alpha \eta_x \zeta_x + a_{22}^\alpha \eta_y \zeta_y + a_{33}^\alpha \eta_z \zeta_z), & B_{23} &= \frac{1}{J} (b_{11}^\alpha \eta_x \zeta_x + b_{22}^\alpha \eta_y \zeta_y + b_{33}^\alpha \eta_z \zeta_z), \\ A_{31} &= \frac{1}{J} (a_{11}^\alpha \xi_x \zeta_x + a_{22}^\alpha \xi_y \zeta_y + a_{33}^\alpha \xi_z \zeta_z), & B_{31} &= \frac{1}{J} (b_{11}^\alpha \xi_x \zeta_x + b_{22}^\alpha \xi_y \zeta_y + b_{33}^\alpha \xi_z \zeta_z), \\ A_{32} &= \frac{1}{J} (a_{11}^\alpha \eta_x \zeta_x + a_{22}^\alpha \eta_y \zeta_y + a_{33}^\alpha \eta_z \zeta_z), & B_{32} &= \frac{1}{J} (b_{11}^\alpha \eta_x \zeta_x + b_{22}^\alpha \eta_y \zeta_y + b_{33}^\alpha \eta_z \zeta_z), \\ A_{33} &= \frac{1}{J} (a_{11}^\alpha \zeta_x^2 + a_{22}^\alpha \zeta_y^2 + a_{33}^\alpha \zeta_z^2), & B_{33} &= \frac{1}{J} (b_{11}^\alpha \zeta_x^2 + b_{22}^\alpha \zeta_y^2 + b_{33}^\alpha \zeta_z^2). \end{aligned} \quad (4.43)$$

Esta ecuación nos permite relacionar el plano computacional con cualquier geometría irregular del plano físico.

4.2.3. Transformación de la ecuación de balance de energía en medios porosos

Para la transformación de la ecuación del balance de energía se lleva a cabo un procedimiento similar a la transformación de las ecuaciones de balance de masa. La ecuación de balance de energía, está definida como:

$$\begin{aligned} \frac{\partial}{\partial t} [\phi (S_o \rho_o U_o + S_w \rho_w U_w + S_g \rho_g U_g) + (1 - \phi) (C_{p,s} \rho_s) T] = \\ -\nabla \cdot [u_o \rho_o h_o + u_w \rho_w h_w + u_g \rho_g h_g] + \nabla \cdot \bar{k}_T \nabla T + \sum_{\alpha} q_{\alpha} - q_L. \end{aligned} \quad (4.44)$$

Siguiendo el mismo procedimiento que para la transformación de la ecuación de balance de masa, se puede escribir la ecuación de balance de energía como:

$$\frac{\partial Q}{\partial t} - \frac{\partial E}{\partial x} - \frac{\partial F}{\partial y} - \frac{\partial G}{\partial z} = S, \quad (4.45)$$

donde:

$$Q = \phi \sum_{\alpha} S_{\alpha} \rho_{\alpha} U_{\alpha} + (1 - \phi) (\rho C)_s T, \quad (4.46)$$

$$E = \sum_{\alpha} a_{11}^{\alpha} \frac{\partial p_{\alpha}}{\partial x} - \sum_{\alpha} b_{11}^{\alpha} \frac{\partial z}{\partial x} + c_{11} \frac{\partial T}{\partial x} = \sum_{\alpha} h_{\alpha} \rho_{\alpha} k_{11} \frac{k_{r\alpha}}{\mu_{\alpha}} \frac{\partial p_{\alpha}}{\partial x} - \sum_{\alpha} h_{\alpha} \rho_{\alpha} k_{11} \frac{k_{r\alpha}}{\mu_{\alpha}} \rho_{\alpha} \varphi \frac{\partial z}{\partial x} + k_{T11} \frac{\partial T}{\partial x}, \quad (4.47)$$

$$F = \sum_{\alpha} a_{22}^{\alpha} \frac{\partial p_{\alpha}}{\partial y} - \sum_{\alpha} b_{22}^{\alpha} \frac{\partial z}{\partial y} + c_{22} \frac{\partial T}{\partial y} = \sum_{\alpha} h_{\alpha} \rho_{\alpha} k_{22} \frac{k_{r\alpha}}{\mu_{\alpha}} \frac{\partial p_{\alpha}}{\partial y} - \sum_{\alpha} h_{\alpha} \rho_{\alpha} k_{22} \frac{k_{r\alpha}}{\mu_{\alpha}} \rho_{\alpha} \varphi \frac{\partial z}{\partial y} + k_{T22} \frac{\partial T}{\partial y}, \quad (4.48)$$

$$G = \sum_{\alpha} a_{33}^{\alpha} \frac{\partial p_{\alpha}}{\partial z} - \sum_{\alpha} b_{33}^{\alpha} \frac{\partial z}{\partial z} + c_{33} \frac{\partial T}{\partial z} = \sum_{\alpha} h_{\alpha} \rho_{\alpha} k_{11} \frac{k_{r\alpha}}{\mu_{\alpha}} \frac{\partial p_{\alpha}}{\partial z} - \sum_{\alpha} h_{\alpha} \rho_{\alpha} k_{33} \frac{k_{r\alpha}}{\mu_{\alpha}} \rho_{\alpha} \varphi \frac{\partial z}{\partial z} + k_{T33} \frac{\partial T}{\partial z}, \quad (4.49)$$

$$S = \sum_{\alpha} q_{\alpha} - q_L. \quad (4.50)$$

La ecuación de balance de energía en el plano computacional transformado, se puede escribir como sigue:

$$\begin{aligned} \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) = \frac{\partial}{\partial \xi} \left[\sum_{\alpha} \left(A_{11} \frac{\partial p_{\alpha}}{\partial \xi} + A_{12} \frac{\partial p_{\alpha}}{\partial \eta} + A_{13} \frac{\partial p_{\alpha}}{\partial \zeta} - B_{11} \frac{\partial z}{\partial \xi} - B_{12} \frac{\partial z}{\partial \eta} - B_{13} \frac{\partial z}{\partial \zeta} \right) + C_{11} \frac{\partial T}{\partial \xi} + C_{12} \frac{\partial T}{\partial \eta} + C_{13} \frac{\partial T}{\partial \zeta} \right] \\ + \frac{\partial}{\partial \eta} \left[\sum_{\alpha} \left(A_{21} \frac{\partial p_{\alpha}}{\partial \xi} + A_{22} \frac{\partial p_{\alpha}}{\partial \eta} + A_{23} \frac{\partial p_{\alpha}}{\partial \zeta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} - B_{23} \frac{\partial z}{\partial \zeta} \right) + C_{21} \frac{\partial T}{\partial \xi} + C_{22} \frac{\partial T}{\partial \eta} + C_{23} \frac{\partial T}{\partial \zeta} \right] \\ + \frac{\partial}{\partial \zeta} \left[\sum_{\alpha} \left(A_{31} \frac{\partial p_{\alpha}}{\partial \xi} + A_{32} \frac{\partial p_{\alpha}}{\partial \eta} + A_{33} \frac{\partial p_{\alpha}}{\partial \zeta} - B_{31} \frac{\partial z}{\partial \xi} - B_{32} \frac{\partial z}{\partial \eta} - B_{33} \frac{\partial z}{\partial \zeta} \right) + C_{31} \frac{\partial T}{\partial \xi} + C_{32} \frac{\partial T}{\partial \eta} + C_{33} \frac{\partial T}{\partial \zeta} \right] + \frac{S}{J}, \end{aligned} \quad (4.51)$$

aquí las componentes de los tensores $A_{i,j}$ y $B_{i,j}$ son similares a aquellos dados en la ecuación de balance de masa. Por último, los coeficientes $C_{(i,j)}$, están definidos por:

$$C_{11} = \frac{1}{J} (k_{T11}\xi_x^2 + k_{T22}\xi_y^2 + k_{T33}\xi_z^2), \quad (4.52)$$

$$C_{12} = \frac{1}{J} (k_{T11}\xi_x\eta_x + k_{T22}\xi_y\eta_y + k_{T33}\xi_z\eta_z), \quad (4.53)$$

$$C_{13} = \frac{1}{J} (k_{T11}\xi_x\zeta_x + k_{T22}\xi_y\zeta_y + k_{T33}\xi_z\zeta_z), \quad (4.54)$$

$$C_{21} = \frac{1}{J} (k_{T11}\xi_x\eta_x + k_{T22}\xi_y\eta_y + k_{T33}\xi_z\eta_z), \quad (4.55)$$

$$C_{22} = \frac{1}{J} (k_{T11}\eta_x^2 + k_{T22}\eta_y^2 + k_{T33}\eta_z^2), \quad (4.56)$$

$$C_{23} = \frac{1}{J} (k_{T11}\eta_x\zeta_x + k_{T22}\eta_y\zeta_y + k_{T33}\eta_z\zeta_z), \quad (4.57)$$

$$C_{31} = \frac{1}{J} (k_{T11}\xi_x\zeta_x + k_{T22}\xi_y\zeta_y + k_{T33}\xi_z\zeta_z), \quad (4.58)$$

$$C_{32} = \frac{1}{J} (k_{T11}\eta_x\zeta_x + k_{T22}\eta_y\zeta_y + \eta_z\zeta_z), \quad (4.59)$$

$$C_{33} = \frac{1}{J} (k_{T11}\zeta_x^2 + k_{T22}\zeta_y^2 + k_{T33}\zeta_z^2). \quad (4.60)$$

Las ecuaciones transformadas (4.42) y (4.51), se utilizan en su forma bidimensional para simular numéricamente el proceso de inyección de agua y de vapor mediante el método de volumen finito. Su uso se explica de manera detallada en los capítulos 5 y 6. Adicionalmente, la discretización de las mismas se presenta en el Apéndice A.

Capítulo 5

Modelado numérico del proceso de inyección de agua

En este capítulo, se abordan las técnicas numéricas y el algoritmo computacional para resolver el modelo matemático para llevar a cabo la simulación del proceso de inyección de agua para la recuperación de hidrocarburos. El modelo matemático está basado en la formulación presión-saturación, la discretización se lleva a cabo mediante el Método de Volumen Finito, la linealización y solución se realiza aplicando la técnica numérica de Newton-Raphson. La metodología presentada en este capítulo, es la base para el desarrollo de la solución numérica del modelo matemático para simular el proceso de inyección de vapor. Si el lector está familiarizado con este tipo de técnicas numéricas, puede consultar el artículo V.L. Teja-Juárez y L.M. de la Cruz [47], donde se proporcionan los modelos numérico y computacional de una manera mucho más concreta, además de los resultados numéricos obtenidos.

5.1. Formulación Presión-Saturación

Reescribiendo la ecuación general de balance de masa en medios porosos en términos de la movilidad de la fase λ_α :

$$\frac{\partial \phi \rho_\alpha S_\alpha}{\partial t} - \nabla \cdot \left(\rho_\alpha \bar{k} \lambda_\alpha (\nabla p_\alpha - \rho_\alpha \varphi \nabla z) \right) = q_\alpha, \quad (5.1)$$

donde $\lambda_\alpha = \frac{k_{r\alpha}}{\mu_\alpha}$.

El sistema de ecuaciones que produce dicha ecuación se encuentra fuertemente acoplado, si se consideran 2 o más fases. Esto quiere decir que las incógnitas de las ecuaciones, por ejemplo, la presión y saturación de la fase dependen una de la otra y se encuentran explícitamente en la misma ecuación. La formulación de flujo fraccional de un sistema de N fases, desacopla el sistema de ecuaciones diferenciales producido por (5.1), en 1 ecuación para la presión y $N - 1$ ecuaciones de transporte para las saturaciones. Esta formulación produce un sistema

de ecuaciones débilmente acoplado que puede resolverse de manera iterativa y secuencial [44].

Ecuación de presión

Para obtener la ecuación de la presión, primero se divide la ecuación (5.1) por la densidad ρ_α :

$$\frac{1}{\rho_\alpha} \left[\frac{\partial(\phi \rho_\alpha S_\alpha)}{\partial t} + \nabla \cdot (\rho_\alpha \vec{u}_\alpha) - q_\alpha \right] = 0, \quad (5.2)$$

sumando sobre todas las fases, se llega a:

$$\sum_\alpha \left\{ \frac{1}{\rho_\alpha} \left[\frac{\partial(\phi \rho_\alpha S_\alpha)}{\partial t} + \nabla \cdot (\rho_\alpha \vec{u}_\alpha) - q_\alpha \right] \right\} = 0, \quad (5.3)$$

desarrollando las derivadas mediante la regla del producto, se obtiene:

$$\sum_\alpha \left\{ \frac{1}{\rho_\alpha} \left[\rho_\alpha S_\alpha \frac{\partial \phi}{\partial t} + \phi S_\alpha \frac{\partial \rho_\alpha}{\partial t} + \rho_\alpha \phi \frac{\partial S_\alpha}{\partial t} + \rho_\alpha \nabla \cdot \vec{u}_\alpha + \vec{u}_\alpha \nabla \cdot \rho_\alpha - q_\alpha \right] \right\} = 0, \quad (5.4)$$

aplicando la ecuación constitutiva (3.14) a la ecuación (5.4), se llega a la siguiente expresión:

$$\frac{\partial \phi}{\partial t} + \sum_\alpha \nabla \cdot \vec{u}_\alpha + \sum_\alpha \frac{1}{\rho_\alpha} \left[\phi S_\alpha \frac{\partial \rho_\alpha}{\partial t} + \vec{u}_\alpha \nabla \cdot \rho_\alpha \right] - \sum_\alpha \frac{q_\alpha}{\rho_\alpha} = 0. \quad (5.5)$$

Definiendo la velocidad total \vec{u} como:

$$\vec{u} = \sum_\alpha \vec{u}_\alpha, \quad (5.6)$$

se tiene la siguiente relación:

$$\nabla \cdot \vec{u} = \nabla \cdot \sum_\alpha \vec{u}_\alpha = \sum_\alpha \nabla \cdot \vec{u}_\alpha, \quad (5.7)$$

Sustituyendo la ecuación de la velocidad total (5.7) en (5.5), se llega a:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \vec{u} + \sum_\alpha \frac{1}{\rho_\alpha} \left[\phi S_\alpha \frac{\partial \rho_\alpha}{\partial t} + \vec{u}_\alpha \nabla \cdot \rho_\alpha \right] - \sum_\alpha \frac{q_\alpha}{\rho_\alpha} = 0. \quad (5.8)$$

Aplicando la definición de la velocidad de Darcy (3.7) para flujo multifásico y la relación de movilidad λ_α :

$$\vec{u} = \sum_\alpha \vec{u}_\alpha = \sum_\alpha \left[-\bar{k} \lambda_\alpha (\nabla p_\alpha - \rho_\alpha \wp \nabla z) \right]. \quad (5.9)$$

Definiendo la función de flujo fraccional como:

$$f_\alpha = \frac{\lambda_\alpha}{\lambda} \rightarrow \lambda_\alpha = f_\alpha \lambda, \quad (5.10)$$

donde $\lambda = \sum_\alpha \lambda_\alpha$ es la movilidad total y por lo tanto $\sum_\alpha f_\alpha = 1$. Insertando la ecuación (5.10) en la ecuación (5.9), se obtiene:

$$\vec{u} = -\bar{k}\lambda \left[\sum_\alpha f_\alpha \nabla p_\alpha - \sum_\alpha f_\alpha \rho_\alpha \wp \nabla z \right]. \quad (5.11)$$

Sustituyendo la ecuación de la velocidad total (5.11) en (5.8), se obtiene la ecuación de la presión en función del flujo fraccional:

$$\frac{\partial \phi}{\partial t} - \nabla \cdot \bar{k}\lambda \left[\sum_\alpha f_\alpha \nabla p_\alpha - \sum_\alpha f_\alpha \rho_\alpha \wp \nabla z \right] + \sum_\alpha \frac{1}{\rho_\alpha} \left[\phi S_\alpha \frac{\partial \rho_\alpha}{\partial t} + \vec{u}_\alpha \nabla \cdot \rho_\alpha \right] - \sum_\alpha \frac{q_\alpha}{\rho_\alpha} = 0. \quad (5.12)$$

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:

$$S_w + S_o = 1, \quad (5.13)$$

$$p_c = p_{cow} = p_o - p_w. \quad (5.14)$$

Considerando el sistema de ecuaciones (5.1) y sustituyendo para ambas fases:

$$\frac{\partial (\phi \rho_w S_w)}{\partial t} - \nabla \cdot \left(\rho_w \bar{k} \lambda_w (\nabla p_w - \rho_w \wp \nabla z) \right) = q_w, \quad (5.15)$$

$$\frac{\partial (\phi \rho_o S_o)}{\partial t} - \nabla \cdot \left(\rho_o \bar{k} \lambda_o (\nabla p_o - \rho_o \wp \nabla z) \right) = q_o, \quad (5.16)$$

donde la ecuación (5.15) es la ecuación de balance de masa para la fase agua y la ecuación (5.16) es la ecuación de balance de masa para la fase aceite. Para la ecuación de la fase agua, se tiene:

$$\frac{\partial (\phi \rho_w S_w)}{\partial t} - \nabla \cdot \left(\rho_w \bar{k} \lambda_w (\nabla (p_o - p_c) - \rho_w \wp \nabla z) \right) = q_w, \quad (5.17)$$

expandiendo las derivadas del gradiente de la ecuación (5.20), se llega a la ecuación de conservación de masa de la fase w en función la presión capilar y la presión del aceite:

$$\frac{\partial (\phi \rho_w S_w)}{\partial t} - \nabla \cdot \left(\rho_w \bar{k} \lambda_w ((\nabla p_o - \nabla p_c) - \rho_w \wp \nabla z) \right) = q_w. \quad (5.18)$$

Las ecuaciones (5.12) y (5.18) se utilizan como punto de partida para desarrollar el modelo numérico para simular el proceso de inyección de agua en este documento.

5.2. Modelo conceptual para flujo incompresible

Se considera que ocurre flujo bifásico en un medio poroso donde existe desplazamiento de aceite por inyección de agua a caudal constante en condiciones isotérmicas.

- Se inyecta agua para desplazar el aceite y posteriormente producirlo.
- Se consideran los efectos de la presión capilar y los efectos de la gravedad.
- Las fluidos son incompresibles e inmiscibles.

Con las consideraciones anteriores la ecuación de conservación de masa de la fase w (5.18) 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:

$$\nabla p_c = \frac{dp_c}{dS_w} \nabla S_w. \quad (5.19)$$

Sustituyendo (5.19) en la ecuación (5.18), se llega a:

$$\phi \frac{\partial S_w}{\partial t} - \nabla \cdot \left[\bar{k} \lambda_w \left(\nabla p_o - \frac{dp_c}{dS_w} \nabla S_w - \rho_w \wp \nabla z \right) \right] = \frac{q_w}{\rho_w}. \quad (5.20)$$

Ahora aplicando las consideraciones hechas a la ecuación de conservación de masa de la fase aceite (5.16), se obtiene:

$$-\phi \frac{\partial S_o}{\partial t} - \nabla \cdot \left(\bar{k} \lambda_o (\nabla p_o - \rho_o \wp \nabla z) \right) = \frac{q_o}{\rho_o}. \quad (5.21)$$

Básicamente, se tienen dos ecuaciones diferenciales parciales con dos incógnitas si se consideran las propiedades del medio poroso y del fluido como conocidas. Las variables a resolver son la presión de aceite p_o y la saturación de agua S_w , puesto que si se conoce la saturación de agua también se conoce la saturación del aceite. Por otra parte, si se conoce la presión capilar; la presión de la fase agua se puede calcular mediante $p_w = p_o - p_c$.

Para facilitar la solución de las ecuaciones (5.20) y (5.21) se utiliza una formulación de Presión-Saturación.

Partiendo de la ecuación de presión general y aplicando las consideraciones del modelo conceptual a la ecuación de presión inmediatamente anterior, se llega a:

$$\begin{aligned} \frac{\partial \phi}{\partial t} - \nabla \cdot \bar{k} \lambda \left[\sum_{\alpha} f_{\alpha} \nabla p_{\alpha} - \sum_{\alpha} f_{\alpha} \rho_{\alpha} \wp \nabla z \right] + \sum_{\alpha} \frac{1}{\rho_{\alpha}} \left[\phi S_{\alpha} \frac{\partial \rho_{\alpha}}{\partial t} + \bar{u}_{\alpha} \nabla \cdot \rho_{\alpha} \right] &= \sum_{\alpha} \frac{q_{\alpha}}{\rho_{\alpha}}, \\ -\nabla \cdot \bar{k} \lambda [f_w \nabla p_w + f_o \nabla p_o - (f_w \rho_w + f_o \rho_o) \wp \nabla z] &= \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}. \end{aligned} \quad (5.22)$$

Además, se tiene que: $f_w + f_o = 1$ y $p_w = p_o - p_c$, por lo tanto desarrollando la ecuación (5.22):

$$\begin{aligned} -\nabla \cdot \bar{k}\lambda [f_w (\nabla p_o - \nabla p_c) + f_o \nabla p_o - (f_w \rho_w + f_o \rho_o) \wp \nabla z] &= \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}, \\ -\nabla \cdot \bar{k}\lambda [\nabla p_o - f_w \nabla p_c - (f_w \rho_w + f_o \rho_o) \wp \nabla z] &= \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}, \\ -\nabla \cdot \bar{k}\lambda \left[\nabla p_o - f_w \frac{dp_c}{dS_w} \nabla S_w - (f_w \rho_w + f_o \rho_o) \wp \nabla z \right] &= \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}. \end{aligned}$$

Finalmente, se tiene:

$$-\nabla \cdot \left[\bar{k}\lambda \nabla p_o - \bar{k}\lambda_w \frac{dp_c}{dS_w} \nabla S_w - \bar{k} (\lambda_w \rho_w + \lambda_o \rho_o) \wp \nabla z \right] = \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}. \quad (5.23)$$

donde la ecuación (5.23), es la ecuación de presión correspondiente al modelo conceptual planteado.

5.3. Modelo numérico en coordenadas cartesianas

5.3.1. Discretización de la ecuación de saturación

En esta subsección se discretiza paso a paso la ecuación de saturación para el caso tridimensional. Observando la ecuación (5.20), ésta tiene un término temporal, por esta razón se necesita un esquema de discretización adecuado para su aproximación. El esquema utilizado para la discretización temporal se conoce como esquema θ el cual se describe a continuación.

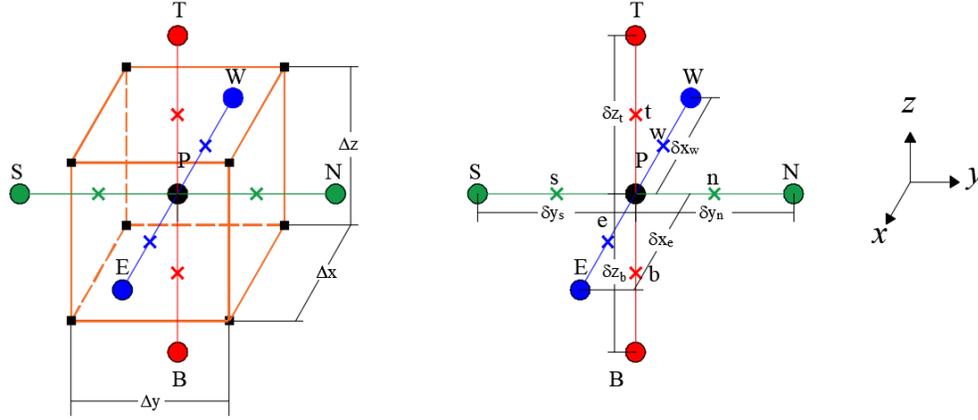
Dada una función $f(\vec{x})$, la integral de n a $n+1$ de dicha función, se aproxima como sigue:

$$\int_n^{n+1} f dt = [\theta f^{n+1} + (1-\theta) f^n] \Delta t, \quad (5.24)$$

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

Integrando la ecuación (5.20), respecto al tiempo y al volumen de control mostrado en la figura 5.1:

$$\int_n^{n+1} \left\{ \int_{\Delta V} \left[\phi \frac{\partial S_w}{\partial t} - \nabla \cdot \left[\bar{k}\lambda_w \left(\nabla p_o - \frac{dp_c}{dS_w} \nabla S_w - \rho_w \wp \nabla z \right) \right] - \frac{q_w}{\rho_w} \right] \Delta V \right\} dt = 0. \quad (5.25)$$


 Figura 5.1: Estencil tridimensional de volumen finito $\Delta V = \Delta x \Delta y \Delta z$

El primer término de la ecuación (5.25), contiene una derivada con respecto al tiempo, por lo tanto las integrales espaciales y temporales, se realizan de una manera muy simple, y el resultado es el siguiente:

$$\phi \int_n^{n+1} \left(\phi \frac{\partial S_w}{\partial t} \right) \Delta V dt = \phi \Delta V \left[\left(S_w \right)^{n+1} - \left(S_w \right)^n \right]. \quad (5.26)$$

Para aproximar los términos restantes de la ecuación (5.25), se utiliza un esquema implícito en el tiempo ($\theta = 1$). Además, el tensor de permeabilidades es considerado un tensor diagonal. Por lo tanto, evaluando los términos restantes, se llega a:

$$\int_n^{n+1} \left\{ \int_{\Delta V} \left[- \frac{\partial}{\partial x} k_{xx} \lambda_w \left(\frac{\partial p_o}{\partial x} - \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial x} - \rho_w \varphi \frac{\partial z}{\partial x} \right) - \frac{\partial}{\partial y} k_{yy} \lambda_w \left(\frac{\partial p_o}{\partial y} - \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial y} - \rho_w \varphi \frac{\partial z}{\partial y} \right) - \frac{\partial}{\partial z} k_{zz} \lambda_w \left(\frac{\partial p_o}{\partial z} - \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial z} - \rho_w \varphi \frac{\partial z}{\partial z} \right) - \frac{q_w}{\rho_w} \right] \Delta V \right\} dt, \quad (5.27)$$

evaluando en el espacio utilizando un esquema centrado para las derivadas:

$$\begin{aligned}
& \int_n^{n+1} \left\{ - \left[\left[k_{xx} \lambda_w \left(\frac{\partial p_o}{\partial x} - \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial x} - \rho_w \wp \frac{\partial z}{\partial x} \right) \right]_w^e \Delta y \Delta z + \left[k_{yy} \lambda_w \left(\frac{\partial p_o}{\partial y} - \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial y} - \rho_w \wp \frac{\partial z}{\partial y} \right) \right]_s^n \Delta x \Delta z \right. \right. \\
& \quad \left. \left. + \left[k_{zz} \lambda_w \left(\frac{\partial p_o}{\partial z} - \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial z} - \rho_w \wp \frac{\partial z}{\partial z} \right) \right]_b^t \Delta x \Delta y \right] - \frac{q_w}{\rho_w} \Delta V \right\} dt = \\
& \int_n^{n+1} \left\{ - \left[k_{xx} \lambda_w \right]_e \left(\frac{p_{o,E} - p_{o,P}}{\delta x_e} \right) - k_{xx} \lambda_w \Big|_w \left(\frac{p_{o,P} - p_{o,W}}{\delta x_w} \right) \right] A_1 + \left[k_{xx} \lambda_w \frac{dp_c}{dS_w} \Big|_e \left(\frac{S_{w,E} - S_{w,P}}{\delta x_e} \right) \right. \\
& \quad - k_{xx} \lambda_w \frac{dp_c}{dS_w} \Big|_w \left(\frac{S_{w,P} - S_{w,W}}{\delta x_w} \right) \right] A_1 + \left[k_{xx} \lambda_w \wp \rho_w \Big|_e \left(\frac{z_{w,E} - z_{w,P}}{\delta x_e} \right) - k_{xx} \lambda_w \wp \rho_w \Big|_w \left(\frac{z_{w,P} - z_{w,W}}{\delta x_w} \right) \right] A_1 \\
& \quad - \left[k_{yy} \lambda_w \Big|_n \left(\frac{p_{o,N} - p_{o,P}}{\delta y_n} \right) - k_{yy} \lambda_w \Big|_s \left(\frac{p_{o,P} - p_{o,S}}{\delta y_s} \right) \right] A_2 + \left[k_{yy} \lambda_w \frac{dp_c}{dS_w} \Big|_n \left(\frac{S_{w,N} - S_{w,P}}{\delta y_n} \right) \right. \\
& \quad - k_{yy} \lambda_w \frac{dp_c}{dS_w} \Big|_s \left(\frac{S_{w,P} - S_{w,S}}{\delta y_s} \right) \right] A_2 + \left[k_{yy} \lambda_w \wp \rho_w \Big|_n \left(\frac{z_{w,N} - z_{w,P}}{\delta y_n} \right) - k_{yy} \lambda_w \wp \rho_w \Big|_s \left(\frac{z_{w,P} - z_{w,S}}{\delta y_s} \right) \right] A_2 \\
& \quad - \left[k_{zz} \lambda_w \Big|_t \left(\frac{p_{o,T} - p_{o,P}}{\delta z_t} \right) - k_{zz} \lambda_w \Big|_b \left(\frac{p_{o,P} - p_{o,B}}{\delta z_b} \right) \right] A_3 + \left[k_{zz} \lambda_w \frac{dp_c}{dS_w} \Big|_t \left(\frac{S_{w,T} - S_{w,P}}{\delta z_t} \right) \right. \\
& \quad - k_{zz} \lambda_w \frac{dp_c}{dS_w} \Big|_b \left(\frac{S_{w,P} - S_{w,B}}{\delta z_b} \right) \right] A_3 + \left[k_{zz} \lambda_w \wp \rho_w \Big|_t \left(\frac{z_{w,T} - z_{w,P}}{\delta z_t} \right) - k_{zz} \lambda_w \wp \rho_w \Big|_b \left(\frac{z_{w,P} - z_{w,B}}{\delta z_b} \right) \right] A_3 \\
& \quad \left. - \frac{q_w}{\rho_w} \Delta V \right\} dt, \tag{5.28}
\end{aligned}$$

evaluando la integral que denota tiempo con un esquema implícito:

$$\begin{aligned}
& \left\{ - \left[k_{xx} \lambda_w \Big|_e \left(\frac{p_{o,E} - p_{o,P}}{\delta x_e} \right) - k_{xx} \lambda_w \Big|_w \left(\frac{p_{o,P} - p_{o,W}}{\delta x_w} \right) \right] A_1 + \left[k_{xx} \lambda_w \frac{dp_c}{dS_w} \Big|_e \left(\frac{S_{w,E} - S_{w,P}}{\delta x_e} \right) \right. \right. \\
& \quad - k_{xx} \lambda_w \frac{dp_c}{dS_w} \Big|_w \left(\frac{S_{w,P} - S_{w,W}}{\delta x_w} \right) \right] A_1 + \left[k_{xx} \lambda_w \wp \rho_w \Big|_e \left(\frac{z_{w,E} - z_{w,P}}{\delta x_e} \right) - k_{xx} \lambda_w \wp \rho_w \Big|_w \left(\frac{z_{w,P} - z_{w,W}}{\delta x_w} \right) \right] A_1 \\
& \quad - \left[k_{yy} \lambda_w \Big|_n \left(\frac{p_{o,N} - p_{o,P}}{\delta y_n} \right) - k_{yy} \lambda_w \Big|_s \left(\frac{p_{o,P} - p_{o,S}}{\delta y_s} \right) \right] A_2 + \left[k_{yy} \lambda_w \frac{dp_c}{dS_w} \Big|_n \left(\frac{S_{w,N} - S_{w,P}}{\delta y_n} \right) \right. \\
& \quad - k_{yy} \lambda_w \frac{dp_c}{dS_w} \Big|_s \left(\frac{S_{w,P} - S_{w,S}}{\delta y_s} \right) \right] A_2 + \left[k_{yy} \lambda_w \wp \rho_w \Big|_n \left(\frac{z_{w,N} - z_{w,P}}{\delta y_n} \right) - k_{yy} \lambda_w \wp \rho_w \Big|_s \left(\frac{z_{w,P} - z_{w,S}}{\delta y_s} \right) \right] A_2 \\
& \quad - \left[k_{zz} \lambda_w \Big|_t \left(\frac{p_{o,T} - p_{o,P}}{\delta z_t} \right) - k_{zz} \lambda_w \Big|_b \left(\frac{p_{o,P} - p_{o,B}}{\delta z_b} \right) \right] A_3 + \left[k_{zz} \lambda_w \frac{dp_c}{dS_w} \Big|_t \left(\frac{S_{w,T} - S_{w,P}}{\delta z_t} \right) \right. \\
& \quad - k_{zz} \lambda_w \frac{dp_c}{dS_w} \Big|_b \left(\frac{S_{w,P} - S_{w,B}}{\delta z_b} \right) \right] A_3 + \left[k_{zz} \lambda_w \wp \rho_w \Big|_t \left(\frac{z_{w,T} - z_{w,P}}{\delta z_t} \right) - k_{zz} \lambda_w \wp \rho_w \Big|_b \left(\frac{z_{w,P} - z_{w,B}}{\delta z_b} \right) \right] A_3 \\
& \quad \left. - \frac{q_w}{\rho_w} \Delta V \right\}^{n+1} \Delta t = \\
& \left\{ - \left[T_{w,e} (p_{o,E} - p_{o,P}) - T_{w,w} (p_{o,P} - p_{o,W}) \right] + \left[T_{w,e} \frac{dp_c}{dS_w} \Big|_e (S_{w,E} - S_{w,P}) \right. \right. \\
& \quad - T_{w,w} \frac{dp_c}{dS_w} \Big|_w (S_{w,P} - S_{w,W}) \left. \right] + \left[T_{w,e} \wp \rho_w \Big|_e (z_{w,E} - z_{w,P}) - T_{w,w} \wp \rho_w \Big|_w (z_{w,P} - z_{w,W}) \right] \\
& \quad - \left[T_{w,n} (p_{o,N} - p_{o,P}) - T_{w,s} (p_{o,P} - p_{o,S}) \right] + \left[T_{w,n} \frac{dp_c}{dS_w} \Big|_n (S_{w,N} - S_{w,P}) \right. \\
& \quad - T_{w,s} \frac{dp_c}{dS_w} \Big|_s (S_{w,P} - S_{w,S}) \left. \right] + \left[T_{w,n} \wp \rho_w \Big|_n (z_{w,N} - z_{w,P}) - T_{w,s} \wp \rho_w \Big|_s (z_{w,P} - z_{w,S}) \right] \\
& \quad - \left[T_{w,t} (p_{o,T} - p_{o,P}) - T_{w,b} (p_{o,P} - p_{o,B}) \right] + \left[T_{w,t} \frac{dp_c}{dS_w} \Big|_t (S_{w,T} - S_{w,P}) \right. \\
& \quad - T_{w,b} \frac{dp_c}{dS_w} \Big|_b (S_{w,P} - S_{w,B}) \left. \right] + \left[T_{w,t} \wp \rho_w \Big|_t (z_{w,T} - z_{w,P}) - T_{w,b} \wp \rho_w \Big|_b (z_{w,P} - z_{w,B}) \right] \\
& \quad \left. - \frac{q_{w,P}}{\rho_w} \Delta V \right\}^{n+1} \Delta t, \tag{5.29}
\end{aligned}$$

donde la transmisibilidad está definida por $T_{w,i} = \frac{k_{jj}\lambda_w A_j}{\delta x_i}$ para $j = 1, 2, 3$ y el subíndice i se refiere a las caras del volumen discreto e, w, n, s, t o b , además el peso específico se define como: $\gamma_w = \rho_w \wp$.

Sumando las ecuaciones (5.26) y (5.29) y reorganizando términos, se obtiene la discretización tridimensional mediante volumen finito de la ecuación de saturación:

$$\begin{aligned}
\frac{\phi \Delta V}{\Delta t} [(S_w) - (S_w)^n]_P = & \\
& \left[T_{w,e} (p_{o,E} - p_{o,P}) - T_{w,w} (p_{o,P} - p_{o,W}) \right] - \left[T_{w,e} \frac{dp_c}{dS_w} \Big|_e (S_{w,E} - S_{w,P}) \right. \\
& \left. - T_{w,w} \frac{dp_c}{dS_w} \Big|_w (S_{w,P} - S_{w,W}) \right] - \left[T_{w,e} \gamma_w \Big|_e (z_{w,E} - z_{w,P}) - T_{w,w} \gamma_w \Big|_w (z_{w,P} - z_{w,W}) \right] \\
& \left[T_{w,n} (p_{o,N} - p_{o,P}) - T_{w,s} (p_{o,P} - p_{o,S}) \right] - \left[T_{w,n} \frac{dp_c}{dS_w} \Big|_n (S_{w,N} - S_{w,P}) \right. \\
& \left. - T_{w,s} \frac{dp_c}{dS_w} \Big|_s (S_{w,P} - S_{w,S}) \right] - \left[T_{w,n} \gamma_w \Big|_n (z_{w,N} - z_{w,P}) - T_{w,s} \gamma_w \Big|_s (z_{w,P} - z_{w,S}) \right] \\
& \left[T_{w,t} (p_{o,T} - p_{o,P}) - T_{w,b} (p_{o,P} - p_{o,B}) \right] - \left[T_{w,t} \frac{dp_c}{dS_w} \Big|_t (S_{w,T} - S_{w,P}) \right. \\
& \left. - T_{w,b} \frac{dp_c}{dS_w} \Big|_b (S_{w,P} - S_{w,B}) \right] - \left[T_{w,t} \gamma_w \Big|_t (z_{w,T} - z_{w,P}) - T_{w,b} \gamma_w \Big|_b (z_{w,P} - z_{w,B}) \right] \\
& - \frac{q_{w,P}}{\rho_w} \Delta V, \tag{5.30}
\end{aligned}$$

donde el subíndice $n + 1$, se omite por simplicidad. Siguiendo la metodología presentada anteriormente para discretizar la ecuación de presión (5.23), se llega a:

$$\begin{aligned}
& \left[T_e (p_{o,E} - p_{o,P}) - T_w (p_{o,P} - p_{o,W}) \right] - \left[T_{w,e} \frac{dp_c}{dS_w} \Big|_e (S_{w,E} - S_{w,P}) - T_{w,w} \frac{dp_c}{dS_w} \Big|_w (S_{w,P} - S_{w,W}) \right] \\
& - \left[\frac{k_{xx}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_1}{\delta x} \Big|_e (z_{w,E} - z_{w,P}) - \frac{k_{xx}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_1}{\delta x} \Big|_w (z_{w,P} - z_{w,W}) \right] \\
& \left[T_{w,n} (p_{o,N} - p_{o,P}) - T_{w,s} (p_{o,P} - p_{o,S}) \right] - \left[T_{w,n} \frac{dp_c}{dS_w} \Big|_n (S_{w,N} - S_{w,P}) - T_{w,s} \frac{dp_c}{dS_w} \Big|_s (S_{w,P} - S_{w,S}) \right] \\
& - \left[\frac{k_{yy}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_2}{\delta y} \Big|_n (z_{w,N} - z_{w,P}) - \frac{k_{yy}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_2}{\delta y} \Big|_s (z_{w,P} - z_{w,S}) \right] \\
& \left[T_{w,t} (p_{o,T} - p_{o,P}) - T_{w,b} (p_{o,P} - p_{o,B}) \right] - \left[T_{w,t} \frac{dp_c}{dS_w} \Big|_t (S_{w,T} - S_{w,P}) - T_{w,b} \frac{dp_c}{dS_w} \Big|_b (S_{w,P} - S_{w,B}) \right] \\
& - \left[\frac{k_{zz}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_3}{\delta z} \Big|_t (z_{w,T} - z_{w,P}) - \frac{k_{zz}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_3}{\delta z} \Big|_b (z_{w,P} - z_{w,B}) \right] \\
& = - \left[\frac{q_w}{\rho_w} + \frac{q_o}{\rho_o} \right] \Delta V, \tag{5.31}
\end{aligned}$$

donde T_i implica el cálculo de la transmisibilidad considerando la movilidad total λ , y $T_{w,i}$ se refiere a la transmisibilidad considerando la movilidad de la fase w .

Se observa que las ecuaciones (5.30) y (5.31) están acopladas y son no lineales, por lo tanto, en la siguiente sección se explica de manera breve el método de Newton-Raphson para la linealización y solución numérica de sistemas de ecuaciones diferenciales parciales.

5.3.2. Método de Newton-Raphson para resolver sistemas de ecuaciones diferenciales parciales

Este apartado está basado en la metodología reportada en el texto [44]. Un sistema general de ecuaciones diferenciales parciales no lineales, se puede escribir como sigue:

$$\mathcal{L}_m\{F_m[\mathbf{p}(\mathbf{x})]\} = f_m(\mathbf{x}), \quad m = 1, 2, \dots, M, \quad x \in \Omega, \quad (5.32)$$

donde \mathcal{L}_m denota un operador lineal diferencial, F_m es una función no lineal, $\mathbf{p} = (p_1, p_2, \dots, p_M)$ es el vector de las variables dependientes, $\mathbf{f} = (f_1, f_2, \dots, f_M)$ es un vector dado y M es el número total de ecuaciones. El método de Newton-Raphson para resolver el sistema (5.32) establece una solución iterativa. Como primer paso para $F_m(\mathbf{p} + \delta\mathbf{p})$, se expande en series de Taylor como sigue:

$$F_m(\mathbf{p} + \delta\mathbf{p}) = F_m(\mathbf{p}) + \nabla F_m(\mathbf{p}) \cdot \delta\mathbf{p} + \mathcal{O}(|\delta\mathbf{p}|^2), \quad (5.33)$$

donde $|\delta\mathbf{p}|^2$ es la norma euclidiana de $\delta\mathbf{p}$. Si se truncan los términos de alto orden $\mathcal{O}(|\delta\mathbf{p}|^2)$ relativos a $\delta\mathbf{p}$, $F_m(\mathbf{p} + \delta\mathbf{p})$, se puede aproximar como:

$$F_m(\mathbf{p} + \delta\mathbf{p}) \approx F_m(\mathbf{p}) + \nabla F_m(\mathbf{p}) \cdot \delta\mathbf{p}. \quad (5.34)$$

Si se sustituye (5.34) en (5.32), se obtiene el siguiente sistema de ecuaciones iterativas:

$$\mathcal{L}_m[F_m\mathbf{p}^k(\mathbf{x}) + \nabla F_m(\mathbf{p}^k) \cdot \delta\mathbf{p}^{k+1}] = f_m(\mathbf{x}), \quad m = 1, 2, \dots, M, \quad x \in \Omega, \quad (5.35)$$

donde \mathbf{p}^k es la k -ésima solución de \mathbf{p} y $\nabla F_m(\mathbf{p}^k)$ es $\nabla F_m(\mathbf{p})$ en $\mathbf{p} = \mathbf{p}^k$, con una solución inicial \mathbf{p}^0 . En el sistema de ecuaciones dado por (5.35), el vector de corrección $\delta\mathbf{p}^{k+1}$ es la incógnita. Este sistema puede escribirse de la siguiente manera:

$$\mathcal{L}_m[\nabla F_m(\mathbf{p}^k) \cdot \delta\mathbf{p}^{k+1}] = g_m(\mathbf{x}), \quad m = 1, 2, \dots, M, \quad x \in \Omega, \quad (5.36)$$

donde $g_m(\mathbf{x}) = f_m(\mathbf{x}) - \mathcal{L}_m[F_m\mathbf{p}^k(\mathbf{x})]$, además $F_m(\mathbf{p}^k)$ y $\nabla F_m(\mathbf{p}^k)$ son fijos. Ahora la ecuación (5.36) es un sistema lineal para $\delta\mathbf{p}^{k+1}$. Note que $\nabla F_m(\mathbf{p}^k)$ es el Jacobiano de la matriz de F_m y g_m es el residual del sistema diferencial original (ecuación (5.32)) en \mathbf{p}^k . Se obtiene una nueva solución para el vector \mathbf{p}^{k+1} sumando el vector de corrección $\delta\mathbf{p}^{k+1}$ a la iteración previa, es decir:

$$\mathbf{p}^{k+1} = \mathbf{p}^k + \delta\mathbf{p}^{k+1}. \quad (5.37)$$

La iteración continúa hasta que el valor de la norma Euclidiana de $\delta\mathbf{p}^{k+1}$ es menor que un valor prescrito.

5.3.3. Solución simultánea de las ecuaciones de presión y saturación

El método de Newton-Rapshon es el más utilizado para la solución del flujo de fluidos en medios porosos. La principal ventaja que proporciona el método es su estabilidad numérica en comparación con los métodos que utilizan una discretización explícita en el tiempo. Su principal desventaja es su alto costo computacional.

Para utilizar el método de Newton-Rapshon, primeramente se define un vector $g_m(\mathbf{x})$ con los residuales de las ecuaciones diferenciales discretas. Escribiendo la ecuación para la presión de la fase aceite (5.31) en forma residual, se obtiene:

$$\begin{aligned}
R_o(p_o, S_w) = & \left[T_e(p_{o,E} - p_{o,P}) - T_w(p_{o,P} - p_{o,W}) \right] - \left[T_{w,e} \frac{dp_c}{dS_w} \Big|_e (S_{w,E} - S_{w,P}) - T_{w,w} \frac{dp_c}{dS_w} \Big|_w (S_{w,P} - S_{w,W}) \right] \\
& - \left[\frac{k_{xx}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_1}{\delta x} \Big|_e (z_{w,E} - z_{w,P}) - \frac{k_{xx}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_1}{\delta x} \Big|_w (z_{w,P} - z_{w,W}) \right] \\
& \left[T_{w,n}(p_{o,N} - p_{o,P}) - T_{w,s}(p_{o,P} - p_{o,S}) \right] - \left[T_{w,n} \frac{dp_c}{dS_w} \Big|_n (S_{w,N} - S_{w,P}) - T_{w,s} \frac{dp_c}{dS_w} \Big|_s (S_{w,P} - S_{w,S}) \right] \\
& - \left[\frac{k_{yy}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_2}{\delta y} \Big|_n (z_{w,N} - z_{w,P}) - \frac{k_{yy}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_2}{\delta y} \Big|_s (z_{w,P} - z_{w,S}) \right] \\
& \left[T_{w,t}(p_{o,T} - p_{o,P}) - T_{w,b}(p_{o,P} - p_{o,B}) \right] - \left[T_{w,t} \frac{dp_c}{dS_w} \Big|_t (S_{w,T} - S_{w,P}) - T_{w,b} \frac{dp_c}{dS_w} \Big|_b (S_{w,P} - S_{w,B}) \right] \\
& - \left[\frac{k_{zz}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_3}{\delta z} \Big|_t (z_{w,T} - z_{w,P}) - \frac{k_{zz}(\lambda_w \gamma_w + \lambda_o \gamma_o) A_3}{\delta z} \Big|_b (z_{w,P} - z_{w,B}) \right] \\
& + \left[\frac{q_w}{\rho_w} + \frac{q_o}{\rho_o} \right] \Delta V.
\end{aligned} \tag{5.38}$$

El residual para la ecuación de saturación de la fase agua, se escribe como:

$$\begin{aligned}
R_w(p_o, S_w) = & \frac{\phi \Delta V}{\Delta t} [(S_w) - (S_w)^n]_P \\
& - \left[T_{w,e}(p_{o,E} - p_{o,P}) - T_{w,w}(p_{o,P} - p_{o,W}) \right] + \left[T_{w,e} \frac{dp_c}{dS_w} \Big|_e (S_{w,E} - S_{w,P}) \right. \\
& \left. - T_{w,w} \frac{dp_c}{dS_w} \Big|_w (S_{w,P} - S_{w,W}) \right] + \left[T_{w,e} \wp \rho_w \Big|_e (z_{w,E} - z_{w,P}) - T_{w,w} \wp \rho_w \Big|_w (z_{w,P} - z_{w,W}) \right] \\
& - \left[T_{w,n}(p_{o,N} - p_{o,P}) - T_{w,s}(p_{o,P} - p_{o,S}) \right] + \left[T_{w,n} \frac{dp_c}{dS_w} \Big|_n (S_{w,N} - S_{w,P}) \right. \\
& \left. - T_{w,s} \frac{dp_c}{dS_w} \Big|_s (S_{w,P} - S_{w,S}) \right] + \left[T_{w,n} \wp \rho_w \Big|_n (z_{w,N} - z_{w,P}) - T_{w,s} \wp \rho_w \Big|_s (z_{w,P} - z_{w,S}) \right] \\
& - \left[T_{w,t}(p_{o,T} - p_{o,P}) - T_{w,b}(p_{o,P} - p_{o,B}) \right] + \left[T_{w,t} \frac{dp_c}{dS_w} \Big|_t (S_{w,T} - S_{w,P}) \right. \\
& \left. - T_{w,b} \frac{dp_c}{dS_w} \Big|_b (S_{w,P} - S_{w,B}) \right] + \left[T_{w,t} \gamma_w \Big|_t (z_{w,T} - z_{w,P}) - T_{w,b} \gamma_w \Big|_b (z_{w,P} - z_{w,B}) \right] \\
& - \frac{q_{w,P}}{\rho_w} \Delta V.
\end{aligned} \tag{5.39}$$

Como siguiente paso se deben seleccionar las variables primarias que se resolverán mediante la iteración de Newton-Rapshon. Se seleccionan las variables p_o^{n+1} y S_w^{n+1} como las

variables primarias, las cuales serán también las incógnitas, entonces el sistema a resolver tiene la siguiente forma:

$$\begin{pmatrix} \frac{\partial R_o^k}{\partial p_o} & \frac{\partial R_o^k}{\partial S_w} \\ \frac{\partial R_w^k}{\partial p_o} & \frac{\partial R_w^k}{\partial S_w} \end{pmatrix} \begin{pmatrix} \delta p_o^{k+1} \\ \delta S_w^{k+1} \end{pmatrix} = \begin{pmatrix} -R_o^k \\ -R_w^k \end{pmatrix}, \quad (5.40)$$

donde las derivadas del Jacobiano se tienen que calcular en el dominio discreto. Con estas consideraciones y llevando a cabo las operaciones matemáticas correspondientes, las componentes del Jacobiano para el residual R_o para el caso tridimensional, se expresan como sigue:

$$\begin{aligned} & \frac{\partial R_o(p_o, S_w)}{\partial P_{o,B}} \Big|_k \delta p_{o,T}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial P_{o,S}} \Big|_k \delta p_{o,S}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial p_{o,W}} \Big|_k \delta p_{o,W}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial p_{o,P}} \Big|_k \delta p_{o,P}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial p_{o,E}} \Big|_k \delta p_{o,E}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w)}{\partial p_{o,N}} \Big|_k \delta p_{o,N}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial p_{o,T}} \Big|_k \delta p_{o,T}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial S_{w,B}} \Big|_k \delta S_{w,T}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial S_{w,S}} \Big|_k \delta S_{w,S}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial S_{w,W}} \Big|_k \delta S_{w,W}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w)}{\partial S_{w,P}} \Big|_k \delta S_{w,P}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial S_{w,E}} \Big|_k \delta S_{w,E}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial S_{w,N}} \Big|_k \delta S_{w,N}^{k+1} + \frac{\partial R_o(p_o, S_w)}{\partial S_{w,T}} \Big|_k \delta S_{w,T}^{k+1} = -R_o, \end{aligned} \quad (5.41)$$

aquí las derivadas se evalúan numéricamente mediante diferencias finitas de primer orden. Para las componentes del Jacobiano correspondientes a la ecuación de saturación, se obtiene:

$$\begin{aligned} & \frac{\partial R_w(p_o, S_w)}{\partial P_{o,B}} \Big|_k \delta p_{o,T}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial P_{o,S}} \Big|_k \delta p_{o,S}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial p_{o,W}} \Big|_k \delta p_{o,W}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial p_{o,P}} \Big|_k \delta p_{o,P}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial p_{o,E}} \Big|_k \delta p_{o,E}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w)}{\partial p_{o,N}} \Big|_k \delta p_{o,N}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial p_{o,T}} \Big|_k \delta p_{o,T}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial S_{w,B}} \Big|_k \delta S_{w,T}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial S_{w,S}} \Big|_k \delta S_{w,S}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial S_{w,W}} \Big|_k \delta S_{w,W}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w)}{\partial S_{w,P}} \Big|_k \delta S_{w,P}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial S_{w,E}} \Big|_k \delta S_{w,E}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial S_{w,N}} \Big|_k \delta S_{w,N}^{k+1} + \frac{\partial R_w(p_o, S_w)}{\partial S_{w,T}} \Big|_k \delta S_{w,T}^{k+1} = -R_w. \end{aligned} \quad (5.42)$$

Las ecuaciones (5.41), (5.42) junto con los residuales (5.38) y (5.39), se utilizan para construir el sistema lineal a resolver, el cual se muestra en la figura 5.2, donde el subíndice l se refiere al total de volúmenes discretos.

$$\begin{bmatrix} \frac{\partial R_o^k}{\partial p_o} \\ \frac{\partial R_o^k}{\partial S_w} \\ \frac{\partial R_w^k}{\partial p_o} \\ \frac{\partial R_w^k}{\partial S_w} \end{bmatrix} \begin{bmatrix} \delta p_o^{k+1} \\ \delta S_w^{k+1} \end{bmatrix} = \begin{bmatrix} -R_o^k \\ -R_w^k \end{bmatrix}$$

Figura 5.2: Sistema de ecuaciones lineales para las variables primarias p_o y S_w .

5.3.4. Implementación computacional

Esta sección describe la metodología computacional para implementar la solución del flujo de fluidos bifásico e incompresible en medios porosos.

Algoritmo general e implementación en la CPU

Con el objeto de clarificar la implementación computacional, en este apartado se proporciona una breve descripción del código computacional desarrollado, tomando en cuenta el pseudocódigo mostrado en la figura 5.3. Cabe aclarar que se utilizó *C++* como lenguaje de programación, las bibliotecas *Eigen* [48] y *Cusp 5.1* [49] para llevar a cabo la solución del sistema de ecuaciones lineales en la CPU y la GPU, respectivamente, además para generar las gráficas de los resultados obtenidos se utilizó *Python 2.7* con la biblioteca *Matplotlib* [50].

```

01 /* Inicialización de todas las variables y datos iniciales,
02 definición de todos los arreglos y matrices
03 para construir el sistema de ecuaciones lineales */
04 WHILE (Time_step <= Total_time) DO
05     WHILE ( Delta_Sw < epsilon & Iteration < Max_iter) DO
06         Calcular las componentes del Jacobiano referentes a la fase aceite y Ro
07         Calcular las componentes del Jacobiano referentes a la fase agua y Rw
08         Construir el Jacobiano usando el formato CRS
09         Resolver el sistema de ecuaciones utilizando el método BICGSTAB
10     END WHILE
11     Actualizar las variables primarias
12     IF (!(Time_step / Frequency)) THEN
13         Salvar o imprimir las variables primarias (po y Sw)
14     END IF
15 END WHILE
16 /* Liberar memoria y finalizar la simulación */

```

Figura 5.3: Pseudocódigo general

En las primeras tres líneas del pseudocódigo mostrado en la figura (5.3), se declaran e inicializan todas las variables y arreglos requeridos con los valores adecuados, esto incluye las condiciones iniciales y de frontera, valores petrofísicos, tamaño de la malla, paso de tiempo, etc. En la línea 4 se inician los cálculos numéricos y se llevan a cabo dentro del ciclo *while* hasta que se alcanza el número total de pasos de tiempo definido por el usuario. Dentro de este primer ciclo, se tiene otro ciclo anidado que implementa el algoritmo de Newton-Raphson (NR). Este ciclo anidado comienza en la línea 5 y se lleva a cabo hasta que la norma del cambio de la saturación de agua ($|\delta S_w|$) sea menor que un valor prescrito (ϵ) o se alcance el número máximo de iteraciones prescrito del algoritmo NR. Las líneas 6 a 9 representan los pasos principales para resolver el problema y en ellos se consume el mayor porcentaje de tiempo de CPU. En las líneas 6 y 7 se calcula el Jacobiano, esto significa calcular las componentes discretizadas de los residuos, las ecuaciones (5.38)-(5.42) y sus correspondientes derivadas (5.41) y (5.42). Vale la pena mencionar que todas las derivadas se realizan numéricamente utilizando diferencias finitas de primer orden. Luego, en el paso 9, el Jacobiano

se genera utilizando el formato de almacenamiento de filas comprimidas CRS (*Compressed Row Storage*) para el ahorro de memoria. En los cálculos realizados, estos tres pasos toman alrededor del 8% del tiempo de cómputo total. En la línea 9 se resuelve el sistema lineal de ecuaciones utilizando el método del gradiente biconjugado estabilizado (*BICGSTAB*, por sus siglas en inglés) el cual está contenido en la biblioteca EIGEN. Este paso lleva alrededor del 75% del tiempo de cómputo total de la CPU. Una vez que el algoritmo NR ha llegado a la convergencia, la línea 11 indica la actualización de todas las variables primarias para pasar al siguiente paso de tiempo. Finalmente, se guardan los resultados (variables primarias) cada vez que la variable `iteration` es divisible por la variable `Frecuency` prescrita. Esta última variable es importante para la implementación en CUDA para la GPU.

Implementación del código numérico en la GPU

Para la implementación en la GPU se utiliza CUDA y la biblioteca CUSP [49], esta biblioteca proporciona una interfaz de alto nivel para manipular matrices dispersas y resolver sistemas lineales dispersos. El Apéndice C contiene un ejemplo de programación utilizando CUDA, además se muestra como definir y utilizar las funciones *kernel*. Nuestra implementación numérica está casi totalmente realizada en la GPU, lo que significa que la cantidad de intercambio de información entre la CPU y la GPU es relativamente baja. En este sentido, el pseudocódigo del algoritmo principal para la implementación de la GPU es similar al que se muestra en la figura 5.3. Se necesitan recalcar las principales diferencias: a) el ciclo que comienza en la línea 4 requiere todas las variables y arreglos definidos en las líneas 1-3, por lo tanto, un primer intercambio de información se realiza desde la CPU a la GPU antes de iniciar el ciclo de solución, sin embargo, esta inicialización lleva un tiempo de cómputo mínimo debido al hecho de que las matrices o arreglos más grandes se construyen directamente en la GPU; b) las operaciones en las líneas 6, 7, 8, 9 y 11 están todas codificadas en CUDA, por lo tanto, se tienen varias funciones *kernel* que se ejecutan en la GPU; c) la operación de la línea 13 requiere una transferencia de información de la GPU a la CPU; cabe señalar que esto se hace cada vez que la variable `iteration` es divisible por la variable `Frecuency` y puede ser solamente una vez, por ejemplo, cuando la simulación finaliza o cuando el usuario requiere la información de la solución final.

Las funciones kernel se ejecutan en paralelo por hilos o *threads*. Estos hilos están definidos por índices globales que pertenecen a la malla de bloques. Los tamaños de mallas y bloque son definidos por el usuario. Esta malla se puede definir para 1, 2 o 3 dimensiones. Cada bloque tiene un número finito de hilos, generalmente hasta 1024 hilos. El tamaño máximo de malla viene dado por las especificaciones de fabricación de cada tarjeta gráfica. Todas estas características de las GPU modernas se pueden consultar en el sitio web de NVIDIA. Teniendo todo esto en cuenta, solo es posible paralelizar eficientemente códigos numéricos programados para no exceder el máximo número de hilos que se pueden ejecutar en la malla de bloques. Por otro lado, es fácil paralelizar funciones que ejecutan las mismas operaciones sobre las entradas de matrices, ya que solo los índices de los subprocesos deben definirse y esta definición se reemplaza en cada ciclo. Como ejemplo de este método, las figuras 5.4 y

5.5 muestran un extracto de los códigos para calcular el bloque jacobiano correspondiente al residual R_o y sus derivadas ($\partial R_o/\partial p_o$), en la CPU y la GPU, respectivamente.

```

/* ... Definición de la función que calcula un bloque del Jacobiano en C++...*/
01 void jacobianCoeff_RoPo3D (*parámetros físicos como argumentos) {
02   for (i=1; i<nx-1; i++){
03     for (j=1; j<ny-1; j++){
04       for (k=1; k<nz-1; k++){
05         interpolatePermeability(k, k_e, k_w, k_n, k_s, k_t, k_b);
06         interpolateSaturation(Sw, Sw_e, Sw_w, Sw_n, Sw_s, Sw_t, Sw_b);
07         calculate_kro(Sw, kro(Sw),kro_e, kro_w, kro_n, kro_s, kro_t, kro_b);
08         /* ... Algunos otros cálculos para poder calcular F(Po)... */
09         F(Po)=Tr_e*(Po);
10         F(Po+deltaPo)=Tr_e*(Po+deltaPo);
11         Block_RoPo3D[3D(i,j,k)]=firstOrderDerivative(F(Po), F(Po+deltaPo), deltaPo);
12       }
13     }
14   }
15 }

```

Figura 5.4: Función para calcular el bloque del Jacobiano $\partial R_o/\partial p_o$ en la CPU

En la función `jacobianCoeff_RoPo3D ()` se calculan todos los coeficientes del bloque del Jacobiano $\partial R_o/\partial p_o$. El código es estándar y se basa en matrices tridimensionales que contienen algunas variables relacionadas con las coordenadas cartesianas para la simulación numérica. Por lo tanto, tenemos tres ciclos anidados, un ciclo para cada eje. Dentro de estos ciclos se ejecutan varias funciones que se encargan de llevar a cabo varios métodos numéricos, entre ellos: interpolaciones para la permeabilidad absoluta y las saturaciones desde los centros de los volúmenes a sus caras (líneas 5 y 6), así como el cálculo de la permeabilidad relativa usando datos de tablas (línea 7), también el cálculo de coeficientes del residual y sus derivadas (líneas 9 y 10), y el ensamblaje del bloque correspondiente (línea 11).

```

/* ... Definición de la función que calcula un bloque del Jacobiano en CUDA... */
01 __global__ void jacobianCoeff_RoPo3D (*parámetros físicos como argumentos){
02   int i = threadIdx.x + blockIdx.x*blockDim.x;
03   int j = threadIdx.y + blockIdx.y*blockDim.y;
04   int k = threadIdx.z + blockIdx.z*blockDim.z;
05   if(i>0 && i < nx-1 && j > 0 && j < ny-1 && k > 0 && k < nz-1){
06     interpolatePermeability(k, k_e, k_w, k_n, k_s, k_t, k_b);
07     interpolateSaturation(Sw, Sw_e, Sw_w, Sw_n, Sw_s, Sw_t, Sw_b);
08     calculate_kro(Sw, kro(Sw),kro_e, kro_w, kro_n, kro_s, kro_t, kro_b);
09     /* ... Algunos otros cálculos para poder calcular F(Po)... */
10     F(Po)=Tr_e*(Po);
11     F(Po+deltaPo)=Tr_e*(Po+deltaPo);
12     Block_RoPo3D[3D(i,j,k)]=firstOrderDerivative(F(Po), F(Po+deltaPo), deltaPo);
13   }
14 }

```

Figura 5.5: Función para calcular el bloque del Jacobiano $\partial R_o/\partial p_o$ en la GPU

De la misma manera que en la figura 5.4, en la figura 5.5 se muestra un extracto de código de la función kernel en la cual se implementa el cálculo del bloque del jacobiano $\partial R_o / \partial p_o$. La función kernel `jacobianCoeff_RoPo3D ()` se ejecuta completamente en la GPU. Lo primero que se debe hacer es determinar los índices de los hilos de ejecución (véase las líneas 2-4). Usando estos índices es posible que cada hilo del bloque de la malla, ejecute las operaciones definidas en las líneas 5-12 al mismo tiempo. La línea 5 es necesaria para garantizar que los índices se encuentren dentro de los límites de los arreglos. Las líneas 6 a 12 consisten de funciones kernel, similares a las funciones definidas en la CPU, pero usando el índice para realizar cada uno de los métodos numéricos necesarios para calcular el bloque correspondiente del Jacobiano. Para ejecutar la función kernel `jacobianCoeff_RoPo3D ()` se utilizan dos tipos de memoria: 1.- Memoria Global para almacenar las variables primarias (S_w y p_o), las propiedades de los medios porosos (k , ϕ , etc.) y algunos otros arreglos o matrices importantes para llevar a cabo la simulación; 2.- La Memoria Constante se utiliza para almacenar valores constantes, es decir, las viscosidades, los factores de conversión y algunas tablas de propiedades (permeabilidad relativa). La figura 5.6 muestra esquemáticamente cómo se almacenan las variables dentro de la memoria de la GPU.

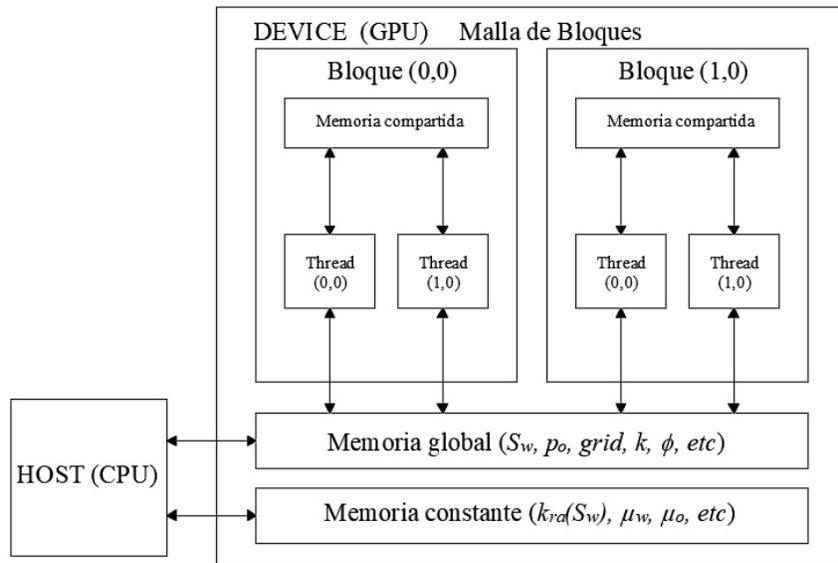


Figura 5.6: Uso de la memoria de la GPU para el almacenamiento de variables (Ilustración adaptada de [28].)

Una vez construido el Jacobiano se procede a la solución del sistema de ecuaciones lineales. Para evitar copias intermedias de GPU a CPU y viceversa, el Jacobiano es envuelto (*wrapped*) utilizando apuntadores mediante la biblioteca *Thrust*. La figura 5.7 muestra un ejemplo de este procedimiento.

```

01 thrust::device_ptr<int>   wrapped_device_I(device_I);
02 thrust::device_ptr<int>   wrapped_device_J(device_J);
03 thrust::device_ptr<double> wrapped_A(A);
04 thrust::device_ptr<double> wrapped_X(X);
05 thrust::device_ptr<double> wrapped_B(B);
06 typedef typename cusp::array1d_view< thrust::device_ptr<int>   > DeviceIndexArrayView;
07 typedef typename cusp::array1d_view< thrust::device_ptr<double> > DeviceValueArrayView;
08 DeviceIndexArrayView row_indices  (wrapped_device_I, wrapped_device_I + elementsNonZero);
09 DeviceIndexArrayView column_indices(wrapped_device_J, wrapped_device_J + elementsNonZero);
10 DeviceValueArrayView values      (wrapped_A, wrapped_A + elementsMass);
11 DeviceValueArrayView X_d         (wrapped_X, wrapped_X + 2*sizeNodes);
12 DeviceValueArrayView B_d         (wrapped_B, wrapped_B + 2*sizeNodes);
13 /* Definir y combinar los arreglos unidimensionales para construir una matriz en formato CRS */
14 typedef cusp::crs_matrix_view<DeviceIndexArrayView, DeviceIndexArrayView, DeviceValueArrayView> DeviceView;
15 DeviceView A_d (2*sizeNodes, 2*sizeNodes, elementsNonZero, row_ptr, column_indices, values);

```

Figura 5.7: Procedimiento para envolver el Jacobiano utilizando la biblioteca Thrust.

En las primeras 2 líneas de la figura 5.7 se definen los apuntadores que “envolverán” los índices I, J correspondientes al tamaño del Jacobiano, las líneas 3 a 5 se utilizan para declarar los apuntadores que envuelven al Jacobiano (matriz de coeficientes A), al vector incógnita X y al vector solución B. En las líneas 6 y 7 se definen objetos de arreglos unidimensionales enteros y de punto flotante, los cuales se utilizan en las líneas 8 a 12 para envolver los índices, los vectores y los valores de la matriz de coeficientes del sistema de ecuaciones lineales. En la línea 14 se define el objeto en el cual se almacenará el sistema de ecuaciones lineales. En la línea 15 se utilizan los datos antes envueltos para construir el sistema de ecuaciones lineales. Una vez construido el sistema se procede a resolver el sistema de ecuaciones por medio del método BICGSTAB, el cual esta contenido en la biblioteca CUSP.

5.4. Validación del código de inyección de agua

5.4.1. 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 (véase referencias [43] y [51]). En el modelo conceptual de Buckley-Leverett se considera lo siguiente:

1. El desplazamiento ocurre en un medio unidimensional (Ver figura 5.8).
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.

Tabla 5.1: Parámetros para la simulación del problema de Buckley-Leverett.

Propiedad	Valor
Longitud del dominio (L_x)	1,000 (ft)
Permeabilidad absoluta (k)	100.0 (mD)
Porosidad (ϕ)	0.20
Viscosidad del agua (μ_w)	0.42 (cP)
Viscosidad del aceite (μ_o)	15.5 (cP)
Saturación residual del agua (S_{wr})	0.40
Saturación residual del aceite (S_{or})	0.18
Velocidad de inyección (u_w^{inj})	2.0E-06 (ft/s)
Presión de producción (p_o^{out})	1,000 (psi)

(Datos adaptados de [43].)

Tabla 5.2: Permeabilidades relativas k_{rw} y k_{ro} .

S_w	0.4	0.42	0.44	0.46	0.48	0.5	0.52	0.54	0.56	0.58	0.6
k_{rw}	0.0	0.0005	0.0020	0.0045	0.008	0.0125	0.018	0.0245	0.032	0.0405	0.05
k_{ro}	1.0	0.9025	0.81	0.7225	0.64	0.5625	0.49	0.4225	0.36	0.3025	0.25
S_w	0.62	0.64	0.66	0.68	0.7	0.72	0.74	0.76	0.78	0.8	0.82
k_{rw}	0.0605	0.072	0.0845	0.098	0.1125	0.128	0.1445	0.162	0.1805	0.20	0.45
k_{ro}	0.2025	0.16	0.1225	0.09	0.0625	0.04	0.0225	0.01	0.0025	0.0	0.0

(Datos tomados de [43].)

La solución analítica para la posición del frente de saturación de agua S_w del problema de Buckley-Leverett, está dada por:

$$x(S_w, t) = \frac{G_w(S_w)}{\phi A} V(t), \quad (5.45)$$

donde la función de distribución $G(S_w)$, se define como la derivada del flujo fraccional de la fase agua $f_w(S_w)$ respecto a S_w :

$$F_w(S_w) = \frac{df_w(S_w)}{dS_w}. \quad (5.46)$$

En este trabajo se calculó la función de distribución utilizando la tabla 5.2 mediante derivadas numéricas con incremento para la saturación de agua de 1×10^{-7} . Una vez conocida la función de distribución, mediante la ecuación (5.45), se puede calcular la posición del frente de saturación S_w para cualquier tiempo t .

Después de obtener la solución analítica se procedió a realizar la solución numérica mediante el código desarrollado para diferentes tiempos de simulación. Cabe aclarar que para calcular las derivadas dentro de los bloques del Jacobiano, se utilizaron incrementos de 1×10^{-5} para la saturación, y 1×10^{-3} para la presión. El criterio de paro para salir del ciclo

de Newton-Rapshon, se seleccionó igual a: $|\delta S_w| < 1 \times 10^{-5}$. Estos parámetros numéricos se determinaron empíricamente por prueba y error, ya que con éstos, se obtuvo buena estabilidad al ejecutar el código con los diferentes tamaños de malla utilizados para este problema.

Las figuras 5.10, 5.11 y 5.12, muestran los perfiles de saturación de agua (S_w), obtenidos para diferentes tamaños de malla y 120 días de simulación. Los resultados se reportan cada 30 días y son congruentes con la solución analítica. En la figura 5.10 se utilizó un tamaño de malla de 100 volúmenes, en ella se puede apreciar que la solución numérica es cualitativamente similar a la solución analítica, sin embargo tiene un error cuantitativamente significativo. Para este tamaño de malla, la desviación cuadrática media (*Root Mean Square Deviation*, RMSD) calculada en todo el dominio es de 0.01875, tomando en cuenta el último tiempo de simulación. Por otra parte, si se considera el error absoluto donde el frente de saturación cae abruptamente, se tienen errores en las curvas de hasta el 20 %.

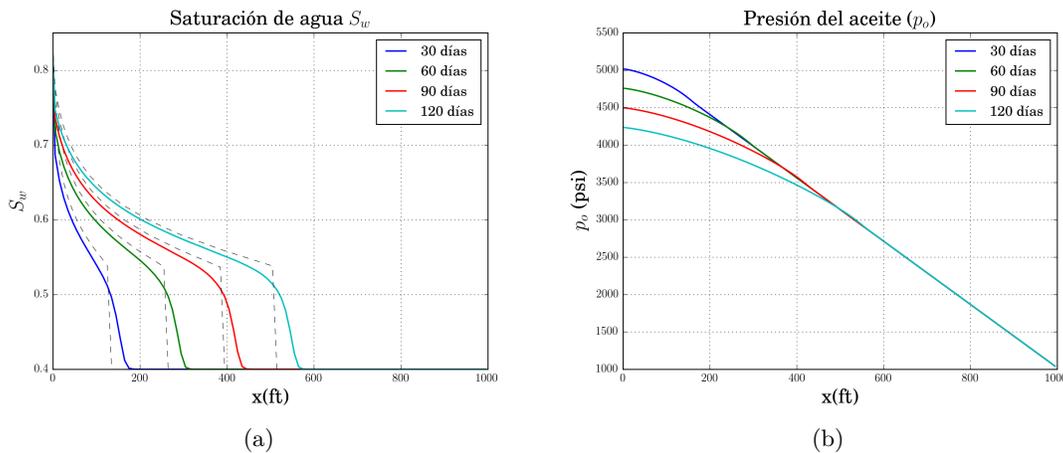


Figura 5.10: a) Perfiles de saturación S_w y b) perfiles de presión p_o , para un tamaño de malla de 100 volúmenes.

La figura 5.11, muestra los resultados obtenidos para un tamaño de malla de 1,000 volúmenes. En ésta, se nota que la solución numérica es aun más semejante a la solución analítica, cualitativa y cuantitativamente. La RMSD obtenida es de 0.00962, si la comparamos con el valor obtenido en la simulación de 100 volúmenes, ésta disminuyó aproximadamente 50 %, lo cual significa que esta simulación numérica se aproxima mejor a la solución analítica.

En la figura 5.12, se muestran los resultados utilizando un tamaño de malla de 10,000 volúmenes. Para este caso, la RMSD obtenida tiene un valor de 0.00827. Si se comparan los valores de la media cuadrática obtenidos para 10,000 y para 1,000 volúmenes, se tiene que los valores son muy similares; esto significa que aun cuando el error disminuye respecto a la solución analítica si se utilizan más nodos, se tiene una mejoría apenas perceptible utilizando más de 1,000 volúmenes para este problema en particular.

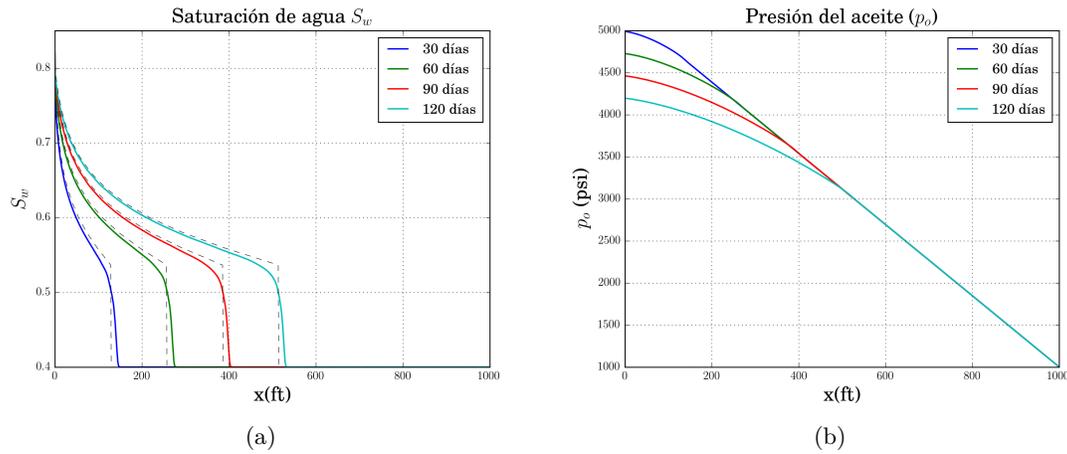


Figura 5.11: a) Perfiles de saturación S_w y b) perfiles de presión p_o , para un tamaño de malla de 1,000 volúmenes.

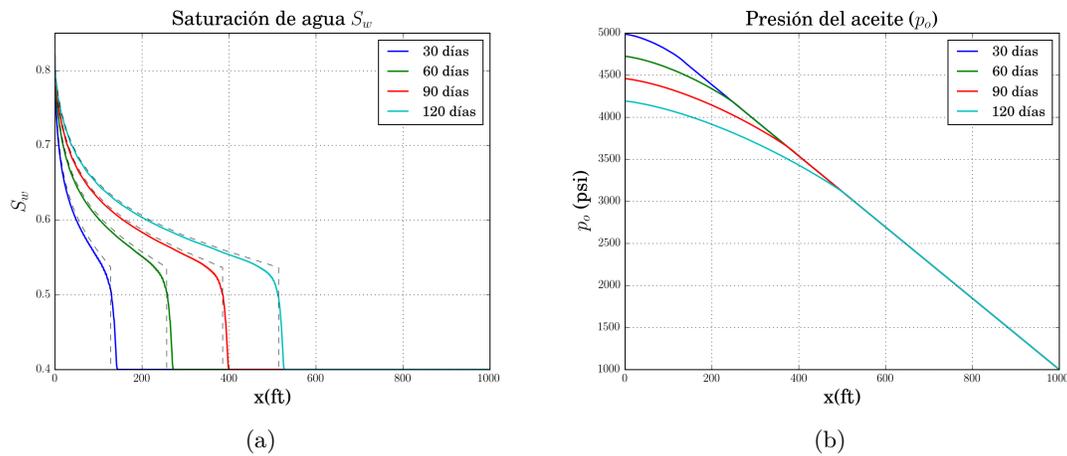


Figura 5.12: a) Perfiles de saturación S_w y b) perfiles de presión p_o , para un tamaño de malla de 10,000 volúmenes.

Para complementar la información de las curvas mostradas, en la tabla 5.3 se proporciona el error absoluto obtenido, donde el perfil de saturación S_w cae abruptamente según su posición, también se muestra la RMSD calculada para todo el dominio en diferentes tiempos de simulación y los distintos tamaños de malla utilizados.

Tabla 5.3: Valores obtenidos de S_w para diferentes tiempos de simulación.

Número de volúmenes discretos	Posición x (ft)	Tiempo de simulación (días)	S_w Numérica	S_w Analítica	Error %	RMSD En todo el dominio
100	145	30	0.4721	0.4	18.033	0.01510
	275	60	0.4792	0.4	19.809	0.01590
	405	90	0.4801	0.4	20.037	0.01598
	535	120	0.4786	0.4	19.666	0.01875
500	145	30	0.42238	0.4	5.5942	0.01127
	275	60	0.42049	0.4	5.1216	0.01095
	405	90	0.40972	0.4	2.4295	0.01104
	535	120	0.40849	0.4	2.1213	0.01093
1,000	145	30	0.40784	0.4	1.9593	0.00987
	275	60	0.40399	0.4	0.99737	0.01027
	405	90	0.40080	0.4	0.19905	0.00981
	535	120	0.40056	0.4	0.13941	0.00962
5,000	145	30	0.40028	0.4	0.07096	0.00944
	275	60	0.39998	0.4	0.00525	0.00942
	405	90	0.39999	0.4	0.00170	0.00897
	535	120	0.40001	0.4	0.00150	0.00839
10,000	145	30	0.400130	0.4	0.03159	0.00916
	275	60	0.400020	0.4	0.00568	0.00917
	405	90	0.400001	0.4	0.00093	0.00881
	535	120	0.400001	0.4	0.00091	0.00827

Los resultados mostrados para este problema indican que el problema de Buckley Leverett fue bien implementado, por lo que podrían utilizarse como punto de referencia para futuras implementaciones numéricas.

5.4.2. Problema de los 5 pozos

El modelo conocido como de los 5 pozos o "*Five Spot*", describe el desplazamiento de aceite por agua en un dominio isotrópico, en el cual se coloca un pozo productor y un pozo inyector en las esquinas opuestas del dominio (ver figura 5.13). Este modo de colocar los pozos emula la simetría de colocar 1 pozo inyector y 4 pozos productores equidistantes al pozo inyector. En el modelo conceptual, se toman en cuenta las siguientes consideraciones:

1. El desplazamiento ocurre en un medio bidimensional.
2. El medio es isotrópico.
3. Se consideran los efectos de la presión capilar.
4. El efecto de la fuerza de gravedad es despreciable.
5. Existe una fuente (pozo inyector) y un sumidero (pozo productor).
6. Los fluidos son incompresibles e inmiscibles.

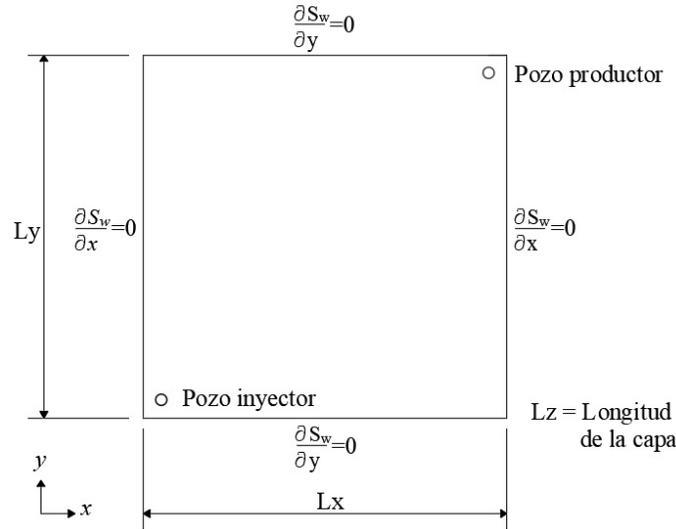


Figura 5.13: Dominio físico para el problema "*Five Spot*".

Tomando en cuenta las consideraciones del modelo conceptual descrito, las ecuaciones (5.20) y (5.23), se reducen a:

$$-\nabla \cdot \left[\bar{k}\lambda \nabla p_o - \bar{k}\lambda_w \frac{dp_c}{dS_w} \nabla S_w \right] = \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}, \quad (5.47)$$

$$\phi \frac{\partial S_w}{\partial t} - \nabla \cdot \left[\bar{k}\lambda_w \left(\nabla p_o - \frac{dp_c}{dS_w} \nabla S_w \right) \right] = \frac{q_w}{\rho_w}. \quad (5.48)$$

Utilizando la metodología numérica descrita en la sección anterior para discretizar, linealizar y resolver el flujo de fluidos bifásico en medios porosos, el sistema lineal de ecuaciones

resultante tiene una matriz Jacobiana formada por bloques pentadiagonales (véase figura 5.14).

Se seleccionó el problema reportado por Chen [43], para verificar la implementación del código numérico bidimensional. Los parámetros para llevar a cabo la simulación numérica se muestran en la tabla 5.4.

Figura 5.14: Sistema de ecuaciones lineales para el problema "Five Spot".

Tabla 5.4: Parámetros para la simulación del problema de los 5 pozos.

Propiedad	Valor
Tamaño del dominio $L_x \times L_y \times L_z$	1,000x1,000x100 (ft)
Permeabilidad absoluta (k)	100.0 (mD)
Porosidad (ϕ)	0.20
Viscosidad del agua (μ_w)	0.096 (cP)
Viscosidad del aceite (μ_o)	1.14 (cP)
Saturación residual del agua (S_{wr})	0.22
Saturación residual del aceite (S_{or})	0.20
Posición del pozo inyector (x_{inj}, y_{inj})	50 , 50 (ft)
Posición del pozo productor (x_{prod}, y_{prod})	950 , 950 (ft)
Presión de inyección en el fondo pozo (p_{wb}^{inj})	3,700 (psi)
Presión de producción en el fondo del pozo (p_{wb}^{out})	3,500 (psi)

(Datos tomados de [43].)

Las permeabilidades relativas, están definidas por [43]:

$$k_{rw} = k_{rwmax} \left(\frac{S_w - S_{wr}}{1 - S_{or} - S_{wr}} \right)^2, \tag{5.49}$$

$$k_{ro} = \left(\frac{S_o - S_{or}}{1 - S_{or} - S_{wr}} \right)^2, \tag{5.50}$$

donde: $k_{rwmax} = 0.65$, representa el valor máximo de la permeabilidad relativa de la fase agua, $S_{wr} = 0.22$, se refiere al valor de la saturación de agua residual y $S_{or} = 0.2$, 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 [43]:

$$p_{cow}(S_w) = p_{cow,min} - \bar{B} \ln \left(\frac{S_w - S_{wr}}{1 - S_{wr}} \right), \quad (5.51)$$

donde: \bar{B} es un valor constante igual a -10.1335 tal que $p_{cow} = p_{cow,max}$ cuando $S_w = S_{wr}$, $p_{cow,min} = 0$ y $p_{cow,max} = 70$ *psi*.

Los incrementos utilizados para calcular las derivadas se seleccionaron de 1×10^{-4} para la saturación, y 1×10^{-2} para la presión. El criterio de paro para salir del ciclo de Newton-Raphson se seleccionó igual a: $|\delta S_w| < 1 \times 10^{-3}$. Al igual que en el caso anterior, estos parámetros numéricos se determinaron empíricamente tomando en cuenta la estabilidad del código numérico al ejecurse para distintos tamaños de malla. El tiempo total para la simulación fue de 8,000 días y el reporte de los resultados es cada 500 días. Debido a que no existe solución analítica para el problema de los 5 pozos, éste se validó con los resultados reportados por Chen [43].

La figura 5.15, muestra los perfiles saturación S_w para seis tiempos de simulación seleccionados y un tamaño de malla de 90x90. Si 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.

Por otra parte, las figuras 5.16 y 5.17, muestran los perfiles de presión p_o y el campo de velocidad normalizado para la fase agua. Los tiempos de simulación mostrados son los mismos que para la saturación, tomando en cuenta ambas figuras, se nota que el comportamiento corresponde a lo descrito para el perfil de saturación, por ejemplo, para 500 días de simulación se muestra el perfil de presión que corresponde al campo de velocidad de 500 días, el cual indica que el agua ha avanzado hasta casi la mitad del domino.

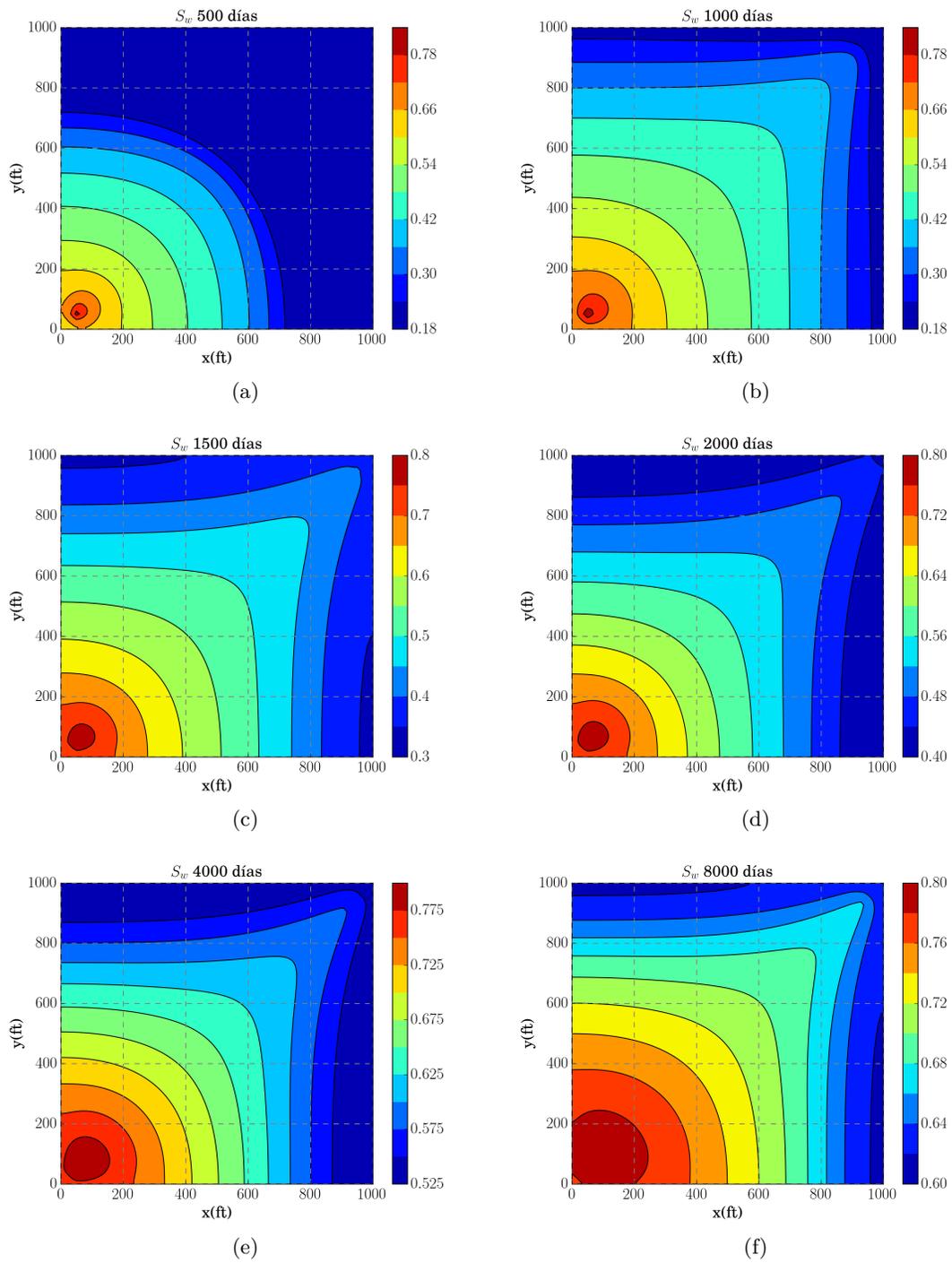


Figura 5.15: Perfiles de saturación S_w para un tamaño de malla de 90x90 celdas.

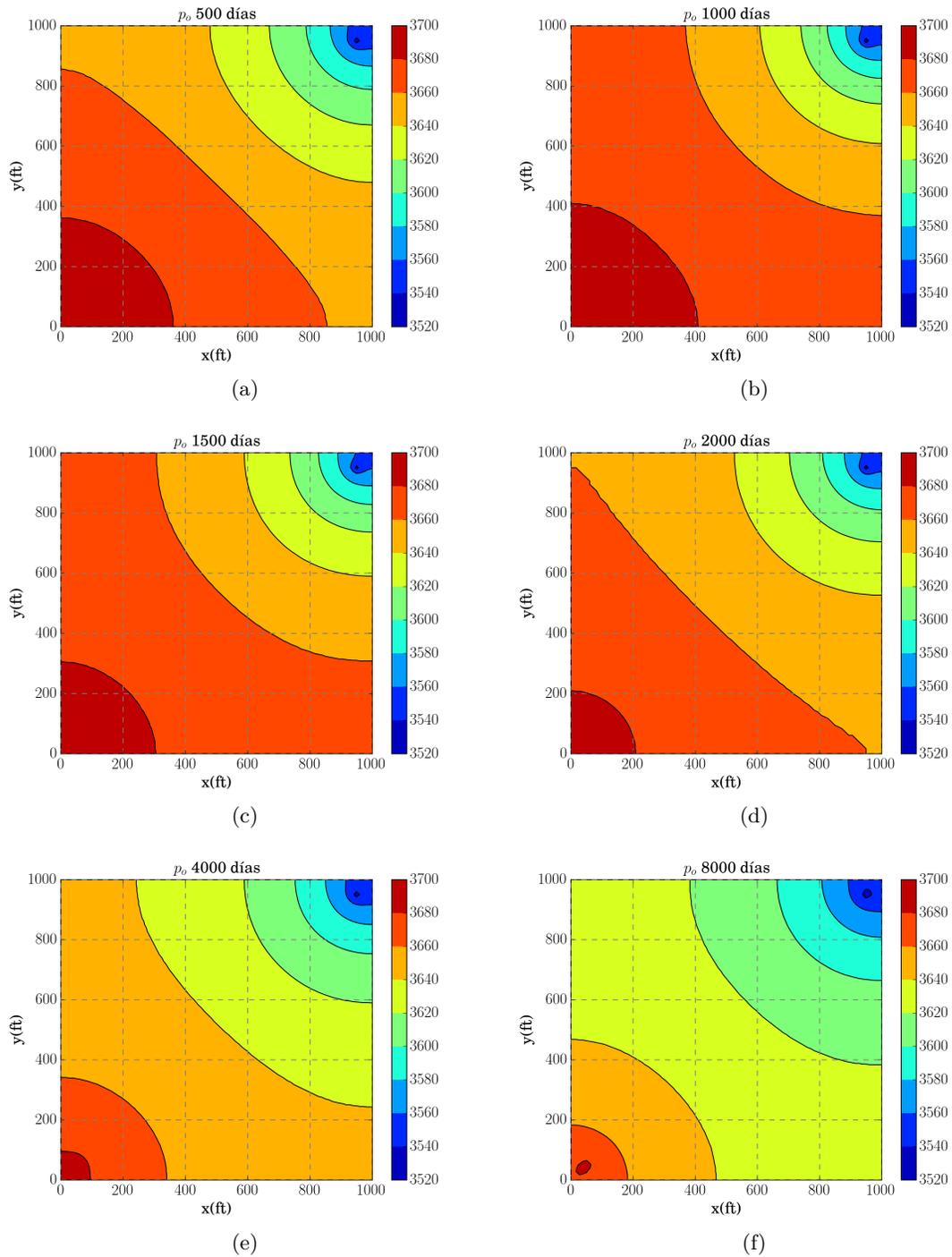


Figura 5.16: Perfiles de presión de la fase aceite p_o (psi) para un tamaño de malla de 90x90 celdas.

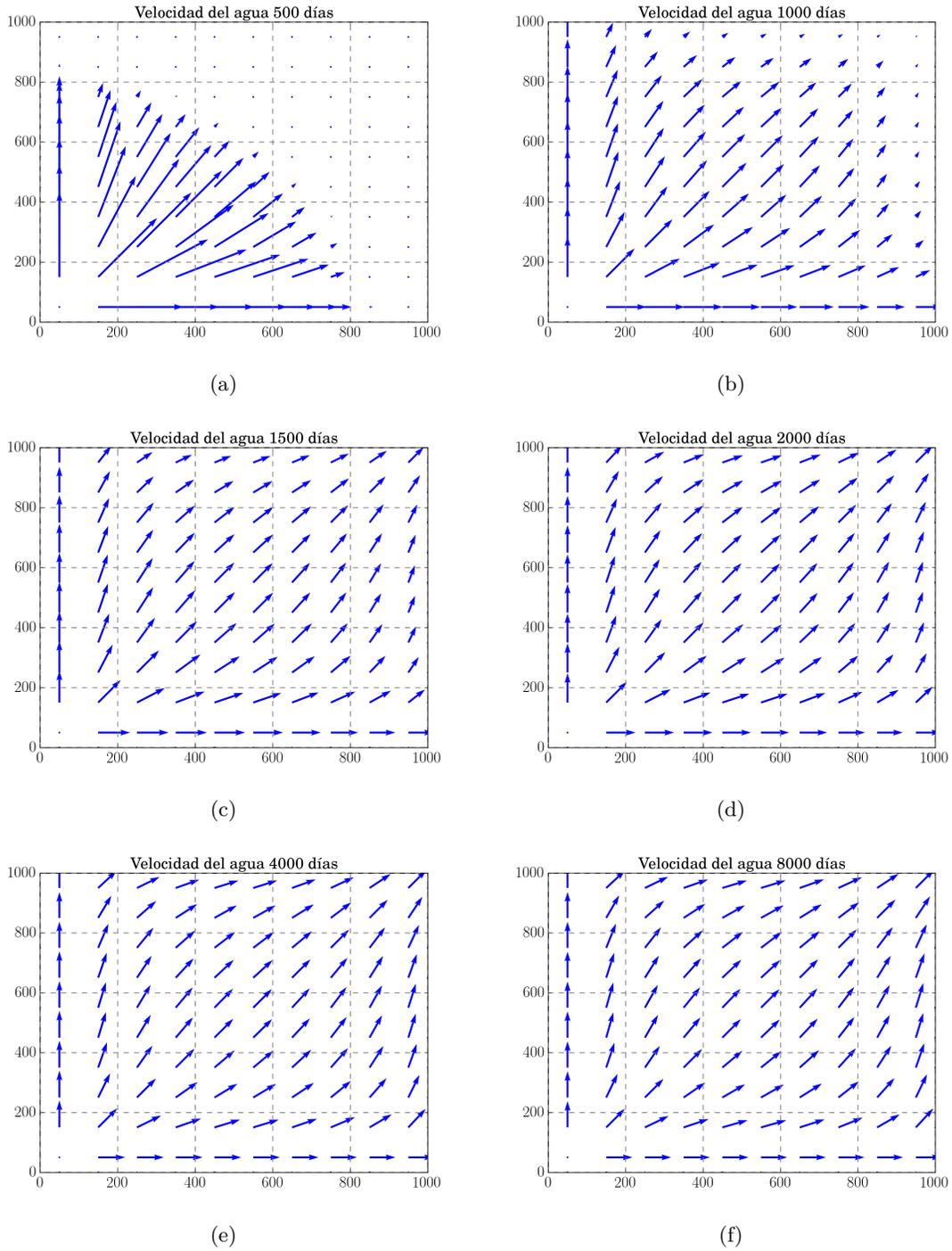


Figura 5.17: Campo de velocidad de la fase agua para un tamaño de malla de 90x90 celdas.

Para validar los resultados obtenidos, éstos se comparan con los resultados de las curvas de producción de fluidos y el flujo fraccional reportados por Chen [43] para el pozo productor. La figura 5.18, muestra la producción de agua y aceite durante todo el tiempo de simulación. Se puede notar que los resultados son cualitativamente similares. En la figura 5.18 b), se aprecia que el corte de agua ocurre pasados los 1,000 días de simulación, al igual que el resultado reportado por Chen. Mediante la comparación de estos resultados se deduce que el modelo numérico bidimensional está correctamente implementado.

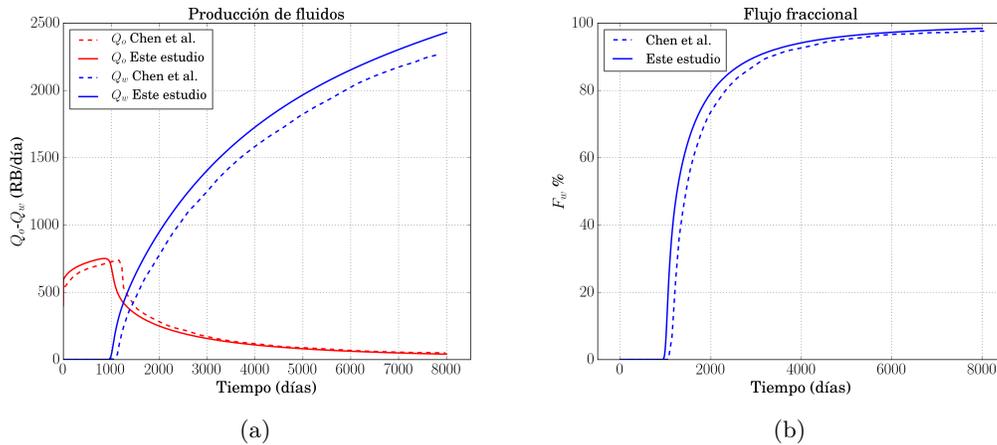


Figura 5.18: a) Producción de fluidos y b) flujo fraccional F_w . Resultados reportados por Chen, línea punteada, y los obtenidos en este trabajo, línea continua.

5.4.3. Pozos horizontales: The Seventh SPE Project

El problema reportado por la SPE como *benchmark* describe la inyección de agua y producción de aceite mediante el uso de pozos horizontales, este fue adaptado por Chen para flujo en dos fases (véase referencias [52] y [43]). Las consideraciones del modelo conceptual son las siguientes:

1. El desplazamiento ocurre en un medio tridimensional.
2. Medio isotrópico.
3. Se consideran los efectos de la presión capilar y las fuerzas de gravedad.
4. Existen fuentes (pozo inyector) y sumideros (pozo productor), el número de éstas depende de las capas utilizables para la simulación.
5. Los fluidos son incompresibles e inmiscibles.

Las ecuaciones para modelar este problema físico son:

$$-\nabla \cdot \left[\bar{k} \lambda \nabla p_o - \bar{k} \lambda_w \frac{dp_c}{dS_w} \nabla S_w - (\lambda_o \rho_o + \lambda_w \rho_w) \phi \nabla z \right] = \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}, \quad (5.52)$$

$$\phi \frac{\partial S_w}{\partial t} - \nabla \cdot \left[\bar{k} \lambda_w \left(\nabla p_o - \frac{dp_c}{dS_w} \nabla S_w - \rho_w \phi \nabla z \right) \right] = \frac{q_w}{\rho_w}. \quad (5.53)$$

Para resolver este problema, la SPE propone un tamaño de malla de 10x10x7 volúmenes o celdas, la cual está refinada al centro del eje y , con el objetivo de colocar los pozos de inyección y de producción. La longitud de las celdas en el eje x son uniformes e iguales a 300ft ($\delta x = 300\text{ft}$). Para las celdas en el eje y , la distribución es como sigue: $\delta y_1 = \delta y_9 = 620\text{ft}$, $\delta y_2 = \delta y_8 = 400\text{ft}$, $\delta y_3 = \delta y_7 = 200\text{ft}$, $\delta y_4 = \delta y_6 = 100\text{ft}$ y $\delta y_5 = 60\text{ft}$. Las longitudes de celda en el eje z son: $\delta z_k = 20\text{ft}$ para $k_{1,2,3,4}$, $\delta z_5 = 30\text{ft}$ y $\delta z_k = 50\text{ft}$. El pozo de inyección está en el centro de la capa δz_6 , colocado en la celda δy_5 y éste atraviesa todas las celdas paralelas al eje x . El pozo de producción está colocado en el centro de la capa δz_1 , en la celda centrada δy_5 y éste atraviesa sólo las celdas $\delta x_{6,7,8}$. La figura 5.19, muestra la distribución de las celdas de malla y la posición de los pozos. La saturación inicial para cada capa de ancho δz , está distribuida de la siguiente manera: 0.289 para la capa 1, 0.348 para la 2, 0.473 para la 3, 0.649 para la 4, 0.869 para la 5 y 1.0 para la capa 6. El centro de la capa 1 se ubica a 3,600ft de profundidad, y éste se toma en cuenta como referencia para los pozos. Las permeabilidades relativas $k_{r\alpha}$ y la presión capilar p_{cow} como función de la saturación S_w , se proporcionan en la tabla 5.5, y los parámetros físicos para llevar a cabo la simulación se muestran en la tabla 5.6.

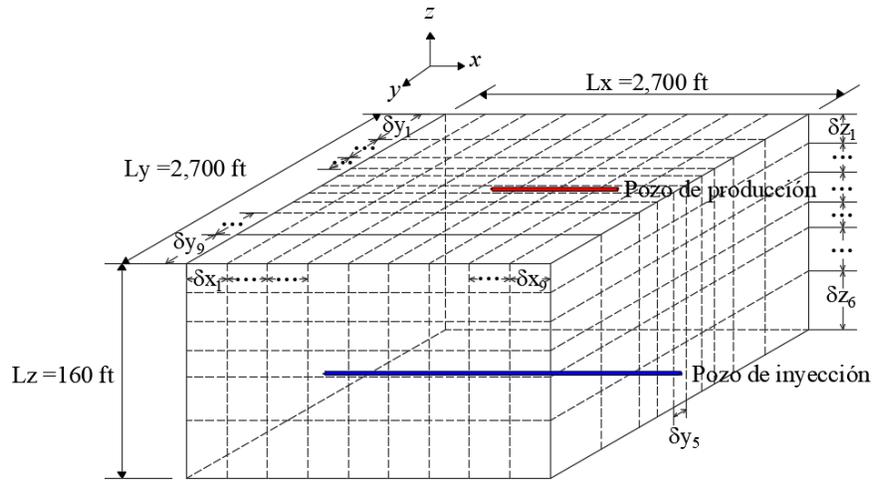


Figura 5.19: Dominio físico para el problema de los pozos horizontales. (Ilustración adaptada de [43].)

Tabla 5.5: Permeabilidades relativas k_{rw} , k_{ro} y presión capilar p_{cow} .

S_w	k_{rw}	k_{ro}	p_{cow}
0.2200	0.0000	1.0000	6.3000
0.3000	0.0700	0.4000	3.6000
0.4000	0.1500	0.1250	2.7000
0.0500	0.2400	0.0649	2.2500
0.6000	0.3300	0.0048	1.8000
0.8000	0.6500	0.0000	0.9000
0.9000	0.8300	0.0000	0.4500
1.0000	1.0000	0.0000	0.0000

(Datos tomados de [52].)

Tabla 5.6: Parámetros para la simulación del problema *The Seventh SPE Project*.

Propiedad	Valor
Tamaño del dominio $L_x \times L_y \times L_z$	2,700x2,700x160 (ft)
Permeabilidad absoluta en las capas x - y ($k_{11} = k_{22}$)	300 (mD)
Permeabilidad absoluta en dirección z (k_{33})	30 (mD)
Porosidad (ϕ)	0.20
Viscosidad del agua (μ_w)	0.96 (cP)
Viscosidad del aceite (μ_o)	0.954 (cP)
Saturación residual del agua (S_{wr})	0.22
Saturación residual del aceite (S_{or})	0
Presión de inyección en el fondo pozo (p_{wb}^{inj})	3,651.4 (psi)
Presión de producción en el fondo del pozo (p_{wb}^{out})	3,513.6 (psi)

(Datos tomados de [52].)

Los incrementos utilizados para calcular las derivadas se seleccionaron de 1×10^{-5} para la saturación, y 1×10^{-2} para la presión. El criterio de paro para salir del ciclo de Newton-Raphson se seleccionó igual a: $|\delta S_w| < 1 \times 10^{-4}$. El tiempo total para la simulación fue de 1,500 días y el reporte de los resultados es cada 100 días.

La figura 5.20, muestra los perfiles de saturación S_w en la capa del pozo productor para seis tiempos de simulación seleccionados. Fijando la atención en los perfiles de 100 y 300 días de simulación, se nota que el agua apenas ha comenzado a barrer el aceite presente en la capa, para los perfiles de 600 y 900 días, el agua móvil casi ha llegado al otro extremo del dominio, por último, en los perfiles de 1,200 y 1,500 días, se aprecia que el agua móvil ha cubierto poco más de la mitad del dominio mostrado.

En la figura 5.21, se presentan los perfiles de presión del aceite p_o , en ésta se notan cambios apenas perceptibles en el perfil de presión del aceite. Este comportamiento es el esperado, ya que para todos los tiempos de simulación la saturación de aceite S_o es mayor que la saturación de agua S_w . Además, la presión capilar es relativamente baja para cualquier tiempo de simulación.

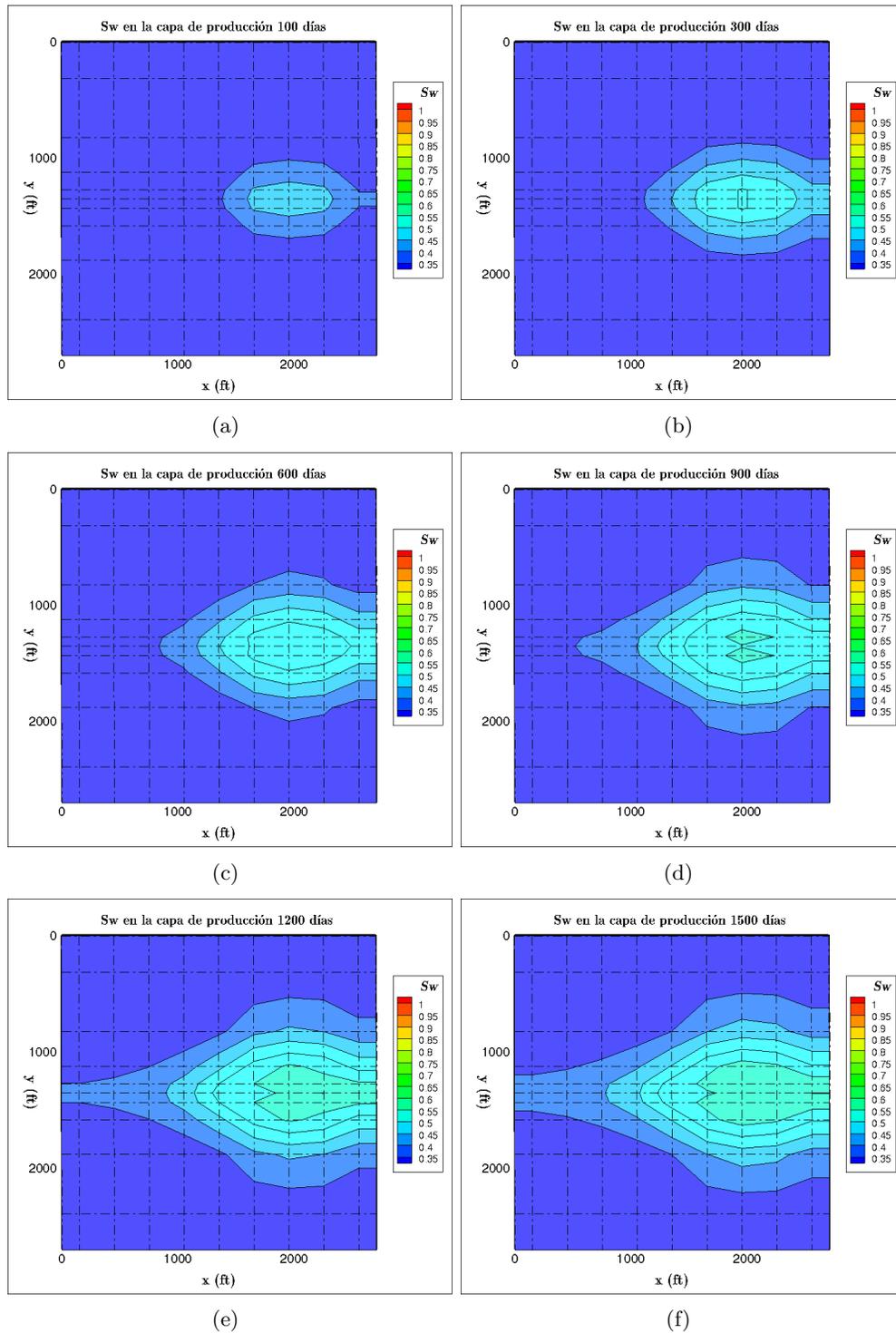


Figura 5.20: Perfiles de saturación de agua S_w en la capa de producción.

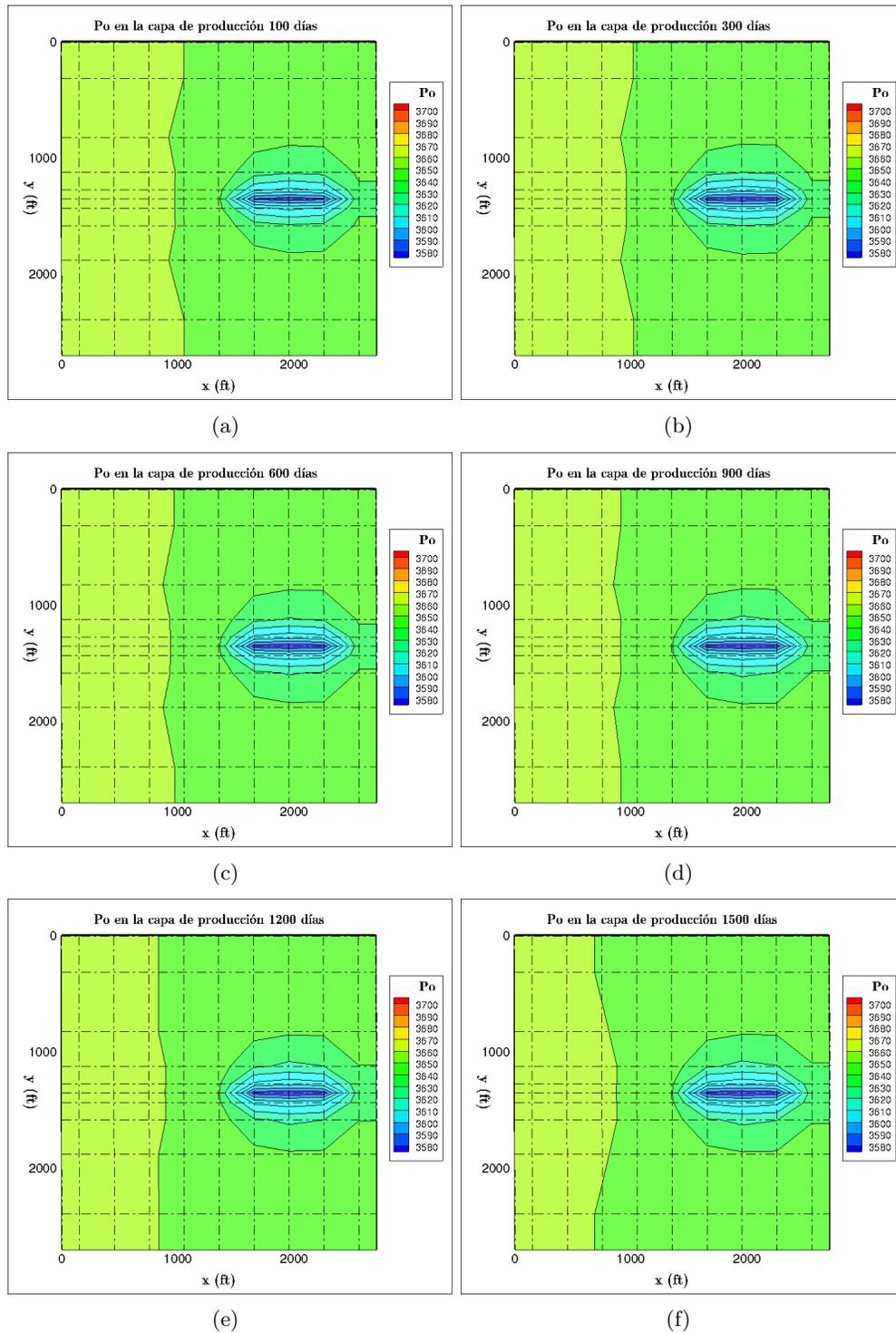


Figura 5.21: Perfiles de presión de la fase aceite p_o (psi) en la capa de producción.

Para validar los resultados obtenidos, éstos se comparan con los resultados de las curvas de producción de aceite presentadas por Chen [44] basadas en el problema de la SPE [52]. La figura 5.22, muestra la producción de aceite durante todo el tiempo de simulación. Se puede notar que los resultados son cualitativamente similares a los reportados por Chen [44] y la SPE [52], aunque en la curva reportada por Chen, la producción de aceite declina ligeramente más rápido, en esta figura, también se comparan las curvas de producción de aceite acumulado. En ambas curvas existe una pequeña diferencia, ésta puede deberse a cuestiones del método numérico utilizado, ya que el autor reporta un paso de tiempo adaptativo y en este estudio el paso de tiempo es fijo. Otra causa de esta diferencia, puede deberse al factor de conversión para convertir barriles de reservorio (RB) a barriles de inventario o superficie (STB), puesto que en este trabajo, se utilizó el factor de conversión reportado por la SPE [52]. Los resultados obtenidos para este problema indican que el modelo numérico tridimensional fue correctamente implementado.

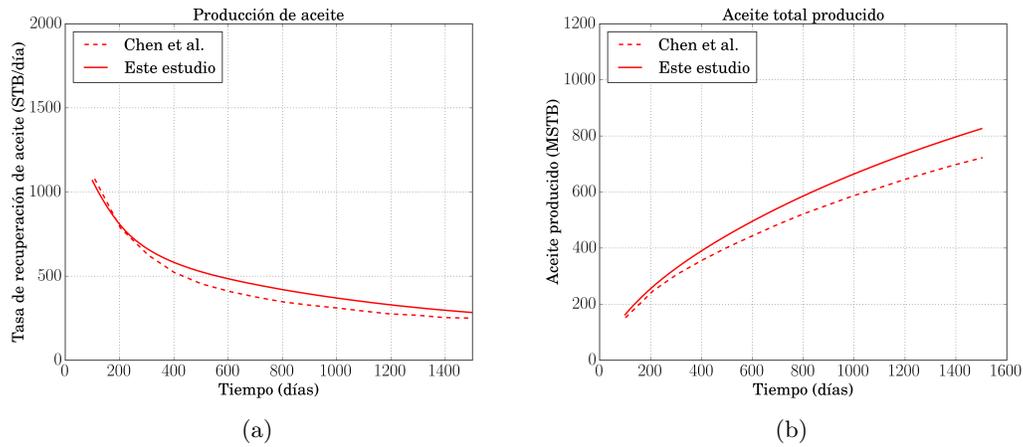


Figura 5.22: a) Curvas de producción de aceite y b) producción de aceite acumulado. Resultados reportados por Chen, línea punteada, y los obtenidos en este trabajo, línea continua.

5.5. Modelo numérico en dominios no ortogonales

En esta sección se transforman y se resuelven numéricamente las ecuaciones gobernantes de flujo incompresible bifásico en medios porosos para simular numéricamente el proceso de inyección de agua en dominios no ortogonales.

5.5.1. Transformación de la ecuación de saturación de agua

Siguiendo el procedimiento matemático presentado en el Capítulo 4 de este documento, la ecuación de saturación de agua es:

$$\phi \frac{\partial S_w}{\partial t} - \nabla \cdot \left[\bar{k} \lambda_w \left(\nabla p_o - \frac{dp_c}{dS_w} \nabla S_w - \rho_w \phi \nabla z \right) \right] = \frac{q_w}{\rho_w}, \quad (5.54)$$

que también puede escribirse como:

$$\frac{\partial Q}{\partial t} - \frac{\partial E}{\partial x} - \frac{\partial F}{\partial y} - \frac{\partial G}{\partial z} = S, \quad (5.55)$$

donde:

$$Q = \phi S_w, \quad (5.56)$$

$$E = a_{11}^w \frac{\partial p_o}{\partial x} - b_{11}^w \frac{\partial S_w}{\partial x} - c_{11}^w \frac{\partial z}{\partial x} = k_{11} \frac{k_{rw}}{\mu_w} \frac{\partial p_o}{\partial x} - k_{11} \frac{k_{rw}}{\mu_w} \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial x} - k_{11} \frac{k_{rw}}{\mu_w} \rho_w \phi \frac{\partial z}{\partial x}, \quad (5.57)$$

$$F = a_{22}^w \frac{\partial p_o}{\partial y} - b_{22}^w \frac{\partial S_w}{\partial y} - c_{22}^w \frac{\partial z}{\partial y} = k_{22} \frac{k_{rw}}{\mu_w} \frac{\partial p_o}{\partial y} - k_{22} \frac{k_{rw}}{\mu_w} \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial y} - k_{22} \frac{k_{rw}}{\mu_w} \rho_w \phi \frac{\partial z}{\partial y}, \quad (5.58)$$

$$G = a_{33}^w \frac{\partial p_o}{\partial x} - b_{33}^w \frac{\partial S_w}{\partial x} - c_{33}^w \frac{\partial z}{\partial x} = k_{11} \frac{k_{rw}}{\mu_w} \frac{\partial p_o}{\partial z} - k_{11} \frac{k_{rw}}{\mu_w} \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial z} - k_{33} \frac{k_{rw}}{\mu_w} \rho_w \phi \frac{\partial z}{\partial z}, \quad (5.59)$$

$$S = \frac{q_w}{\rho_w}. \quad (5.60)$$

Aplicando la metodología descrita en el Capítulo 4, la ecuación (5.55), puede escribirse como:

$$\begin{aligned} \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) &= \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_\alpha}{\partial \xi} + A_{12} \frac{\partial p_\alpha}{\partial \eta} + A_{13} \frac{\partial p_\alpha}{\partial \zeta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} - B_{13} \frac{\partial S_w}{\partial \zeta} - C_{11} \frac{\partial z}{\partial \xi} - C_{12} \frac{\partial z}{\partial \eta} - C_{13} \frac{\partial z}{\partial \zeta} \right] \\ &+ \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_\alpha}{\partial \xi} + A_{22} \frac{\partial p_\alpha}{\partial \eta} + A_{23} \frac{\partial p_\alpha}{\partial \zeta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} - B_{23} \frac{\partial S_w}{\partial \zeta} - C_{21} \frac{\partial z}{\partial \xi} - C_{22} \frac{\partial z}{\partial \eta} - C_{23} \frac{\partial z}{\partial \zeta} \right] \\ &+ \frac{\partial}{\partial \zeta} \left[A_{31} \frac{\partial p_\alpha}{\partial \xi} + A_{32} \frac{\partial p_\alpha}{\partial \eta} + A_{33} \frac{\partial p_\alpha}{\partial \zeta} - B_{31} \frac{\partial S_w}{\partial \xi} - B_{32} \frac{\partial S_w}{\partial \eta} - B_{33} \frac{\partial S_w}{\partial \zeta} - C_{31} \frac{\partial z}{\partial \xi} - C_{32} \frac{\partial z}{\partial \eta} - C_{33} \frac{\partial z}{\partial \zeta} \right] + \frac{S}{J}, \end{aligned} \quad (5.61)$$

donde las componentes de los tensores diagonales $A_{i,j}$, $B_{i,j}$ y $C_{i,j}$, están dadas por:

$$\begin{aligned} A_{11} &= \frac{1}{J} (a_{11}^w \xi_x^2 + a_{22}^w \xi_y^2 + a_{33}^w \xi_z^2), & B_{11} &= \frac{1}{J} (b_{11}^w \xi_x^2 + b_{22}^w \xi_y^2 + b_{33}^w \xi_z^2), \\ A_{12} &= \frac{1}{J} (a_{11}^w \xi_x \eta_x + a_{22}^w \xi_y \eta_y + a_{33}^w \xi_z \eta_z), & B_{12} &= \frac{1}{J} (b_{11}^w \xi_x \eta_x + b_{22}^w \xi_y \eta_y + b_{33}^w \xi_z \eta_z), \\ A_{13} &= \frac{1}{J} (a_{11}^w \xi_x \zeta_x + a_{22}^w \xi_y \zeta_y + a_{33}^w \xi_z \zeta_z), & B_{13} &= \frac{1}{J} (b_{11}^w \xi_x \zeta_x + b_{22}^w \xi_y \zeta_y + b_{33}^w \xi_z \zeta_z), \\ A_{21} &= \frac{1}{J} (a_{11}^w \xi_x \eta_x + a_{22}^w \xi_y \eta_y + a_{33}^w \xi_z \eta_z), & B_{21} &= \frac{1}{J} (b_{11}^w \xi_x \eta_x + b_{22}^w \xi_y \eta_y + b_{33}^w \xi_z \eta_z), \\ A_{22} &= \frac{1}{J} (a_{11}^w \eta_x^2 + a_{22}^w \eta_y^2 + a_{33}^w \eta_z^2), & B_{22} &= \frac{1}{J} (b_{11}^w \eta_x^2 + b_{22}^w \eta_y^2 + b_{33}^w \eta_z^2), \\ A_{23} &= \frac{1}{J} (a_{11}^w \eta_x \zeta_x + a_{22}^w \eta_y \zeta_y + a_{33}^w \eta_z \zeta_z), & B_{23} &= \frac{1}{J} (b_{11}^w \eta_x \zeta_x + b_{22}^w \eta_y \zeta_y + b_{33}^w \eta_z \zeta_z), \\ A_{31} &= \frac{1}{J} (a_{11}^w \xi_x \zeta_x + a_{22}^w \xi_y \zeta_y + a_{33}^w \xi_z \zeta_z), & B_{31} &= \frac{1}{J} (b_{11}^w \xi_x \zeta_x + b_{22}^w \xi_y \zeta_y + b_{33}^w \xi_z \zeta_z), \\ A_{32} &= \frac{1}{J} (a_{11}^w \eta_x \zeta_x + a_{22}^w \eta_y \zeta_y + a_{33}^w \eta_z \zeta_z), & B_{32} &= \frac{1}{J} (b_{11}^w \eta_x \zeta_x + b_{22}^w \eta_y \zeta_y + b_{33}^w \eta_z \zeta_z), \\ A_{33} &= \frac{1}{J} (a_{11}^w \zeta_x^2 + a_{22}^w \zeta_y^2 + a_{33}^w \zeta_z^2), & B_{33} &= \frac{1}{J} (b_{11}^w \zeta_x^2 + b_{22}^w \zeta_y^2 + b_{33}^w \zeta_z^2). \end{aligned} \quad (5.62)$$

$$\begin{aligned}
C_{11} &= \frac{1}{j} (c_{11}^w \xi_x^2 + c_{22}^w \xi_y^2 + c_{33}^w \xi_z^2), \\
C_{12} &= \frac{1}{j} (c_{11}^w \xi_x \eta_x + c_{22}^w \xi_y \eta_y + c_{33}^w \xi_z \eta_z), \\
C_{13} &= \frac{1}{j} (c_{11}^w \xi_x \zeta_x + c_{22}^w \xi_y \zeta_y + c_{33}^w \xi_z \zeta_z), \\
C_{21} &= \frac{1}{j} (c_{11}^w \xi_x \eta_x + c_{22}^w \xi_y \eta_y + c_{33}^w \xi_z \eta_z), \\
C_{22} &= \frac{1}{j} (c_{11}^w \eta_x^2 + c_{22}^w \eta_y^2 + c_{33}^w \eta_z^2), \\
C_{23} &= \frac{1}{j} (c_{11}^w \eta_x \zeta_x + c_{22}^w \eta_y \zeta_y + c_{33}^w \eta_z \zeta_z), \\
C_{31} &= \frac{1}{j} (c_{11}^w \xi_x \zeta_x + c_{22}^w \xi_y \zeta_y + c_{33}^w \xi_z \zeta_z), \\
C_{32} &= \frac{1}{j} (c_{11}^w \eta_x \zeta_x + c_{22}^w \eta_y \zeta_y + c_{33}^w \eta_z \zeta_z), \\
C_{33} &= \frac{1}{j} (c_{11}^w \zeta_x^2 + c_{22}^w \zeta_y^2 + c_{33}^w \zeta_z^2).
\end{aligned} \tag{5.63}$$

5.5.2. Transformación de la ecuación de presión

La ecuación de presión es:

$$-\nabla \cdot \left[\bar{k} \lambda \nabla p_o - \bar{k} \lambda_w \frac{dp_c}{dS_w} \nabla S_w - (\lambda_o \rho_o + \lambda_w \rho_w) \wp \nabla z \right] = \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}, \tag{5.64}$$

ésta puede escribirse de la forma:

$$\frac{\partial Q}{\partial t} - \frac{\partial E}{\partial x} - \frac{\partial F}{\partial y} - \frac{\partial G}{\partial z} = S, \tag{5.65}$$

donde:

$$Q = 0, \tag{5.66}$$

$$E = a_{11} \frac{\partial p}{\partial x} - b_{11} \frac{\partial S_w}{\partial x} - c \frac{\partial z}{\partial x} = k_{11} \lambda \frac{\partial p_o}{\partial x} - k_{11} \lambda_w \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial x} - (\lambda_w \rho_w + \lambda_o \rho_o) \wp \frac{\partial z}{\partial x}, \tag{5.67}$$

$$F = a_{22} \frac{\partial p}{\partial y} - b_{22} \frac{\partial S_w}{\partial y} - c \frac{\partial z}{\partial y} = k_{22} \lambda \frac{\partial p_o}{\partial y} - k_{22} \lambda_w \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial y} - (\lambda_w \rho_w + \lambda_o \rho_o) \wp \frac{\partial z}{\partial y}, \tag{5.68}$$

$$G = a_{33} \frac{\partial p}{\partial z} - b_{33} \frac{\partial S_w}{\partial z} - c \frac{\partial z}{\partial z} = k_{33} \lambda \frac{\partial p_o}{\partial z} - k_{33} \lambda_w \frac{dp_c}{dS_w} \frac{\partial S_w}{\partial z} - (\lambda_w \rho_w + \lambda_o \rho_o) \wp \frac{\partial z}{\partial z}, \tag{5.69}$$

$$S = \frac{q_w}{\rho_w} + \frac{q_o}{\rho_o}. \tag{5.70}$$

La ecuación (5.65), puede escribirse como (véase el Capítulo 4):

$$\begin{aligned}
& \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_\alpha}{\partial \xi} + A_{12} \frac{\partial p_\alpha}{\partial \eta} + A_{13} \frac{\partial p_\alpha}{\partial \zeta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} - B_{13} \frac{\partial S_w}{\partial \zeta} - C_{11} \frac{\partial z}{\partial \xi} - C_{12} \frac{\partial z}{\partial \eta} - C_{13} \frac{\partial z}{\partial \zeta} \right] \\
& + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_\alpha}{\partial \xi} + A_{22} \frac{\partial p_\alpha}{\partial \eta} + A_{23} \frac{\partial p_\alpha}{\partial \zeta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} - B_{23} \frac{\partial S_w}{\partial \zeta} - C_{21} \frac{\partial z}{\partial \xi} - C_{22} \frac{\partial z}{\partial \eta} - C_{23} \frac{\partial z}{\partial \zeta} \right] \\
& + \frac{\partial}{\partial \zeta} \left[A_{31} \frac{\partial p_\alpha}{\partial \xi} + A_{32} \frac{\partial p_\alpha}{\partial \eta} + A_{33} \frac{\partial p_\alpha}{\partial \zeta} - B_{31} \frac{\partial S_w}{\partial \xi} - B_{32} \frac{\partial S_w}{\partial \eta} - B_{33} \frac{\partial S_w}{\partial \zeta} - C_{31} \frac{\partial z}{\partial \xi} - C_{32} \frac{\partial z}{\partial \eta} - C_{33} \frac{\partial z}{\partial \zeta} \right] = -\frac{S}{J},
\end{aligned} \tag{5.71}$$

aquí las componentes de los tensores A_{ij} , B_{ij} y C_{ij} , tienen forma similar a aquellos dados para la ecuación de saturación. Cabe señalar que las variantes se encuentran en la construcción de los coeficientes a_{ii} , b_{ii} y c_{ii} contenidos en las ecuaciones (5.67)-(5.69).

El procedimiento para la discretización de las ecuaciones de saturación (5.61) y presión (5.71) se muestran en el Apéndice A. Por otra parte, el algoritmo numérico general y el procedimiento para la construcción del Jacobiano es idéntico al presentado en secciones previas de este capítulo, por lo que la implementación del modelo numérico en la CPU y la GPU se lleva a cabo en forma similar que para el modelo en coordenadas cartesianas.

5.5.3. Resultados de la simulación numérica en dominios no ortogonales

Verificación del código numérico

Para verificar la implementación del modelo numérico, se llevó a cabo la simulación del problema bidimensional simétrico de los 5 pozos utilizando los mismos parámetros físicos y numéricos dados en la sección anterior de este capítulo. Se utilizó una malla de 30x30 celdas para dicha simulación, además el tiempo de simulación se acortó a 2,500 días considerando un paso de tiempo de 1 día.

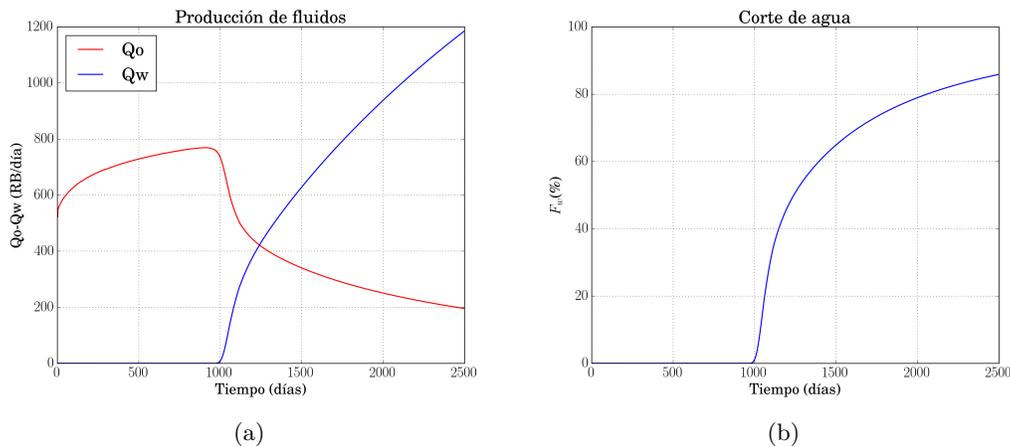


Figura 5.23: a) Curvas de producción de fluidos y b) flujo fraccional F_w .

Como se puede apreciar en la figura 5.23, las curvas de producción de fluidos y de flujo fraccional presentan el mismo comportamiento cualitativo que para el modelo cartesiano; por lo que se puede decir que el modelo numérico para dominios no ortogonales fue bien implementado, puesto que proporciona la misma solución que el modelo numérico cartesiano para un problema ortogonal. Cabe aclarar que en la literatura no existen problemas de referencia (*benchmark*) que aborden dominios irregulares.

Resultados para el dominio no ortogonal no. 1

Después de llevar a cabo la simulación en un medio regular se llevaron a cabo varias simulaciones en dominios no ortogonales considerando los mismos parámetros físicos y numéricos que para la simulación cartesiana. También se optó por conservar la misma distribución de los pozos inyector y productor. El primer dominio no ortogonal seleccionado tiene forma de paralelogramo, los centros de los volúmenes de control se muestran en la figura 5.24 para un tamaño de malla 40x40 celdas y una longitud de 1,000ft. En este problema en particular se eligió un tiempo de simulación de 2,500 días, esto con la finalidad de realizar un comparativo con los resultados del dominio ortogonal, puesto que los dominios tienen el mismo tamaño de área.

La figura 5.25, contiene los perfiles de saturación S_w para los tiempos de simulación de 250 y 750 días. Observando la figura 5.25 a), se nota que el agua ha avanzado hasta poco más de 500ft tomando como referencia el punto de inyección, mientras que en la figura 5.25 b), se observa que el agua móvil ya ha cubierto casi la mitad de dominio.

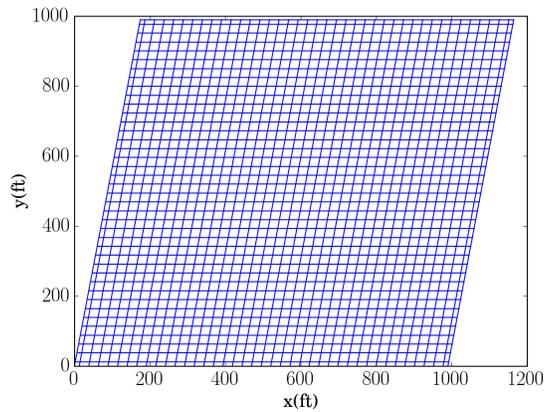


Figura 5.24: Centros de los volúmenes de control del dominio no ortogonal no. 1.

Por otro lado, la figura 5.26, muestra los campos de velocidad de la fase agua, en ésta se nota con más facilidad el avance del agua en el medio. Los campos de velocidad son congruentes con el comportamiento físico del problema.

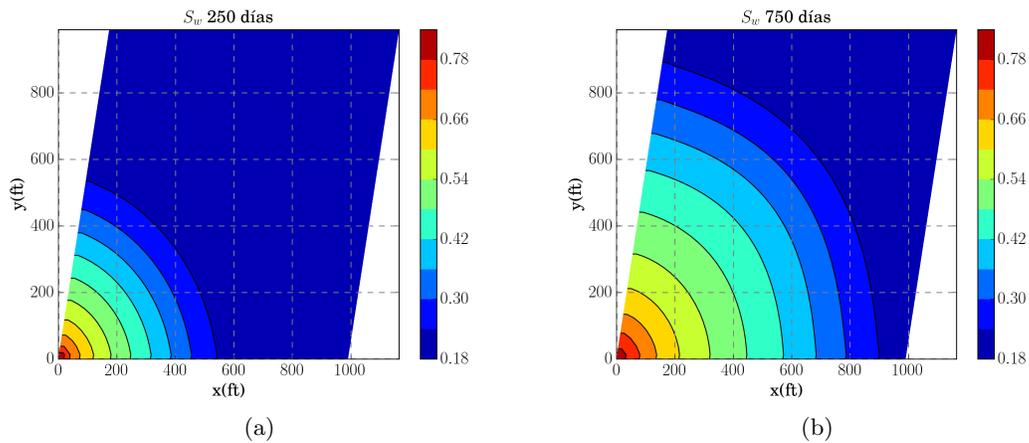


Figura 5.25: Perfiles de saturación S_w para a) 250 y b) 750 días de simulación.

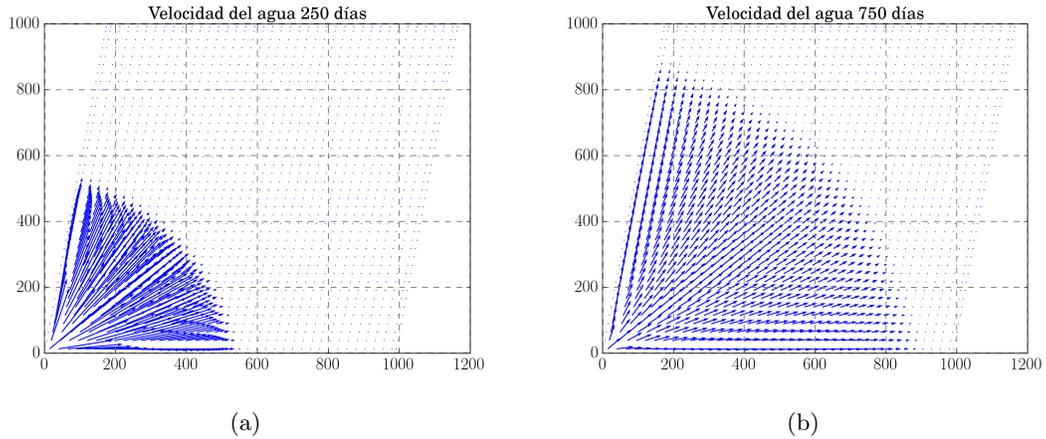


Figura 5.26: Vectores de velocidad de la fase agua para a) 250 y b) 750 días de simulación.

Los perfiles de presión p_o , se presentan en la figura 5.27. La figura 5.27 a), corresponde a la presión p_o para 250 días de simulación, en ésta se nota que la mayor presión se concentra en un radio aproximado de 380 ft. Una vez transcurridos 750 días (véase la figura 5.27 b)) el radio de presión abarca aproximadamente 500 ft, lo que indica que la presión aumenta con el avance del frente de agua.

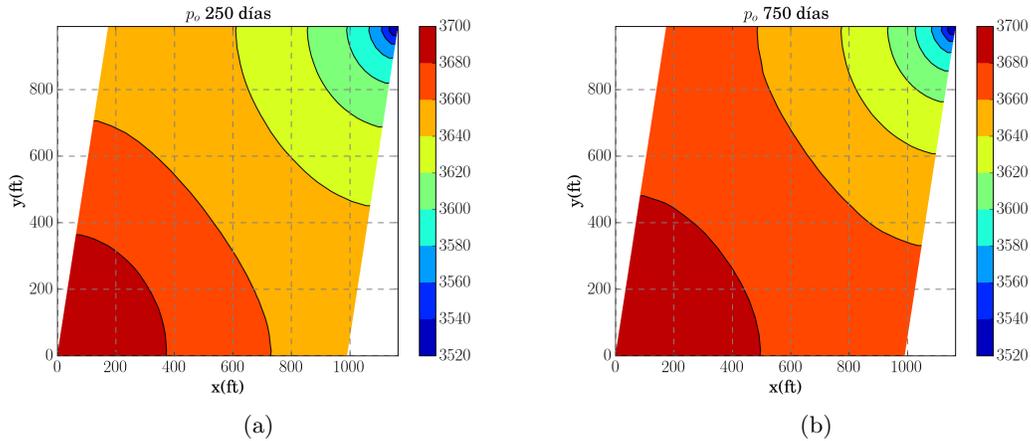


Figura 5.27: Perfiles de presión p_o (psi) para a) 250 y b) 750 días de simulación.

La figura 5.28, muestra la producción de agua y aceite, y la curva de flujo fraccional F_w , durante todo el tiempo de simulación. En esta figura, se puede notar que la producción de aceite y agua son menores cuantitativamente hablando, en comparación con su contraparte ortogonal de la figura 5.23, por ejemplo, para este caso se tiene un pico máximo de producción de 540 barriles por día, mientras que para el caso ortogonal se producen 780 barriles por día aproximadamente. En la figura 5.28 b), se aprecia que el corte de agua ocurre poco antes de

los 1,500 días de simulación, es decir, casi 500 días después que el problema ortogonal. Por lo tanto, este resultado sugiere que la forma del medio, influye directamente en el comportamiento físico del problema, puesto que para este problema en particular se tiene la misma área de simulación que en el dominio ortogonal.

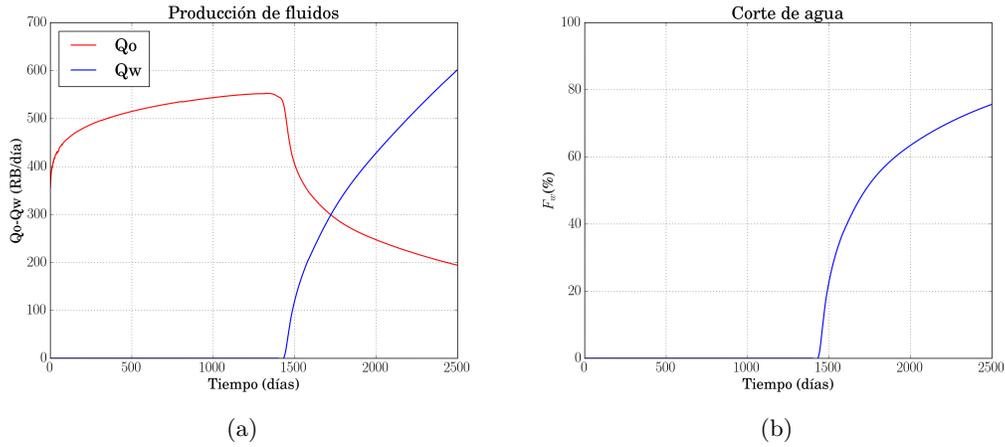


Figura 5.28: a) Curvas de producción de fluidos y b) flujo fraccional F_w .

Resultados para el dominio no ortogonal no. 2

Para este problema se seleccionó un dominio no ortogonal con forma de arco, el cual tiene las siguientes dimensiones: radio interior de 380ft, un radio exterior de 500ft y una apertura de 145 grados. La figura 5.29, muestra los centros de los volúmenes de control para un tamaño de malla 10x40 celdas. La distribución de los pozos productor e inyector es como sigue: el pozo inyector esta colocado en una esquina inferior izquierda del dominio y el pozo productor en la esquina superior derecha del dominio. El tiempo de simulación total es de 1,000 días.

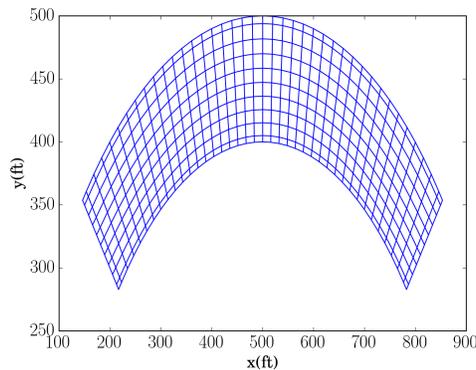


Figura 5.29: Centros de los volúmenes de control del dominio no ortogonal no. 2.

La figura 5.30, muestra los perfiles de saturación de agua S_w para los tiempos de simulación de 100 y 500 días. En la figura 5.30 a), se nota que el frente de agua ha avanzado hasta casi los 500ft y cubre aproximadamente la mitad del área de dominio, mientras que en la figura 5.30 b), el agua ya ha barrido todo el dominio y se comienza a producir por el pozo productor.

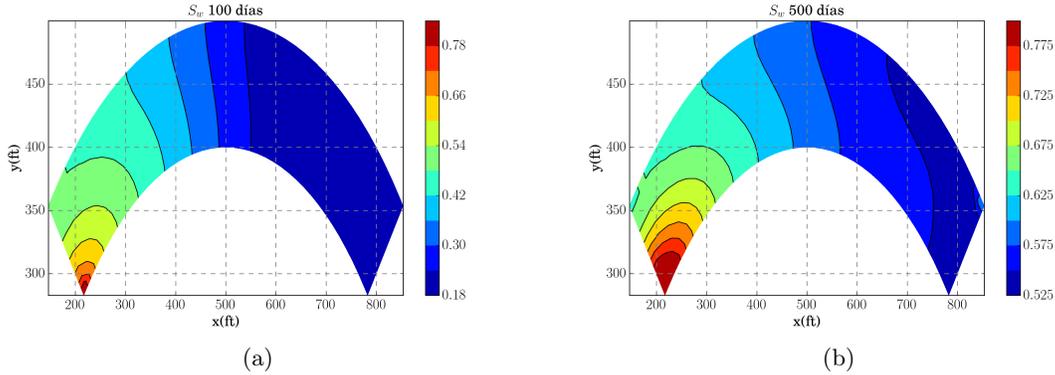


Figura 5.30: Perfiles de saturación S_w para a) 100 y b) 500 días de simulación.

Los campos de velocidad de la fase agua se muestran en la figura 5.31, observando esta figura se puede comprobar que el avance del frente de agua a los 100 días de simulación es hasta casi la mitad del dominio, mientras que a los 500 días ya ha cubierto todo el dominio.

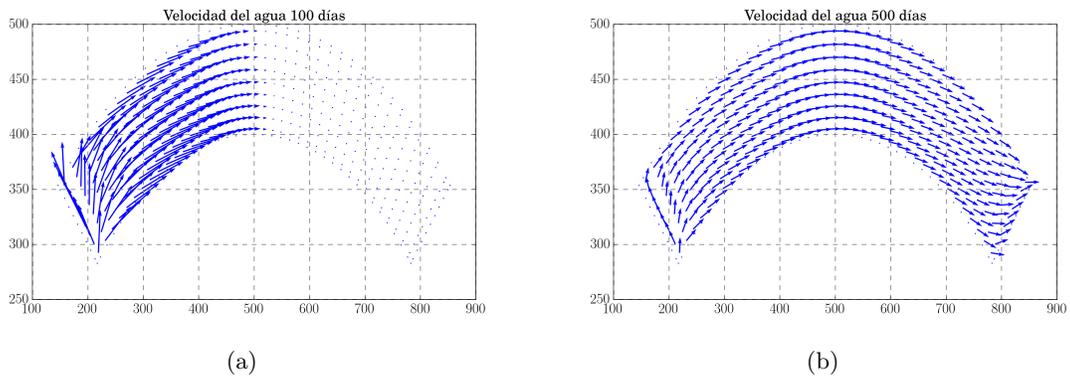


Figura 5.31: Vectores de velocidad de la fase agua para a) 100 y b) 500 días de simulación.

Los perfiles de presión p_o , correspondientes a los perfiles de saturación S_w y a los campos de velocidad mostrados en las figuras 5.30 y 5.31, se proporcionan en la figura 5.32. La figura 5.32 a), muestra la presión p_o para 100 días de simulación, en ella se nota que la mayor presión se concentra cerca de la posición del pozo inyector el cual se ubica la esquina inferior izquierda. Además, también se nota que en casi todo el dominio las isolíneas de presión son casi perpendiculares a las fronteras curvas de la geometría. Una vez transcurridos 500 días

(véase figura 5.32 b)), el perfil de presión se ha estabilizado en todo el dominio con líneas casi perpendiculares a las fronteras curvas, también se puede notar una isolínea circular que rodea al pozo productor.

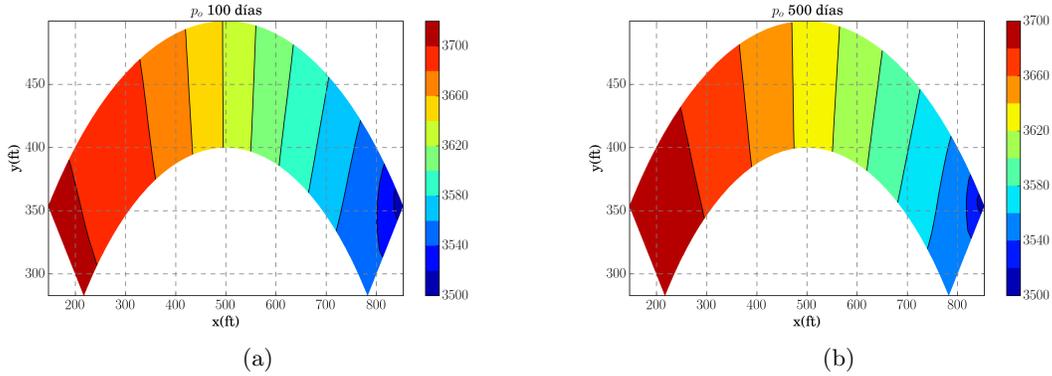


Figura 5.32: Perfiles de presión p_o (psi) para a) 100 y b) 500 días de simulación.

En la figura 5.33, se muestran la producción de agua y aceite, y la curva de flujo fraccional F_w durante todo el tiempo de simulación. En la figura 5.33 a), se nota que la producción máxima de aceite es de 390 barriles por día aproximadamente y este pico se da poco después de que ocurre el corte de agua a los 200 días de simulación. En la figura 5.33 b), correspondiente al flujo fraccional F_w , se nota que después de los 300 días el agua representa el 80% de los fluidos producidos, mientras que para los 1,000 días de simulación el agua producida es superior al 95% de los dos fluidos producidos.

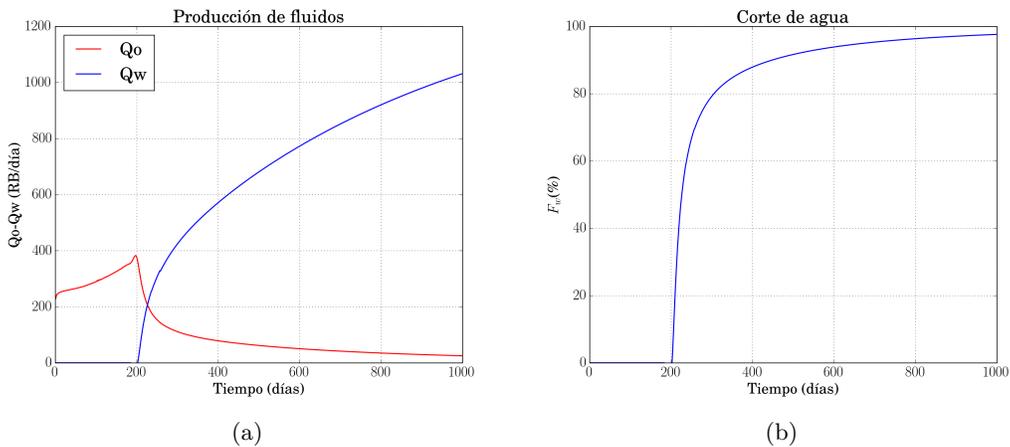


Figura 5.33: a) Curvas de producción de fluidos y b) flujo fraccional F_w .

Resultados para el dominio no ortogonal no. 3

En este ejemplo se seleccionó un dominio no ortogonal con fronteras curvas creadas en un programa de diseño asistido por computadora. La figura 5.34, muestra los centros de los volúmenes de control para un tamaño de malla 64x64 celdas, la distribución de los pozos productor e inyector se conserva y el tiempo de simulación total es de 1,000 días.

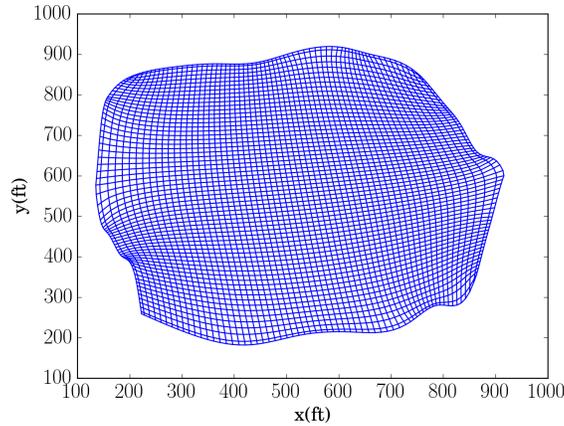


Figura 5.34: Centros de los volúmenes de control del dominio no ortogonal 3

La figura 5.35, muestra los perfiles de saturación de agua S_w para tiempos de simulación de 100 y 500 días. En la figura 5.35 a), se nota que el frente de agua ha avanzado hasta casi los 600ft, tomando como referencia el eje x , mientras que en la figura b), el frente de agua ya ha llegado hasta el otro extremo del dominio y por lo tanto, se comienza a producir por el pozo productor.

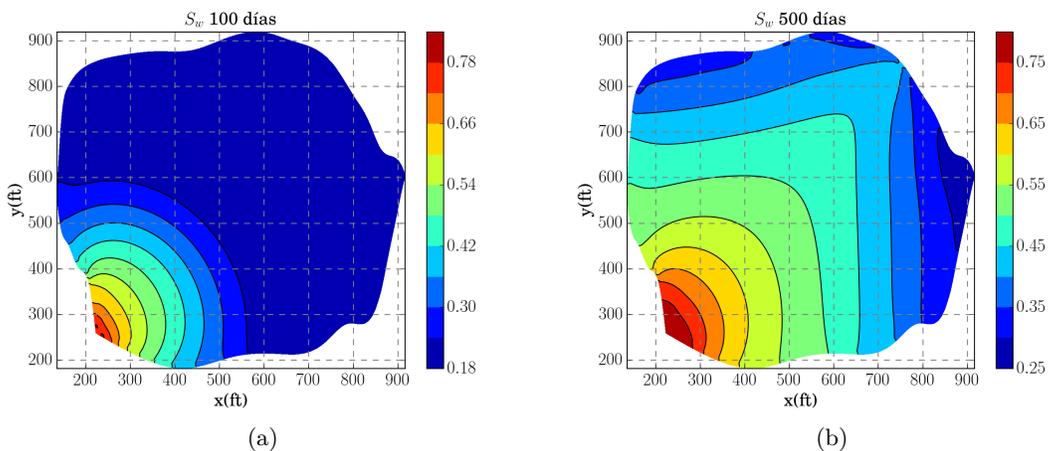


Figura 5.35: Perfiles de saturación S_w para a) 100 y b) 500 días de simulación.

Por otra parte, la figura 5.36, muestra los campos de velocidad de la fase agua. En la figura 5.36 a), se puede ver que el avance del frente de agua a los 100 días de simulación es casi una tercera parte del dominio. Observando la figura 5.36 b), se nota que la fase agua ya ha cubierto la mayoría del dominio, concordando con lo descrito en los perfiles de saturación.

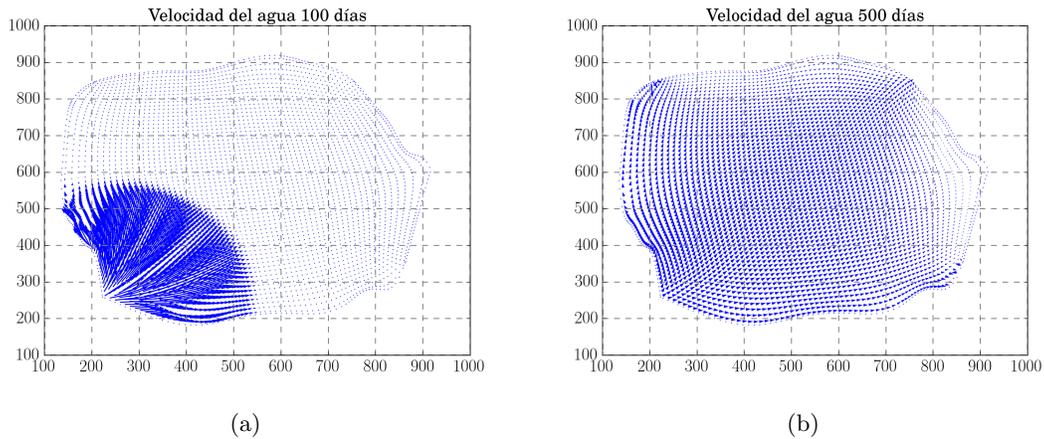


Figura 5.36: Vectores de velocidad de la fase agua para a) 100 y b) 500 días de simulación.

Los perfiles de presión p_o , se muestran en la figura 5.37. La figura 5.37 a), muestra la presión p_o para 100 días de simulación. En esta figura, se nota que la mayor presión se concentra en un radio de 400ft aproximadamente, tomando como referencia la esquina inferior izquierda y el eje y donde se encuentra el pozo inyector. La figura 5.37 b), muestra los perfiles de presión a los 500 días de simulación, aquí se nota que el radio de mayor presión ha disminuido hasta los 340ft aproximadamente, tomando la misma referencia que en el caso anterior. Sin embargo, la presión ha aumentado en casi todo el dominio, considerando que las figuras tienen la misma escala de colores.

La figura 5.38, muestra las curvas correspondientes a la producción de fluidos y el flujo fraccional F_w , durante todo el tiempo de simulación. En la figura 5.38 a), se puede notar que la producción máxima de aceite es de 700 barriles por día aproximadamente y este pico ocurre poco después del corte de agua a los 355 días de simulación. En la figura 5.38 b), se nota que después de los 800 días el agua representa el 80 % de los fluidos producidos, mientras que si se toma en cuenta a los 1,000 días de simulación el agua producida representa el 85 % de los dos fluidos producidos.

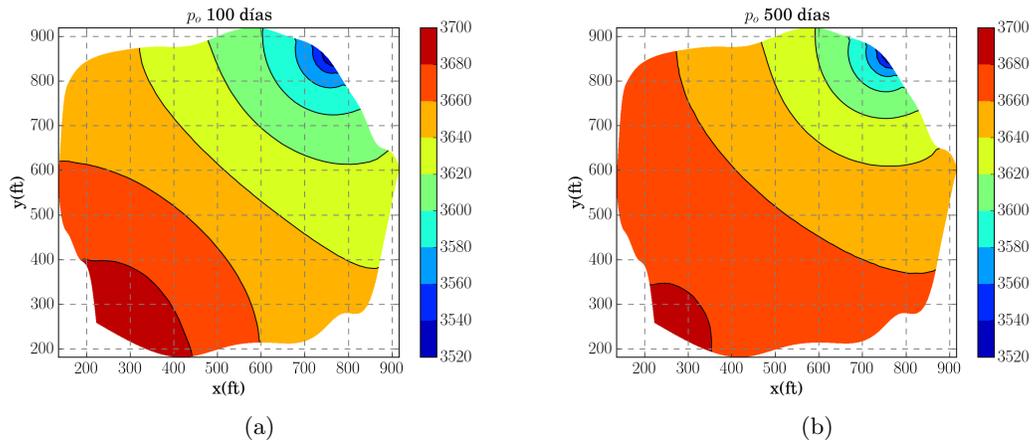


Figura 5.37: Perfiles de presión p_o (psi) para a) 100 y b) 500 días de simulación.

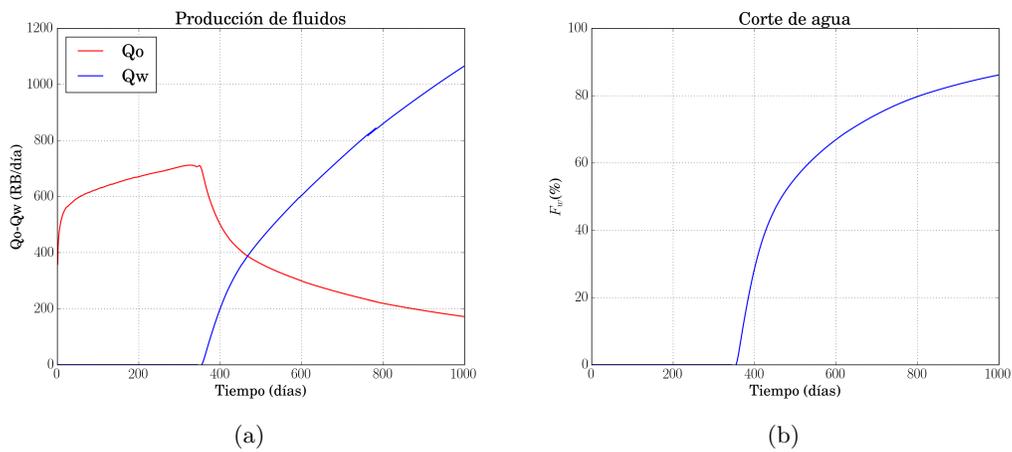


Figura 5.38: a) Curvas de producción de fluidos y b) flujo fraccional F_w .

A partir de los resultados obtenidos para los tres dominios no ortogonales presentados en esta sección, se puede decir que el código numérico simula correctamente el comportamiento físico inherente al flujo de fluidos bifásico para el proceso de inyección de agua en medios porosos.

5.6. Experimentos numéricos de desempeño

Con objetivo de analizar el desempeño de los códigos numéricos implementados tanto en CPU como en GPU, se seleccionó el problema simétrico de los 5 pozos en el sistema coordenado cartesiano, considerando un tiempo de simulación total de 10 días. Los resultados presentados en esta sección se obtuvieron mediante la ejecución de ambos códigos numéricos en una estación de trabajo que utiliza un procesador Intel(R) i7(R) CPU 3820 3.60 GHz, 16 Gigabytes de memoria RAM y una tarjeta GPU NVIDIA Tesla C2075(R) con 448 cuda-cores y 6 Gigabytes de memoria dedicada. Cabe señalar que cada ejemplo fue ejecutado 3 veces, por lo que los resultados presentados son los promedios aritméticos de las ejecuciones hechas.

Como se señaló en secciones previas, el cálculo de la matriz Jacobiana fue paralelizado. La Tabla 5.7 muestra el tiempo de ejecución promedio para cada paso de Newton-Rapshon. La mayoría de los autores indican que el 70 % de tiempo de cómputo se consume en la solución del sistema de ecuaciones lineales. Sin embargo, como se puede apreciar en la Tabla 5.7 el tiempo de cómputo que se consume en el cálculo del Jacobiano incrementa considerablemente cuando el tamaño de la malla es más grande. A manera de ejemplo, se consumen 0.796s cuando el tamaño de malla es 550x550 en la CPU, mientras que utilizando la GPU sólo se gastan 0.0379s, eso significa 21.0x de aceleración. Este resultado muestra un considerable ahorro de tiempo tomando en cuenta que este procedimiento tiene que ser repetido cada iteración de Newton-Rapshon. También en la tabla 5.7 se nota que la aceleración máxima es de 22.96x y se obtiene cuando se utiliza un tamaño de malla de 150x150 (22,500 nodos), este resultado es debido a que el número máximo de threads (hilos de cálculo) que se pueden ejecutar al mismo tiempo es de 21,504 para la GPU utilizada. De hecho, debido a que se mantuvo constante el tamaño de malla utilizado en la ejecución del kernel para construir el Jacobiano (grid=64x64, block=16x16) y además que la frecuencia de reloj de la GPU (1.15 GHz) es menor a la frecuencia de trabajo de la CPU, la aceleración máxima alcanzada pareciera mantenerse constante en aproximadamente 21x.

Tabla 5.7: Tiempo promedio obtenido para el cálculo del Jacobiano.

Tamaño de malla	CPU C++ (s)	GPU CUDA (s)	Aceleración x
30x30	0.00284	0.00084	3.38x
90x90	0.02340	0.00164	14.26x
150x150	0.07141	0.00311	22.96x
250x250	0.15049	0.00754	19.95x
550x550	0.79686	0.03793	21.00x

Por otra parte, se utilizó el método BICGSTAB con el preconditionador de Jacobi (matriz diagonal del sistema) para resolver el sistema lineal de ecuaciones en ambos casos, tanto para la CPU como en la GPU. Cabe aclarar que los métodos mencionados ya están incluidos en las bibliotecas *Eigen ++* y *CUSP* [48] [49]. Los tiempos de cómputo se muestran en la tabla 5.6. Los resultados indican que la CPU es más rápida que la GPU cuando el sistema lineal es relativamente pequeño (45,000 incógnitas). Por otra parte, cuando el sistema incrementa

de 45,000 a 125,000 incógnitas el tiempo de cómputo obtenido por la GPU es menor en comparación con el obtenido por la CPU. Para un sistema de 605,000 incógnitas se logra una aceleración de 2.2x.

Tabla 5.8: Tiempo de cómputo promedio obtenido para la solución del sistema de ecuaciones lineales mediante el método BICGSTAB con el preconditionador de Jacobi.

Tamaño de malla	Número de incógnitas	CPU	GPU	Aceleración x
		C++ (s)	CUDA (s)	
30x30	1,800	0.007599	0.21234	0.035x
90x90	16,200	0.19308	0.85252	0.226x
150x150	45,000	0.98792	1.0727	0.9209x
250x250	125,000	4.3418	2.7515	1.570x
550x550	605,000	29.408	13.32	2.207x

En un código numérico desarrollado para la GPU sin procesamiento gráfico en tiempo real, los resultados siempre se tienen que transferir de la memoria de la GPU a la memoria de la CPU, para procesarlos más tarde. Este proceso también se tiene que cuantificar de alguna manera, ya que se tiene que llevar a cabo cada paso de tiempo o cuando el usuario así lo requiera. Por esta razón, la aceleración real deber medirse cuando todo el código numérico ha finalizado por completo. En la figura 5.39 a), se puede notar que la GPU es más lenta que la CPU cuando el problema se ejecuta con pocos nodos. Mientras que la GPU es más rápida si el número de nodos aumenta. La aceleración total obtenida se muestra en la figura 5.39 b), en ésta se nota que la mayor aceleración se logra con un tamaño de malla de 550x550 y tiene un valor de aproximadamente 3x, es decir, el tiempo total de cómputo para la CPU fue de 6.8 días, mientras que para la GPU el tiempo de cómputo fue de 2.26 días.

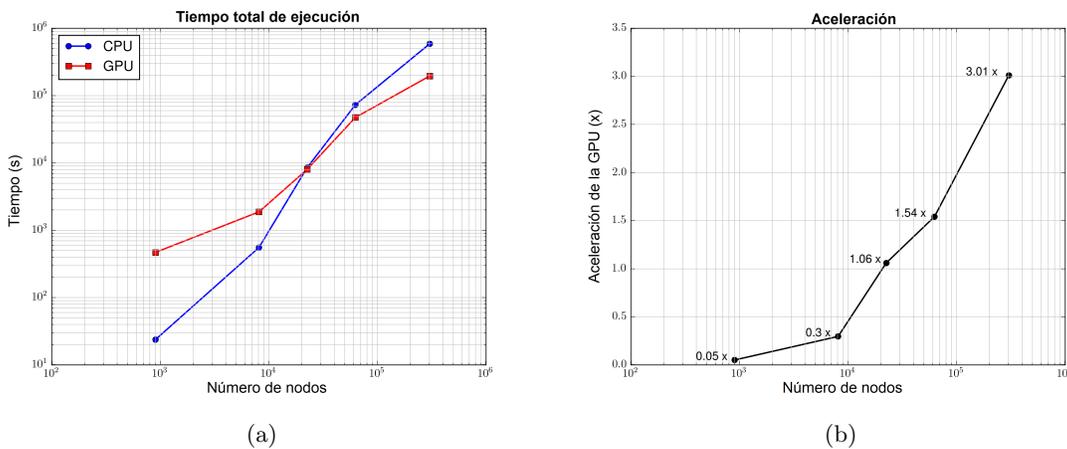


Figura 5.39: a) Tiempo de cómputo total de la CPU y la GPU, y b) aceleración lograda mediante el uso de la GPU.

Capítulo 6

Modelado numérico del proceso de inyección de vapor

En este capítulo, se presenta la metodología numérica-computacional desarrollada para simular el proceso de inyección de vapor. Al igual que en el capítulo anterior, la discretización de las ecuaciones de balance se lleva a cabo mediante el Método de Volumen Finito, la linealización y solución se realiza aplicando la técnica numérica de Newton-Raphson.

6.1. Modelo matemático del proceso térmico de inyección de vapor

El modelo matemático-numérico reportado por Shutler para simular el proceso de inyección de vapor considera los balances de masa de tres fases: aceite, agua y gas, y una ecuación de balance de energía [1]. Las consideraciones del modelo conceptual son las siguientes:

- Se consideran los efectos de la presión capilar.
- Se consideran los efectos de la gravedad.
- Se permite intercambio de masa en las fases agua y gas (vapor).
- El aceite es considerado no volátil, por lo tanto no se considera el fenómeno de destilación de aceite.

Tomando en cuenta la ecuación (3.12), sustituyendo α por la literal o para representar la fase aceite, la ecuación de conservación de masa para la fase aceite es,

$$\frac{\partial \phi \rho_o S_o}{\partial t} - \nabla \cdot \left(\frac{\rho_o \bar{k} k_{ro}}{\mu_o} (\nabla p_o - \rho_o g \nabla z) \right) = q_o. \quad (6.1)$$

Del mismo modo, sustituyendo α por la literal w , para la fase agua se obtiene:

$$\frac{\partial \phi \rho_w S_w}{\partial t} - \nabla \cdot \left(\frac{\rho_w \bar{k} k_{rw}}{\mu_w} (\nabla p_w - \rho_w \wp \nabla z) \right) = q_w + M_g, \quad (6.2)$$

donde el término fuente q_w se ha dividido en 2, el término de producción o inyección q_w más un término de condensación M_g que proviene de la condensación de vapor. La ecuación de conservación de masa para la fase gas es:

$$\frac{\partial \phi \rho_g S_g}{\partial t} - \nabla \cdot \left(\frac{\rho_g \bar{k} k_{rg}}{\mu_g} (\nabla p_g - \rho_g \wp \nabla z) \right) = q_g - M_g, \quad (6.3)$$

para este caso el término fuente q_g se ha dividido en 2, el término de producción o inyección q_g menos el término de condensación de vapor M_g .

Por otra parte, el medio poroso está lleno de fluido, consecuentemente la suma de las saturaciones de los fluidos en el medio es igual a uno. Esto es,

$$S_o + S_w + S_g = 1. \quad (6.4)$$

Otras relaciones constitutivas son:

$$p_w = p_o - p_{cow}, \quad (6.5)$$

$$p_g = p_o + p_{cog}, \quad (6.6)$$

$$p_{cow} = p_{cow}(S_w), \quad (6.7)$$

$$p_{cog} = p_{cog}(S_g). \quad (6.8)$$

Para la ecuación de energía se considera equilibrio térmico entre los fluidos dentro de los poros y la matriz sólida. La ecuación de balance de energía queda,

$$\begin{aligned} & \frac{\partial}{\partial t} [\phi (S_o \rho_o U_o + S_w \rho_w U_w + S_g \rho_g U_g) + (1 - \phi) (C_{p,r} \rho_r) T] = \\ & - \nabla \cdot [v_o \rho_o h_o + v_w \rho_w h_w + v_g \rho_g h_g] + \nabla \cdot \bar{k}_T \nabla T + \sum_{\alpha} q_{\alpha} - q_L. \end{aligned} \quad (6.9)$$

El sistema de ecuaciones diferenciales parciales (6.1), (6.2), (6.3) y (6.9), está sujeto a las ecuaciones constitutivas (6.4)-(6.8). En este sistema de ecuaciones se tienen 7 incógnitas las cuales son: S_o , S_w , S_g , p_o , p_w , p_g y T . Además, las ecuaciones son altamente no lineales y están acopladas. Entonces, para obtener una solución numérica del sistema de ecuaciones, se deben de linealizar y seleccionar 4 variables primarias adecuadas, de tal forma que con las ecuaciones constitutivas se cierre el sistema y se obtengan las incógnitas restantes. La selección de las variables primarias y la metodología de solución se lleva a cabo en las siguientes secciones.

6.2. Modelo numérico del proceso de inyección de vapor en coordenadas cartesianas

6.2.1. Planteamiento del algoritmo de solución

Para resolver el sistema de ecuaciones diferenciales parciales planteado, se ha seleccionado el algoritmo propuesto por Shutler [1].

La figura 6.1, muestra el diagrama de flujo que se utiliza para la implementación computacional de este algoritmo. En el primer ovalo mostrado en esta figura, se declaran e inicializan todas las variables y arreglos requeridos con los valores adecuados, esto incluye las condiciones iniciales y de frontera, valores petrofísicos, tamaño de la malla, paso de tiempo, etc. Posteriormente, siguiendo la flecha de flujo de ejecución, en el primer cuadro se resuelven las ecuaciones de balance de masa mediante la iteración de Newton-Raphson, manteniendo la temperatura constante. Aquí hay una diferencia importante con el algoritmo original, mientras que Shutler seleccionó como variables primarias las presiones p_o , p_{cow} y p_{cog} , en este trabajo se eligieron como variables primarias la presión de la fase aceite p_o y las saturaciones de las fases agua y vapor S_w y S_g , puesto que esto proporciona más generalidad al código, ya que en muchos casos de simulación de yacimientos se consideran despreciables los efectos de la presión capilar, por lo tanto, para estos casos se toma en cuenta una sola presión para las tres fases, lo cual no sería posible si se mantuvieran las variables primarias propuestas en el algoritmo original. Otra diferencia, radica en la discretización temporal de las ecuaciones de balance, la cual consiste en el uso de una discretización completamente implícita para este trabajo, mientras que Shutler utilizó una discretización semi-implícita. Continuando con el diagrama de flujo, una vez obtenidos los nuevos valores para las presiones y saturaciones p_α y S_α , se llega a un rombo de decisión, donde se comparan el cambio en la norma de la saturación de la fase vapor $|\delta S_g|$ con un valor prescrito ϵ , si éste es menor, se prosigue a resolver la ecuación de balance de energía y si es mayor se actualizan los valores de los parámetros dependientes de las variables primarias y se ejecuta otra iteración de Newton-Raphson.

Una vez se resuelven las ecuaciones de balance de masa, se continua hacia abajo siguiendo la flecha de flujo y se resuelve la ecuación de balance de energía, donde se aplica la iteración de Newton-Raphson para su solución. Para llevar a cabo la solución de esta ecuación, se mantienen todas las presiones p_α y saturaciones S_α constantes, se calcula el incremento δT , con el objetivo de obtener la temperatura en el siguiente paso de tiempo T^{n+1} . Una vez se ha actualizado la temperatura, se llega a un rombo de decisión, el cual compara la norma de la temperatura δT con un valor ϵ_T prescrito. Si δT , es mayor que ϵ_T , se actualizan los parámetros físicos dependientes de la temperatura y se ejecuta otra iteración de NR. Si δT , es menor que ϵ_T , y el tiempo total de simulación se ha alcanzado el algoritmo finaliza; pero, si el tiempo de simulación aún no se ha alcanzado, el flujo del algoritmo indica regresar y resolver nuevamente los balances de masa y repetir el ciclo de solución, esto hasta que el tiempo total de simulación sea alcanzado.

Una ventaja de resolver las ecuaciones de balance de masa y energía por separado, es que la ecuación de energía se puede “apagar”, esto permite resolver la inyección de agua de manera isotérmica, considerando $S_g = 0$ y colocando un término fuente para la inyección de la misma. Por otra parte, sí se considera la solución del balance de energía (manteniendo $S_g = 0$), se puede simular la inyección de agua caliente sin hacer grandes modificaciones al código numérico.

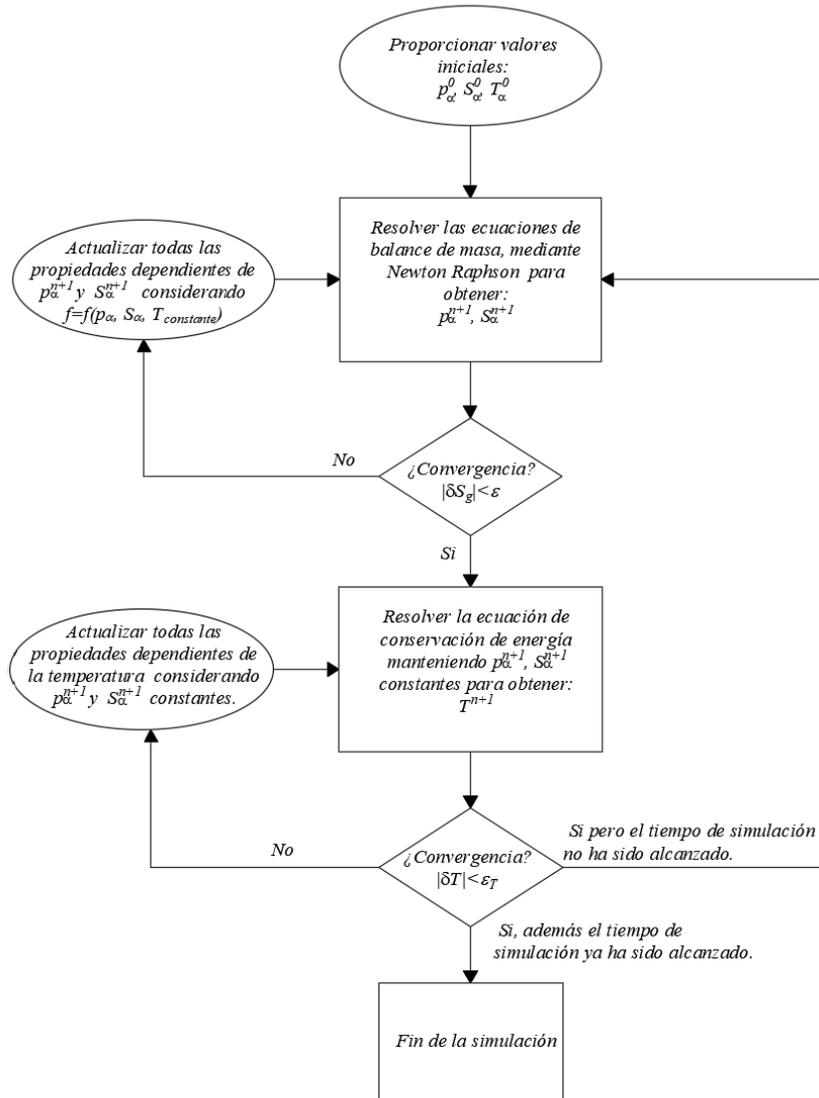


Figura 6.1: Diagrama de flujo para la implementación del código numérico del inyección de vapor.

6.2.2. Discretización de la ecuaciones

En esta subsección, se discretiza la ecuación de balance de masa de la fase aceite para el caso tridimensional. Integrando la ecuación (6.1) respecto al tiempo y a un volumen de control tridimensional, se obtiene:

$$\int_n^{n+1} \left\{ \int_{\Delta V} \left[\frac{\partial \phi \rho_o S_o}{\partial t} - \nabla \cdot \left[\frac{\rho_o \bar{k} k_{ro}}{\mu_o} (\nabla p_o - \rho_o \wp \nabla z) \right] - q_o \right] \Delta V \right\} dt = 0, \quad (6.10)$$

para evaluar la derivada temporal, se considera una aproximación de Euler hacia atrás:

$$\int_n^{n+1} \left(\frac{\partial \phi \rho_o S_o}{\partial t} \right) \Delta V dt = \left[\left(\phi \rho_o S_o \right)^{n+1} - \left(\phi \rho_o S_o \right)^n \right] \Delta V. \quad (6.11)$$

Evaluando los términos espaciales, se llega a:

$$\int_n^{n+1} \left\{ \int_{\Delta x} \left[- \frac{\partial}{\partial x} \frac{\rho_o k_{xx} k_{ro}}{\mu_o} \left(\frac{\partial p_o}{\partial x} - \rho_o \wp \frac{\partial z}{\partial x} \right) - \frac{\partial}{\partial y} \frac{\rho_o k_{yy} k_{ro}}{\mu_o} \left(\frac{\partial p_o}{\partial y} - \rho_o \wp \frac{\partial z}{\partial y} \right) - \frac{\partial}{\partial z} \frac{\rho_o k_{zz} k_{ro}}{\mu_o} \left(\frac{\partial p_o}{\partial z} - \rho_o \wp \frac{\partial z}{\partial z} \right) - q_o \right] \Delta V \right\} dt, \quad (6.12)$$

utilizando un esquema centrado para las derivadas:

$$\begin{aligned} & \int_n^{n+1} \left\{ - \left[\left[\frac{\rho_o k_{xx} k_{ro}}{\mu_o} \left(\frac{\partial p_o}{\partial x} - \rho_o \wp \frac{\partial z}{\partial x} \right) \right] \Big|_w^e \Delta y \Delta z + \left[\frac{\rho_o k_{yy} k_{ro}}{\mu_o} \left(\frac{\partial p_o}{\partial y} - \rho_o \wp \frac{\partial z}{\partial y} \right) \right] \Big|_s^n \Delta x \Delta z \right. \right. \\ & \quad \left. \left. + \left[\frac{\rho_o k_{zz} k_{ro}}{\mu_o} \left(\frac{\partial p_o}{\partial z} - \rho_o \wp \frac{\partial z}{\partial z} \right) \right] \Big|_b^t \Delta x \Delta y \right] - \frac{q_w}{\rho_o} \Delta V \right\} dt = \\ & \int_n^{n+1} \left\{ - \left[\left[\frac{\rho_o k_{xx} k_{ro}}{\mu_o} \right]_e \left(\frac{p_{o,E} - p_{o,P}}{\delta x_e} \right) - \frac{\rho_o k_{xx} k_{ro}}{\mu_o} \Big|_w \left(\frac{p_{o,P} - p_{o,W}}{\delta x_w} \right) \right] A_1 \right. \\ & \quad - \left[\frac{\rho_o k_{xx} k_{ro}}{\mu_o} \wp \rho_o \Big|_e \left(\frac{z_E - z_P}{\delta x_e} \right) - \frac{\rho_o k_{xx} k_{ro}}{\mu_o} \wp \rho_o \Big|_w \left(\frac{z_P - z_W}{\delta x_w} \right) \right] A_1 \\ & \quad + \left[\frac{\rho_o k_{yy} k_{ro}}{\mu_o} \Big|_n \left(\frac{p_{o,N} - p_{o,P}}{\delta y_n} \right) - \frac{\rho_o k_{yy} k_{ro}}{\mu_o} \Big|_s \left(\frac{p_{o,P} - p_{o,S}}{\delta y_s} \right) \right] A_2 \\ & \quad - \left[\frac{\rho_o k_{yy} k_{ro}}{\mu_o} \wp \rho_o \Big|_n \left(\frac{z_N - z_P}{\delta y_n} \right) - \frac{\rho_o k_{yy} k_{ro}}{\mu_o} \wp \rho_o \Big|_s \left(\frac{z_P - z_S}{\delta y_s} \right) \right] A_2 \\ & \quad + \left[\frac{\rho_o k_{zz} k_{ro}}{\mu_o} \Big|_t \left(\frac{p_{o,T} - p_{o,P}}{\delta z_t} \right) - \frac{\rho_o k_{zz} k_{ro}}{\mu_o} \Big|_b \left(\frac{p_{o,P} - p_{o,B}}{\delta z_b} \right) \right] A_3 \\ & \quad \left. - \left[\frac{\rho_o k_{zz} k_{ro}}{\mu_o} \wp \rho_o \Big|_t \left(\frac{z_T - z_P}{\delta z_t} \right) - \frac{\rho_o k_{zz} k_{ro}}{\mu_o} \wp \rho_o \Big|_b \left(\frac{z_P - z_B}{\delta z_b} \right) \right] A_3 \right] - q_o \Delta V \Big\} dt, \quad (6.13) \end{aligned}$$

evaluando la integral que denota tiempo con un esquema implícito:

$$\begin{aligned}
& \left\{ - \left[\frac{\rho_o k_{xx} k_{ro}}{\mu_o} \Big|_e \left(\frac{p_{o,E} - p_{o,P}}{\delta x_e} \right) - \frac{\rho_o k_{xx} k_{ro}}{\mu_o} \Big|_w \left(\frac{p_{o,P} - p_{o,W}}{\delta x_w} \right) \right] A_1 \right. \\
& + \left[\frac{\rho_o k_{xx} k_{ro}}{\mu_o} \wp \rho_o \Big|_e \left(\frac{z_E - z_P}{\delta x_e} \right) - \frac{\rho_o k_{xx} k_{ro}}{\mu_o} \wp \rho_o \Big|_w \left(\frac{z_P - z_W}{\delta x_w} \right) \right] A_1 \\
& - \left[\frac{\rho_o k_{yy} k_{ro}}{\mu_o} \Big|_n \left(\frac{p_{o,N} - p_{o,P}}{\delta y_n} \right) - \frac{\rho_o k_{yy} k_{ro}}{\mu_o} \Big|_s \left(\frac{p_{o,P} - p_{o,S}}{\delta y_s} \right) \right] A_2 \\
& + \left[\frac{\rho_o k_{yy} k_{ro}}{\mu_o} \wp \rho_o \Big|_n \left(\frac{z_N - z_P}{\delta y_n} \right) - \frac{\rho_o k_{yy} k_{ro}}{\mu_o} \wp \rho_o \Big|_s \left(\frac{z_P - z_S}{\delta y_s} \right) \right] A_2 \\
& - \left[\frac{\rho_o k_{zz} k_{ro}}{\mu_o} \Big|_t \left(\frac{p_{o,T} - p_{o,P}}{\delta z_t} \right) - \frac{\rho_o k_{zz} k_{ro}}{\mu_o} \Big|_b \left(\frac{p_{o,P} - p_{o,B}}{\delta z_b} \right) \right] A_3 \\
& + \left. \left[\frac{\rho_o k_{zz} k_{ro}}{\mu_o} \wp \rho_o \Big|_t \left(\frac{z_T - z_P}{\delta z_t} \right) - \frac{\rho_o k_{zz} k_{ro}}{\mu_o} \wp \rho_o \Big|_b \left(\frac{z_P - z_B}{\delta z_b} \right) \right] A_3 - \frac{q_w \Delta V}{\rho_o} \right\}^{n+1} \Delta t = \\
& \left\{ - \left[T_{o,e} \left(p_{o,E} - p_{o,P} \right) - T_{o,w} \left(p_{o,P} - p_{o,W} \right) \right] \right. \\
& + \left[T_{o,e} \gamma_o \Big|_e \left(z_E - z_P \right) - T_{o,w} \gamma_o \Big|_w \left(z_P - z_W \right) \right] \\
& - \left[T_{o,n} \left(p_{o,N} - p_{o,P} \right) - T_{o,s} \left(p_{o,P} - p_{o,S} \right) \right] \\
& + \left[T_{o,n} \gamma_o \Big|_n \left(z_N - z_P \right) - T_{o,s} \gamma_o \Big|_s \left(z_P - z_S \right) \right] \\
& - \left[T_{o,t} \left(p_{o,T} - p_{o,P} \right) - T_{o,b} \left(p_{o,P} - p_{o,B} \right) \right] \\
& + \left. \left[T_{o,t} \gamma_o \Big|_t \left(z_T - z_P \right) - T_{o,b} \gamma_o \Big|_b \left(z_P - z_B \right) \right] - q_{o,P} \Delta V \right\}^{n+1} \Delta t. \tag{6.14}
\end{aligned}$$

Sumando las ecuaciones (6.11) y (6.14), y reorganizando términos, se obtiene la discretización tridimensional:

$$\begin{aligned}
& \frac{\Delta V}{\Delta t} \left[\left(\phi \rho_o S_o \right) - \left(\phi \rho_o S_o \right) \Big|_P^n \right] = \\
& \left[T_{o,e} \left(p_{o,E} - p_{o,P} \right) - T_{o,w} \left(p_{o,P} - p_{o,W} \right) \right] - \left[T_{o,e} \gamma_{o,e} \left(z_E - z_P \right) - T_{o,w} \gamma_{o,w} \left(z_P - z_W \right) \right] \\
& + \left[T_{o,n} \left(p_{o,N} - p_{o,P} \right) - T_{o,s} \left(p_{o,P} - p_{o,S} \right) \right] - \left[T_{o,n} \gamma_{o,n} \left(z_N - z_P \right) - T_{o,s} \gamma_{o,s} \left(z_P - z_S \right) \right] \\
& + \left[T_{o,t} \left(p_{o,T} - p_{o,P} \right) - T_{o,b} \left(p_{o,P} - p_{o,B} \right) \right] - \left[T_{o,t} \gamma_{o,t} \left(z_T - z_P \right) - T_{o,b} \gamma_{o,b} \left(z_P - z_B \right) \right] \\
& + q_{o,P} \Delta V, \tag{6.15}
\end{aligned}$$

donde el subíndice $n + 1$ se omite por simplicidad. Siguiendo la metodología presentada para discretizar la ecuación de balance de masa de la fase agua (6.2), se llega a:

$$\begin{aligned}
& \frac{\Delta V}{\Delta t} \left[\left(\phi \rho_w S_w \right) - \left(\phi \rho_w S_w \right) \Big|_P^n \right] = \\
& \left[T_{w,e} \left(p_{w,E} - p_{w,P} \right) - T_{w,w} \left(p_{w,P} - p_{w,W} \right) \right] - \left[T_{w,e} \gamma_{w,e} \left(z_E - z_P \right) - T_{w,w} \gamma_{w,w} \left(z_P - z_W \right) \right] \\
& + \left[T_{w,n} \left(p_{w,N} - p_{w,P} \right) - T_{w,s} \left(p_{w,P} - p_{w,S} \right) \right] - \left[T_{w,n} \gamma_{w,n} \left(z_N - z_P \right) - T_{w,s} \gamma_{w,s} \left(z_P - z_S \right) \right] \\
& + \left[T_{w,t} \left(p_{w,T} - p_{w,P} \right) - T_{w,b} \left(p_{w,P} - p_{w,B} \right) \right] - \left[T_{w,t} \gamma_{w,t} \left(z_T - z_P \right) - T_{w,b} \gamma_{w,b} \left(z_P - z_B \right) \right] \\
& + q_{w,P} \Delta V - Mg \Delta V, \tag{6.16}
\end{aligned}$$

para la fase gas o vapor, se obtiene:

$$\begin{aligned}
\frac{\Delta V}{\Delta t} \left[\left(\phi \rho_g S_g \right) - \left(\phi \rho_g S_g \right)^n \right]_P = & \\
& \left[T_{g,e} \left(p_{g,E} - p_{g,P} \right) - T_{g,w} \left(p_{g,P} - p_{g,W} \right) \right] - \left[T_{g,e} \gamma_{g,e} \left(z_E - z_P \right) - T_{g,w} \gamma_{g,w} \left(z_P - z_W \right) \right] \\
& + \left[T_{g,n} \left(p_{g,N} - p_{g,P} \right) - T_{g,s} \left(p_{g,P} - p_{g,S} \right) \right] - \left[T_{g,n} \gamma_{g,n} \left(z_N - z_P \right) - T_{g,s} \gamma_{g,s} \left(z_P - z_S \right) \right] \\
& + \left[T_{g,t} \left(p_{g,T} - p_{g,P} \right) - T_{g,b} \left(p_{g,P} - p_{g,B} \right) \right] - \left[T_{g,t} \gamma_{g,t} \left(z_T - z_P \right) - T_{g,b} \gamma_{g,b} \left(z_P - z_B \right) \right] \\
& + q_{g,P} \Delta V - Mg \Delta V, \tag{6.17}
\end{aligned}$$

por último para la ecuación del balance de energía, se obtiene:

$$\begin{aligned}
\frac{\Delta V}{\Delta t} \left\{ \left[\phi \left(S_o \rho_o U_o + S_w \rho_w U_w + S_g \rho_g U_g \right) + (1 - \phi) \left(C_{p,r} \rho_r T \right) \right] - \left[\phi \left(S_o \rho_o h_o + S_w \rho_w h_w + S_g \rho_g h_g \right) + (1 - \phi) \left(\rho c \right)_R T \right]^n \right\} \\
= \left[T_{ho,e} \left(p_{o,E} - p_{o,P} \right) - T_{ho,w} \left(p_{o,P} - p_{o,W} \right) \right] + \left[T_{ho,n} \left(p_{o,N} - p_{o,P} \right) - T_{ho,s} \left(p_{o,P} - p_{o,S} \right) \right] \\
+ \left[T_{ho,t} \left(p_{o,T} - p_{o,P} \right) - T_{ho,b} \left(p_{o,P} - p_{o,B} \right) \right] + \left[T_{hw,e} \left(p_{w,E} - p_{w,P} \right) - T_{hw,w} \left(p_{w,P} - p_{w,W} \right) \right] \\
+ \left[T_{hw,n} \left(p_{w,N} - p_{w,P} \right) - T_{hw,s} \left(p_{w,P} - p_{w,S} \right) \right] + \left[T_{hw,t} \left(p_{w,T} - p_{w,P} \right) - T_{hw,b} \left(p_{w,P} - p_{w,B} \right) \right] \\
+ \left[T_{hg,e} \left(p_{g,E} - p_{g,P} \right) - T_{hg,w} \left(p_{g,P} - p_{g,W} \right) \right] + \left[T_{hg,n} \left(p_{g,N} - p_{g,P} \right) - T_{hg,s} \left(p_{g,P} - p_{g,S} \right) \right] \\
+ \left[T_{hg,t} \left(p_{g,T} - p_{g,P} \right) - T_{hg,b} \left(p_{g,P} - p_{g,B} \right) \right] \\
- \left[\left(T_{hg,e} \gamma_{g,e} + T_{hw,e} \gamma_{w,e} + T_{ho,e} \gamma_{o,e} \right) \left(z_E - z_P \right) - \left(T_{hg,w} \gamma_{g,w} + T_{hw,w} \gamma_{w,w} + T_{ho,w} \gamma_{o,w} \right) \left(z_P - z_W \right) \right] \\
- \left[\left(T_{hg,n} \gamma_{g,n} + T_{hw,n} \gamma_{w,n} + T_{ho,n} \gamma_{o,n} \right) \left(z_N - z_P \right) - \left(T_{hg,s} \gamma_{g,s} + T_{hw,s} \gamma_{w,s} + T_{ho,s} \gamma_{o,s} \right) \left(z_P - z_S \right) \right] \\
- \left[\left(T_{hg,t} \gamma_{g,t} + T_{hw,t} \gamma_{w,t} + T_{ho,t} \gamma_{o,t} \right) \left(z_T - z_P \right) - \left(T_{hg,b} \gamma_{g,b} + T_{hw,b} \gamma_{w,b} + T_{ho,b} \gamma_{o,b} \right) \left(z_P - z_B \right) \right] \\
+ \left[T_{k,e} \left(T_E - T_P \right) - T_{k,w} \left(T_P - T_W \right) \right] + \left[T_{k,n} \left(T_N - T_P \right) - T_{k,s} \left(T_P - T_S \right) \right] \\
+ \left[T_{k,t} \left(T_T - T_P \right) - T_{k,b} \left(T_P - T_B \right) \right] + q_{hg,P} \Delta V + q_{hw,P} - q_L \Delta V. \tag{6.18}
\end{aligned}$$

donde la transmisibilidad $T_{h\alpha}$ involucra la entalpía de la fase α , por ejemplo, $T_{ho,e} = \frac{h_o \rho_o k_{xx} k_{ro} A_1}{\mu_o \delta x_e}$. Además, se define una transmisibilidad T_k , la cual contiene el coeficiente de difusión térmico k_T , i.e., $T_{k,e} = \frac{k_{T,xx} A_1}{\delta x_e}$.

Las ecuaciones discretas (6.15), (6.16), (6.17) y (6.18) son altamente no lineales, por lo que se requiere una metodología numérica de linealización y solución.

6.2.3. Metodología de linealización y solución de los sistemas de ecuaciones diferenciales

Se utiliza el Método de Newton-Rapshon para resolver ambos balances de masa y energía. Como primer paso en el algoritmo de solución se resuelve el balance de masa, en el cual se han seleccionado como variables primarias la presión del aceite p_o^{n+1} , la saturación de la fase agua S_w^{n+1} y la saturación de la fase gas S_g^{n+1} .

Por lo tanto, las incógnitas del sistema de ecuaciones son los incrementos δp_o^{n+1} , δS_w^{n+1} and δS_g^{n+1} , el cual puede escribirse como sigue:

$$\begin{pmatrix} \frac{\partial R_o^k}{\partial p_o} & \frac{\partial R_o^k}{\partial S_w} & \frac{\partial R_o^k}{\partial S_g} \\ \frac{\partial R_w^k}{\partial p_o} & \frac{\partial R_w^k}{\partial S_w} & \frac{\partial R_w^k}{\partial S_g} \\ \frac{\partial R_g^k}{\partial p_o} & \frac{\partial R_g^k}{\partial S_w} & \frac{\partial R_g^k}{\partial S_g} \end{pmatrix} \begin{pmatrix} \delta p_o^{k+1} \\ \delta S_w^{k+1} \\ \delta S_g^{k+1} \end{pmatrix} = \begin{pmatrix} -R_o^k \\ -R_w^k \\ -R_g^k \end{pmatrix}. \quad (6.19)$$

Evaluando las componentes del Jacobiano para los residuales R_o , R_w y R_g , se llega a:

$$\begin{aligned} & \frac{\partial R_o(p_o, S_w, S_g)}{\partial P_{o,B}} \Big|_k \delta p_{o,T}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial P_{o,S}} \Big|_k \delta p_{o,S}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial p_{o,W}} \Big|_k \delta p_{o,W}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial p_{o,P}} \Big|_k \delta p_{o,P}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w, S_g)}{\partial p_{o,E}} \Big|_k \delta p_{o,E}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial p_{o,N}} \Big|_k \delta p_{o,N}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial p_{o,T}} \Big|_k \delta p_{o,T}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,B}} \Big|_k \delta S_{w,T}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,S}} \Big|_k \delta S_{w,S}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,W}} \Big|_k \delta S_{w,W}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,P}} \Big|_k \delta S_{w,P}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,E}} \Big|_k \delta S_{w,E}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,N}} \Big|_k \delta S_{w,N}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,T}} \Big|_k \delta S_{w,T}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,B}} \Big|_k \delta S_{g,T}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,S}} \Big|_k \delta S_{g,S}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,W}} \Big|_k \delta S_{g,W}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,P}} \Big|_k \delta S_{g,P}^{k+1} + \\ & \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,E}} \Big|_k \delta S_{g,E}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,N}} \Big|_k \delta S_{g,N}^{k+1} + \frac{\partial R_o(p_o, S_w, S_g)}{\partial S_{w,T}} \Big|_k \delta S_{g,T}^{k+1} = -R_o, \end{aligned} \quad (6.20)$$

$$\begin{aligned} & \frac{\partial R_w(p_o, S_w, S_g)}{\partial P_{o,B}} \Big|_k \delta p_{o,T}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial P_{o,S}} \Big|_k \delta p_{o,S}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial p_{o,W}} \Big|_k \delta p_{o,W}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial p_{o,P}} \Big|_k \delta p_{o,P}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w, S_g)}{\partial p_{o,E}} \Big|_k \delta p_{o,E}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial p_{o,N}} \Big|_k \delta p_{o,N}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial p_{o,T}} \Big|_k \delta p_{o,T}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,B}} \Big|_k \delta S_{w,T}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,S}} \Big|_k \delta S_{w,S}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,W}} \Big|_k \delta S_{w,W}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,P}} \Big|_k \delta S_{w,P}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,E}} \Big|_k \delta S_{w,E}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,N}} \Big|_k \delta S_{w,N}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,T}} \Big|_k \delta S_{w,T}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,B}} \Big|_k \delta S_{g,T}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,S}} \Big|_k \delta S_{g,S}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,W}} \Big|_k \delta S_{g,W}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,P}} \Big|_k \delta S_{g,P}^{k+1} + \\ & \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,E}} \Big|_k \delta S_{g,E}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,N}} \Big|_k \delta S_{g,N}^{k+1} + \frac{\partial R_w(p_o, S_w, S_g)}{\partial S_{w,T}} \Big|_k \delta S_{g,T}^{k+1} = -R_w, \end{aligned} \quad (6.21)$$

$$\left(\frac{\partial R_T^k}{\partial T} \right) \begin{bmatrix} \delta T_{T,1}^{k+1} \\ \delta T_{T,2}^{k+1} \\ \delta T_{T,3}^{k+1} \\ \vdots \\ \delta T_{T,l-2}^{k+1} \\ \delta T_{T,l-1}^{k+1} \\ \delta T_{T,l}^k \end{bmatrix} = \begin{bmatrix} -R_{T,1}^k \\ -R_{T,2}^k \\ -R_{T,3}^k \\ \vdots \\ -R_{T,l-2}^k \\ -R_{T,l-1}^k \\ -R_{T,l}^k \end{bmatrix}$$

Figura 6.3: Sistema de ecuaciones lineales para resolver el balance de energía con la variable primaria T .

Los sistemas de ecuaciones mostrados en las figuras 6.2 y 6.3, son la parte medular del algoritmo numérico para la simulación del proceso de inyección vapor, y su solución se lleva a cabo de manera secuencial según lo descrito en la sección previa.

6.2.4. Implementación computacional

Implementación del código numérico en el CPU

La implementación en la CPU se llevo a cabo siguiendo el pseudo-código mostrado en la figura 6.4. Además, se utilizan las mismas herramientas de programación que se utilizaron para el código numérico de inyección de agua, es decir, *C++* como lenguaje de programación, las bibliotecas de álgebra lineal *Eigen* y *Cusp 5.1*, y se utilizó *Matplotlib* para la generación de las gráficas.

```

01 /* Inicialización de todas las variables y datos iniciales,
02 definición de todos los arreglos y matrices
03 para construir el sistema de ecuaciones lineales */
04  WHILE (Time_step <= Total_time) DO
05      WHILE ( Delta_Sg < epsilon_g & IterationM < Max_iterM) DO
06          Calcular las componentes del Jacobiano referentes a la fase aceite y Ro
07          Calcular las componentes del Jacobiano referentes a la fase agua y Rw
08          Calcular las componentes del Jacobiano referentes a la fase gas y Rg
09          Construir el Jacobiano usando el formato CRS
10          Resolver el sistema de ecuaciones utilizando el método BICGSTAB o GMRES
11      END WHILE
12  Actualizar las variables primarias de los balances de masa
13      WHILE ( Delta_T < epsilon_e & IterationE < Max_iterE) DO
14          Calcular las componentes del Jacobiano referentes a la temperatura y RT
15          Construir el Jacobiano usando el formato CRS
16          Resolver el sistema de ecuaciones utilizando el método BICGSTAB o GMRES
17      END WHILE
18  IF (!(Time_step % Frequency)) THEN
19      Salvar o imprimir las variables primarias (So, Sw, Sg y T)
20  END IF
21  END WHILE
22 /* Liberar memoria y finalizar la simulación */

```

Figura 6.4: Pseudocódigo general del código numérico de inyección de vapor

En las primeras tres líneas del pseudocódigo mostrado en la figura 6.4, se declaran e inicializan todas las variables y arreglos requeridos con los valores adecuados, esto incluye las condiciones iniciales y de frontera, valores petrofísicos, tamaño de la malla, paso de tiempo, etc. En la línea 4 se inicia la simulación y se lleva a cabo dentro del ciclo *while*, hasta que se alcanza el número total de pasos de tiempo definido por el usuario. Dentro de este ciclo general, existen otros dos ciclos *while*, en los cuales se implementa el algoritmo de NR para la solución de los balances de masa y energía, respectivamente. El primer ciclo anidado que comienza en la línea 5, corresponde a la solución de los balances de masa, y éste se lleva a cabo hasta que la norma del cambio de la saturación de vapor ($|\delta S_g|$), sea menor que un valor prescrito (ϵ_g), o se alcance el número máximo de iteraciones prescrito del algoritmo NR. Las líneas 6 a 10, representan los pasos principales para construir el Jacobiano y resolver el sistema de ecuaciones lineales. En las líneas 6-8, se calculan las componentes de Jacobiano, esto significa calcular las componentes discretizadas de los residuos mediante las ecuaciones (6.15)-

(6.17) y sus correspondientes derivadas (6.20)-(6.22). Se debe aclarar que todas las derivadas, se calculan numéricamente utilizando diferencias finitas de primer orden. Luego, en la línea 9, el Jacobiano se construye utilizando el formato de almacenamiento de filas comprimidas CRS. En la línea 10, se resuelve el sistema lineal de ecuaciones utilizando el método BICGSTAB o GMRES, los cuales están contenidos en la biblioteca EIGEN. Una vez que el algoritmo NR converge, la línea 12 indica la actualización de las variables primarias de los balances de masa. En las líneas 13 a 17, se resuelve la ecuación de balance de energía tomando en cuenta las saturaciones y presiones de las fases constantes constantes. Aquí el ciclo se ejecuta hasta que la norma del cambio en la temperatura ($|\delta T|$), sea menor que un valor prescrito (ϵ_T), o el número de iteraciones de NR no exceda el valor 3 (véase las referencias [1], [5], [6] y [7]). En la línea 14 se calculan el residual y las componentes del Jacobiano pertenecientes al balance de energía, mediante las ecuaciones (6.18) y (6.23), respectivamente. En las líneas 15 y 16, se construye y se resuelve el sistema de ecuaciones lineales pertenecientes a este balance. Posteriormente, en la línea 19 se guardan los resultados (variables primarias) cada vez que la variable `iteration` es divisible por la variable `Frecuency` prescrita. Estos ciclos se repiten hasta que el tiempo de simulación total definido por el usuario se alcance. Finalmente, en la línea 22 se libera la memoria utilizada por el código numérico.

Implementación del código numérico en la GPU

Para la implementación en la GPU se utiliza CUDA como lenguaje estándar de programación, y la biblioteca CUSP como auxiliar en las operaciones matemáticas que requieran álgebra lineal. La implementación numérica está casi totalmente realizada en la GPU. El pseudocódigo del algoritmo principal para la implementación de la GPU es similar al que se muestra en la figura 6.4. Se debe tener en cuenta que las principales diferencias son: a) el ciclo que comienza en la línea 4 requiere todas las variables y arreglos definidos en las líneas 1-3, por lo tanto, un primer intercambio de información se realiza desde la CPU a la GPU antes de iniciar el ciclo de solución, sin embargo, esta inicialización lleva un tiempo de cómputo mínimo debido al hecho de que las matrices o arreglos más grandes se construyen directamente en la GPU; b) las operaciones en los ciclos de solución de los balances de masa y energía están todas codificadas en CUDA; c) la operación de la línea 19 requiere una transferencia de información de la GPU a la CPU; cabe señalar que esto sólo se hace cada vez que la variable `iteration` es divisible por la variable `Frecuency`.

La implementación de las funciones kernel en la GPU y el involucramiento de los vectores para construir el sistema lineal de ecuaciones, se lleva a cabo de manera similar que para el código de inyección de agua, por lo que no se entra en más detalles para esta sección.

6.3. Resultados en coordenadas cartesianas para la simulación del proceso de inyección de vapor

6.3.1. Validación del código numérico de inyección de vapor

Los resultados experimentales reportados por Willman *et al*, son uno de los trabajos más utilizados para la validación de los códigos numéricos de inyección de vapor [35]. El primero en validar su código numérico con los resultados experimentales mencionados fue Shutler en 1970, posteriormente estos resultados serían adoptados como un benchmark por Coats [7], Sarathi [10], Mozaffari [34] entre otros.

El modelo conceptual que se presenta a continuación está basado en el modelo reportado por Shutler para reproducir los resultados experimentales por Willman *et al*. Las consideraciones para el modelo conceptual son:

- El flujo de fluidos trifásico es unidimensional y ocurre en un núcleo homogéneo de 3 *ft* de longitud por 0.5 *ft* de diámetro.
- El medio esta inicialmente saturado con aceite y agua.
- Se inyecta vapor saturado a presión y temperatura constante.
- El único gas presente en la simulación es vapor de agua.

El modelo conceptual representa un segmento de núcleo de 3 pies de longitud (empacado de arena de forma cilíndrica), el cual inicialmente esta saturado con agua y aceite en su interior, en el que se inyecta vapor a presión y temperatura constante por la cara izquierda del núcleo con el objetivo de desplazar el aceite, se producen los tres fluidos en cuestión por el otro extremo del núcleo a presión constante. La figura 6.5, muestra un esquema del modelo físico-conceptual aquí descrito.

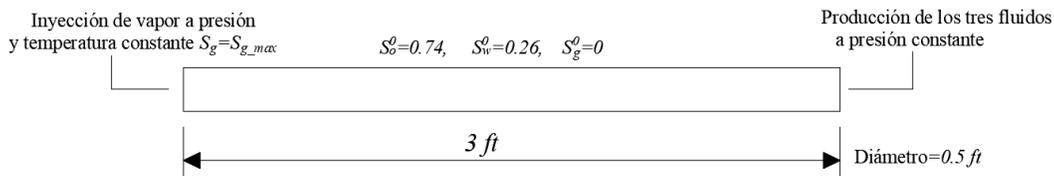


Figura 6.5: Modelo físico para la simulación de inyección de vapor a escala de núcleo.

Los parámetros para la solución numérica del problema se presentan en la tabla 6.1. Para el calculo de las propiedades físicas del agua y vapor, tales como densidad, viscosidad y entalpía las correlaciones utilizadas se encuentran en el Apéndice B. Además, para el calculo de la temperatura de saturación de vapor se utilizaron las tablas de vapor proporcionadas en el texto *Practical heavy oil recovery* [53].

Otro punto importante, es el cálculo de la permeabilidad relativa del aceite k_{ro} en un sistema trifásico, la cual se obtiene mediante el Modelo de Stone's II a partir de las permeabilidades relativas de los sistemas agua-aceite ($k_{ro_{ow}}$) y gas-aceite ($k_{ro_{og}}$), estas permeabilidades están dadas en la tabla 6.2. Cabe aclarar que para las permeabilidades relativas del agua y del vapor, se considera que solamente son función de sus respectivas saturaciones.

Tabla 6.1: Parámetros para simular la inyección de vapor en un núcleo saturado inicialmente de agua y aceite.

Propiedad	Valor
Permeabilidad (k_{11})	856 (mD)
Porosidad (ϕ)	0.262
Longitud (L_x)	3.0 (ft)
Diámetro del núcleo	0.5 (ft)
Conductividad térmica de la roca más el fluido (k_T)	100.0 (BTU/(ft day °F))
Conductividad térmica de sobrecarga ($k_{T,Ob}$)	60.0 (BTU/(ft day °F))
Densidad de la roca por su calor específico ($\rho_R C_R$)	31.0 (BTU/(ft ³ °F))
Densidad de la sobrecarga por su calor específico ($\rho_{Ob} C_{Ob}$)	6.0 (BTU/(ft ³ °F))
Densidad estándar del aceite ($\rho_{o,std}$)	53.8 (lbm/ft ³)
Temperatura inicial (T^0)	80.0 (°F)
Temperatura de inyección (T^{inj})	330.0 (°F)
Presión de inyección (p_g^{inj})	106.0 (psi)
Presión de producción (p_g^{prod})	68.0 (psi)
Saturación inicial de agua (S_w)	0.26
Saturación inicial de vapor (S_g)	0.0
Calidad de vapor	1.0

(Datos tomados de [1].)

Tabla 6.2: Permeabilidades relativas para los sistemas agua-aceite y gas-aceite.

$S_{w_{ow}}$	$k_{rw_{ow}}$	$k_{ro_{ow}}$	p_{cow}	$S_{o_{og}}$	$k_{rg_{og}}$	$k_{ro_{og}}$	p_{cog}
0.2600	0.0000	0.7000	2.2000	0.3000	0.1750	0.0000	0.3800
0.3000	0.0010	0.4750	1.0000	0.4000	0.1050	0.0100	0.2800
0.4000	0.0150	0.1950	0.7000	0.5000	0.0500	0.0400	0.2100
0.5000	0.0500	0.1220	0.5200	0.6000	0.0100	0.1250	0.1600
0.6000	0.1150	0.0750	0.3700	0.7000	0.0010	0.3800	0.1200
0.7000	0.2050	0.0000	0.2300	0.7300	0.0000	0.7000	0.1100

(Datos tomados de [1].)

Shutler validó sus resultados con el perfil de temperatura y con el aceite producido reportados en el trabajo de Willman *et al.* Como se mencionó anteriormente, los resultados experimentales de Willman han servido como benchmark para varios autores, por tal motivo

también se comparan los resultados obtenidos de esta implementación contra los resultados numéricos reportados por Mozaffari en el año 2013 [34].

Para llevar a cabo esta simulación se consideraron sólo 10 volúmenes de control. Esta cantidad de volúmenes o nodos internos es la utilizada por los autores para este problema en particular. Por otro lado, para calcular las derivadas dentro de los bloques del Jacobiano correspondiente al balance de masa, se utilizaron incrementos de 1×10^{-7} para las saturaciones S_w y S_g , y para la presión p_o el incremento para evaluar las derivadas fue de 1×10^{-5} . Además, el criterio de paro para salir del primer ciclo de NR se seleccionó a: $|\delta S_g| < 1 \times 10^{-4}$. Para resolver el balance de energía, se seleccionó un incremento de 1×10^{-5} , para calcular las derivadas con respecto a la temperatura y un criterio de paro de $|\delta T| < 0.1$. Estos parámetros numéricos se determinaron empíricamente, tomando en cuenta la estabilidad y convergencia del código numérico desarrollado.

En la figura 6.6, se presenta el perfil de temperatura obtenido para 0.55 volúmenes porosos (VP) de fluidos producidos. Cabe señalar que un volumen poroso, se refiere al volumen total del núcleo considerado como un cilindro sólido, multiplicado por su porosidad, es decir, el volumen total que pueden ocupar los fluidos en el medio poroso. Para entender mejor esta figura, se debe señalar que los resultados reportados por Shutler se ajustan idealmente al perfil de temperatura medido experimentalmente por Willman, por lo tanto los perfiles son los mismos. También, en esta figura, se puede apreciar que el perfil reportado por Mozaffari (línea punteada de color negro) tiene una forma más suavizada y comienza a declinar antes que el perfil de Shutler (línea punteada de color azul), mientras que el perfil obtenido en este trabajo se asemeja más al perfil de Shutler (línea continua de color verde), por lo tanto se ajusta más al perfil experimental. Las diferencias entre los perfiles pueden deberse a las diferencias entre los algoritmos y los parámetros numéricos utilizados.

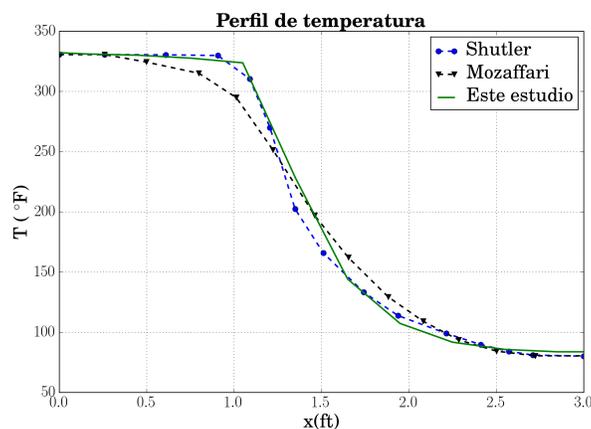


Figura 6.6: Comparación del perfil de temperatura a escala de núcleo.

Otro punto importante para la validación del código es comparar las curvas de producción de aceite, las cuales se muestran en la figura 6.7. En esta figura, se nota que los resultados de producción de aceite reportados por Shutler (línea punteada de color azul) difieren un poco de los reportados experimentalmente por Willman (línea punteada de color rojo); también, se aprecia que la curva reportada por Mozaffari (línea punteada de color negro) es la que más se desvía de los resultados experimentales. Por otra parte, los resultados obtenidos con el código numérico desarrollado en este trabajo se presentan en la línea continua de color verde, en ésta se puede observar una gran similitud con los resultados numéricos y experimentales reportados por Shutler y Willman, respectivamente.

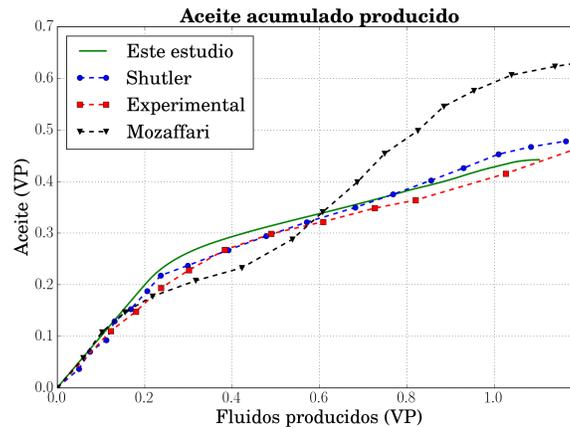


Figura 6.7: Comparación del aceite producido en el núcleo en volúmenes porosos (VP).

También, se visualizan los perfiles de saturación y temperatura para analizar la dinámica del desplazamiento, los cuales se muestran en la figura 6.8 para los distintos tiempos de simulación seleccionados. En la figura 6.8 a), se muestran los perfiles de saturación de aceite S_o , ahí se nota que conforme avanza el tiempo de simulación el aceite desplazado es mayor; sin embargo el barrido agua-vapor en el último tiempo reportado, dejó un banco de aceite el cual se encuentra en la ventana de saturación que va de 0.41 a 0.3 aproximadamente. La figura 6.8 b), presenta los perfiles de saturación de agua S_w , en la cual se nota que el agua se produce desde el primer tiempo visualizado (0.04 días), esto es un indicio de que la condensación de vapor ocurre inmediatamente después del inicio de la inyección de vapor. Adicionalmente, en la misma figura, se aprecia que el pistón de agua avanza hasta que prácticamente desaparece del medio. En la figura 6.8 c), se muestran los perfiles de saturación de vapor S_g , los cuales tienen una estrecha relación con los perfiles de temperatura mostrados en la figura 6.8 d), debido a que el medio poroso tiene que igualar o superar la temperatura de saturación de vapor para que el vapor fluya libremente sin condensarse. Esta restricción física, provoca que el aceite sea desplazado por un pistón de agua caliente seguido por un pistón de vapor. Observando la figura 6.8, se puede notar que los perfiles de saturación y temperatura corresponden muy bien al comportamiento físico del problema descrito en el

Capítulo 2.

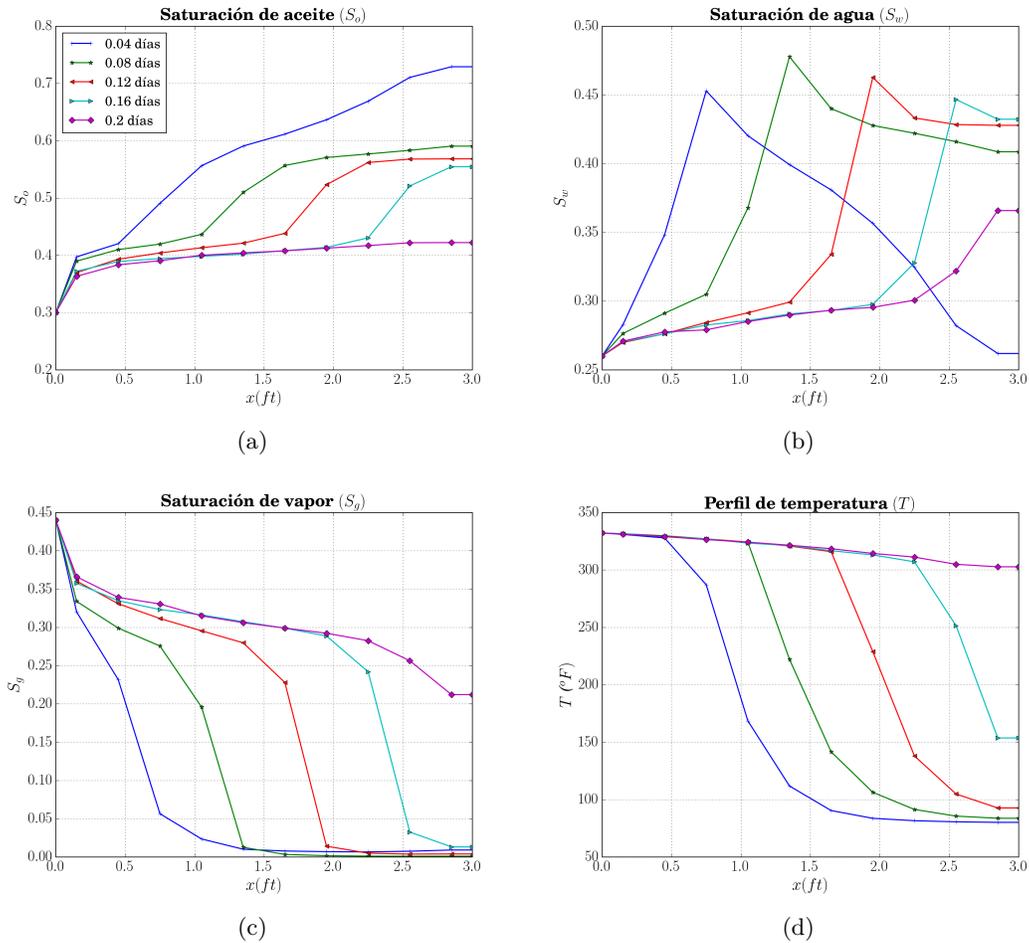


Figura 6.8: Perfiles obtenidos para la simulación de inyección de vapor en un núcleo utilizando 10 volúmenes de control.

Se ejecutó la misma simulación con los mismos parámetros numéricos, considerando un tamaño de malla de 30 volúmenes de control, esto con el objetivo de comprobar la tendencia de la solución numérica. En la figura 6.9, se muestran los resultados obtenidos de los perfiles de las variables primarias, la curva de aceite producido y la curva de vapor producido. Comparando el perfil de saturación de aceite S_o para las simulaciones de 10 y 30 volúmenes se puede notar que tienen la misma tendencia cualitativa y en ambos existe un banco de aceite que no ha sido desplazado. Este resultado se corresponde con la curva de aceite producido, la cual se muestra en la figura 6.9 e). Los datos numéricos obtenidos para el aceite producido en volúmenes porosos son 0.4479 y 0.4536, para 10 y 30 volúmenes, respectivamente. Tomando en cuenta estos parámetros de producción, la diferencia relativa de las curvas es de 1.25 %, por lo que se considera que el problema esta bien resuelto con el tamaño de malla seleccionado.

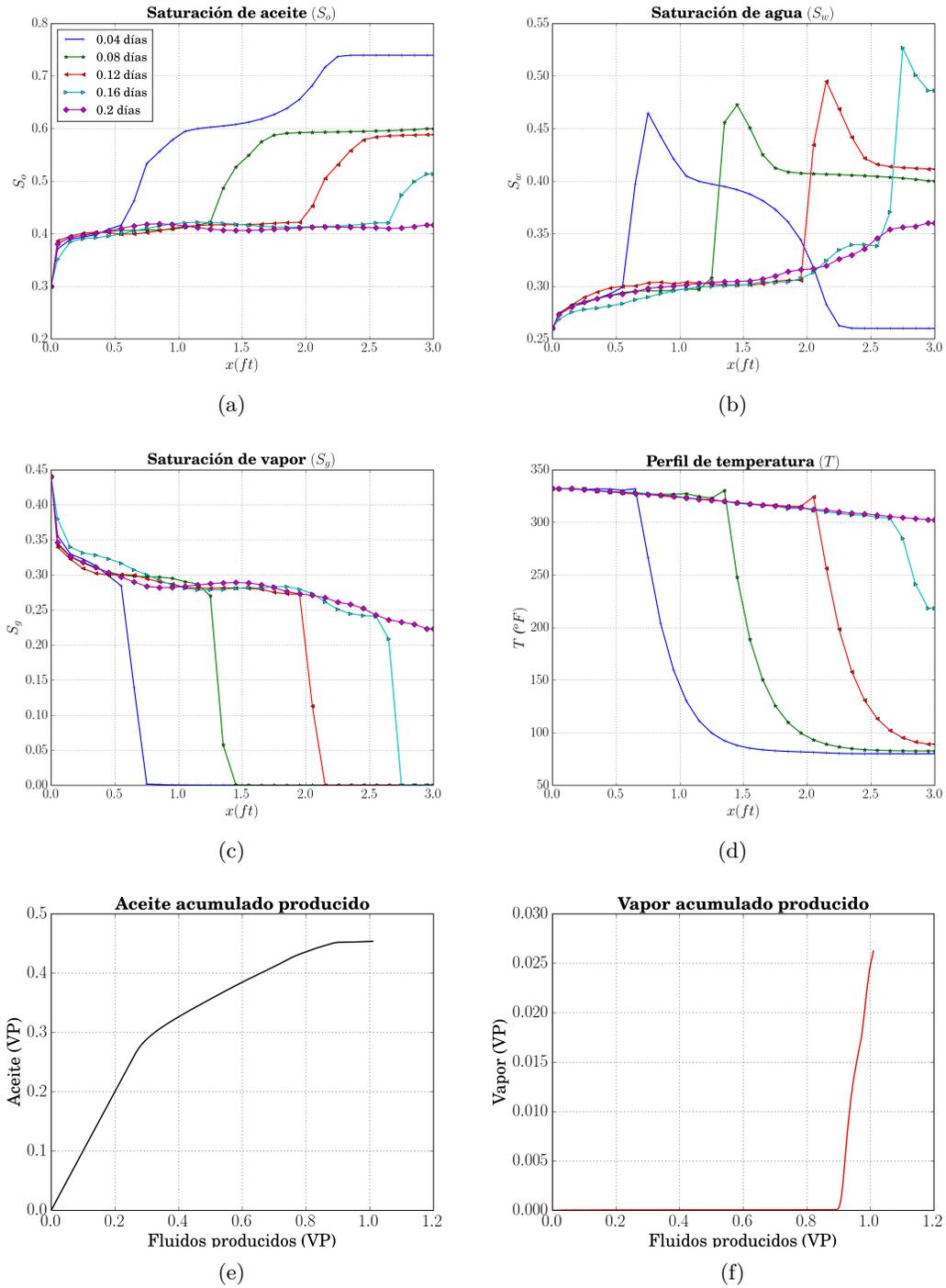


Figura 6.9: Perfiles obtenidos para la simulación de inyección de vapor en un núcleo utilizando 30 volúmenes de control.

6.3.2. Comparación del volumen de recuperación de aceite pesado mediante inyección de agua caliente versus inyección de vapor

Los métodos térmicos de EOR tienen como objetivo fundamental transferir energía a los fluidos del yacimiento para mejorar la movilidad de los mismos, principalmente del aceite y así producirlo con mayor facilidad. Teniendo esto en mente, en esta sección se compara la eficiencia de recuperación de aceite de dos métodos térmicos: inyección continua de agua caliente e inyección continua de vapor. Para llevar a cabo esta comparativa se seleccionó el problema simétrico de los 9 pozos, el cual describe el desplazamiento de aceite pesado por vapor y agua en un dominio isotrópico, en el modelo físico se coloca 1 pozo inyector y 3 pozos productores en las esquinas del dominio (ver figura 6.10). Este modo de colocar los pozos, emula la simetría como si se colocaran 1 pozo inyector en el centro de un dominio cuadrado y 4 pozos productores en las esquinas del dominio y 4 al centro de los bordes o fronteras del mismo. Las consideraciones para el modelo conceptual son:

1. El flujo de fluidos ocurre en un medio bidimensional.
2. El medio es isotrópico.
3. No se consideran los efectos de la presión capilar.
4. El efecto de la fuerza de gravedad es despreciable.
5. Existe una fuente (pozo inyector) y 3 sumideros (pozos productores).
6. Existe transferencia de masa entre la fase agua y la fase vapor.
7. No se consideran pérdidas de calor por las fronteras superior e inferior del yacimiento.

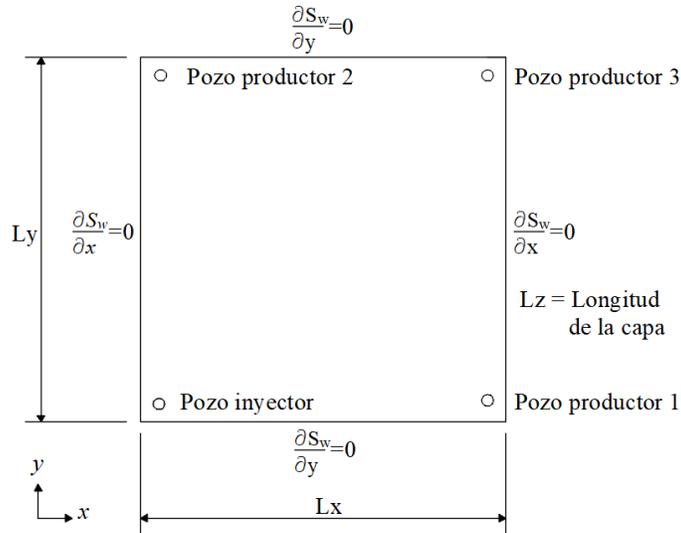


Figura 6.10: Modelo físico para el problema de los nueve pozos.

Los parámetros físicos para llevar a cabo la simulación se muestran en la tabla 6.3. Cabe señalar que casi todos parámetros físicos y las correlaciones de permeabilidad relativa, fueron

tomadas del problema 2A reportado en el cuarto proyecto comparativo de la SPE [36]. Las propiedades del agua y vapor son calculadas de acuerdo con el Apéndice B.

Tabla 6.3: Parámetros para simular el problema simétrico de los 9 pozos.

Propiedad	Valor
Permeabilidad ($k_{11} = k_{22}$)	2,000 (mD)
Porosidad (ϕ)	0.3
Tamaño del dominio ($L_x \times L_y \times L_z$)	100x100x80 (ft)
Conductividad térmica de la roca más el fluido (k_T)	24.0 (BTU/(ft day °F))
Conductividad térmica de sobrecarga ($k_{T,Ob}$)	24.0 (BTU/(ft day °F))
Densidad de la roca por su calor específico ($\rho_R C_R$)	35.0 (BTU/(ft ³ °F))
Densidad estándar del aceite ($\rho_{o,std}$)	60.68 lbm/ft ³
Temperatura inicial (T^0)	100.0 (°F)
Presión inicial (p_o^0)	80.0 (psi)
Temperatura de inyección de vapor (T^{inj})	400.0 (°F)
Temperatura de inyección de agua (T^{inj})	250.0 (°F)
Gasto de inyección equivalente de agua	25.0 (STB)
Presión de producción en el fondo del pozo (p_{wb}^{out})	17.0 (psi)
Saturación inicial de agua (S_w)	0.45
Saturación inicial de vapor (S_g)	0.0
Radio de los pozos	0.3 (ft)
Calidad de vapor	1.0

(Datos adaptados de [36].)

Los parámetros numéricos para llevar a cabo la simulación de inyección de vapor se seleccionaron como sigue: incrementos de 1×10^{-7} para calcular las derivadas con respecto a las variables primarias S_w y S_g , incrementos de 1×10^{-5} para calcular las derivadas con respecto a la presión p_o , ambos parámetros para calcular los bloques del Jacobiano correspondiente al balance de masa. El criterio de paro para salir del primer ciclo de NR se seleccionó a: $|\delta S_g| < 1 \times 10^{-3}$. Para resolver el balance de energía, se seleccionó un incremento de 1×10^{-5} para calcular las derivadas con respecto a la temperatura, y un criterio de paro de $|\delta T| < 0.1$ para salir del segundo ciclo de NR. Por otra parte, los parámetros numéricos para simular la inyección de agua caliente solamente difieren en el criterio para salir del primer ciclo de NR correspondiente al balance de masa, en el cual se considera la saturación de la fase agua S_w para calcular el criterio de salida, es decir, $|\delta S_w| < 1 \times 10^{-4}$. Se seleccionó una tamaño de malla de 10x10 y un tiempo total de 500 días de simulación para ejecutar ambos ejemplos.

Para llevar a cabo esta comparativa primero se muestran los perfiles correspondientes a las variables primarias al final de la simulación (500 días), esto para analizar el barrido del aceite cualitativamente en ambos casos. Posteriormente, se muestran las curvas de producción de aceite con las cuales se calcula una diferencia relativa del aceite producido, con el objetivo

de conocer qué método es el más eficiente.

La figura 6.11, muestra los perfiles de saturación, presión y temperatura obtenidos en la simulación de inyección de agua caliente para el tiempo de 500 días. En la figura 6.11 a), se presenta el perfil de saturación de agua S_w , en éste se nota que el agua ya esta siendo producida en los tres pozos de producción. Sin embargo, el perfil de saturación de agua indica que aunque el agua ya arribó a todos los pozos, el barrido del aceite no ha sido eficiente puesto que existe una gran cantidad de aceite aún en el medio (véase la figura 6.11 b). Este resultado se debe a la enorme diferencia de movilidades entre las fases. En la figura 6.11 c), se observa que la presión es mayor donde se encuentra el pozo inyector (esquina inferior izquierda), y es menor donde se ubican los pozos productores. Además, en la figura 6.11 d), se observa que la transferencia de energía entre los fluidos es lenta puesto que la temperatura sólo se ha elevado en aproximadamente un tercio del dominio, aun cuando la saturación de agua ya se ha incrementado en la mayoría del mismo.

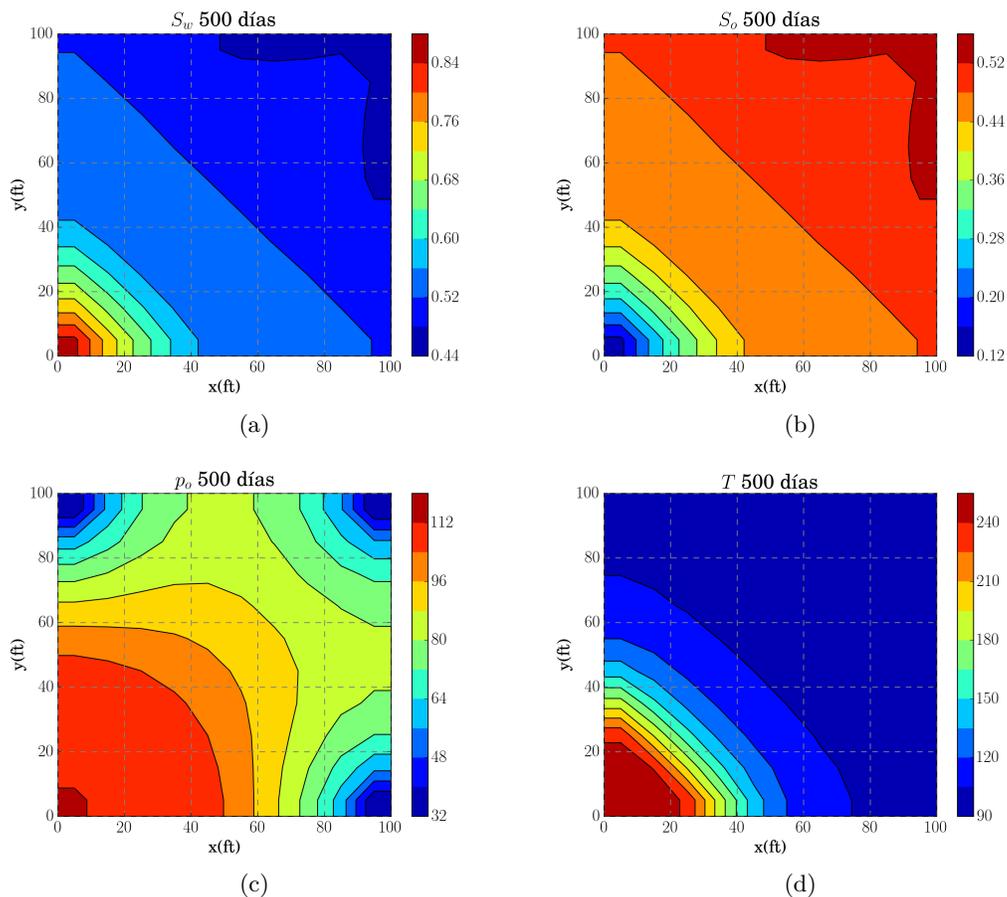


Figura 6.11: Perfiles obtenidos en la simulación de inyección de agua caliente para un tiempo de 500 días.

Por otra parte, la figura 6.12, muestra los perfiles de saturación y temperatura obtenidos en la simulación de inyección de vapor para el tiempo de 500 días. En la figura 6.12 a), se presenta el perfil de saturación de agua S_w , en éste se aprecia que la fase agua es móvil en casi todo el dominio y existe un concentración de esta misma fase que forma un pistón, el cual está delimitado diagonalmente desde los pozos productores 1 y 2. El perfil de saturación de agua indica que aunque esta fase ha aumentado en todo el dominio, la mayoría del barrido del aceite es hecho por la fase vapor, lo que se ve reflejado en las figuras 6.12 b) y 6.12 c); las cuales muestran los perfiles de la fase aceite y la fase vapor, respectivamente. Si se comparan los perfiles de la fase aceite obtenidos en la inyección de agua caliente (6.11 b)) y por inyección de vapor (6.12 b)), es claro que la fase aceite tiene una disminución más notable por el barrido de inyección de vapor. Esto se debe a que el transporte de energía del vapor es mucho mayor comparado con el del agua líquida, también la temperatura de inyección del vapor es más alta comparada con la temperatura en la inyección de agua, lo que provoca una disminución de la viscosidad del aceite, mejorando bastante la movilidad del mismo.

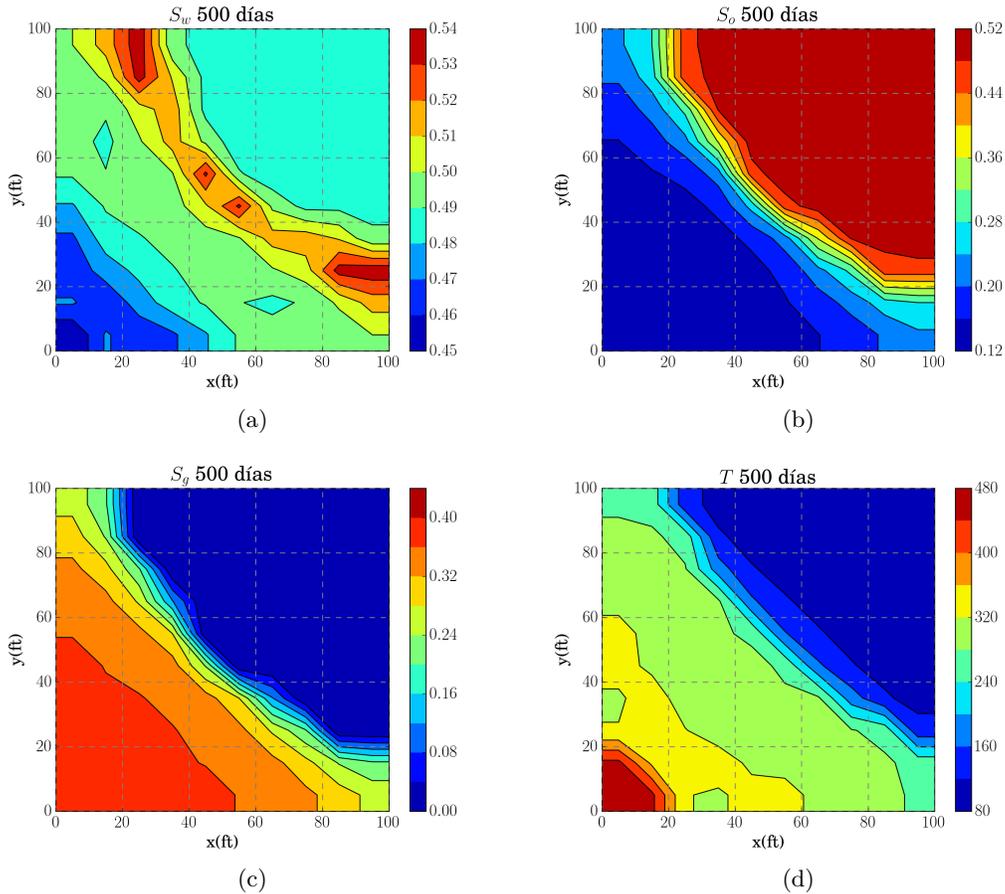


Figura 6.12: Perfiles obtenidos en la simulación de inyección de vapor para un tiempo de 500 días.

La transferencia de energía entre los fluidos en la inyección de vapor es más rápida, puesto que la temperatura se ha elevado en más de la mitad del dominio hasta los $280F^{\circ}$ aproximadamente (véase la figura 6.12 d)), mientras que para el caso anterior se alcanza una temperatura de $250F^{\circ}$ solamente en la vecindad del pozo inyector.

Para realizar una comparación cuantitativa, en cuanto a producción de aceite se refiere, se muestran las curvas de aceite producido en la figura 6.13, correspondiente a los pozos más cercano (1) y más lejano (3) para el caso de inyección de agua caliente. Cabe señalar que el pozo (2) se ha omitido por su simetría con el pozo (1). Observando la figura 6.13 a), se nota que la producción de aceite en el pozo (1) rebasó los 1,000STB de aceite producido, el valor calculado por el código numérico es de 1,087.3STB, mientras que para el pozo (3) se llega hasta los 1,013.6STB. La diferencia entre estos resultados se debe a la distancia del pozo inyector hacia los pozos productores.

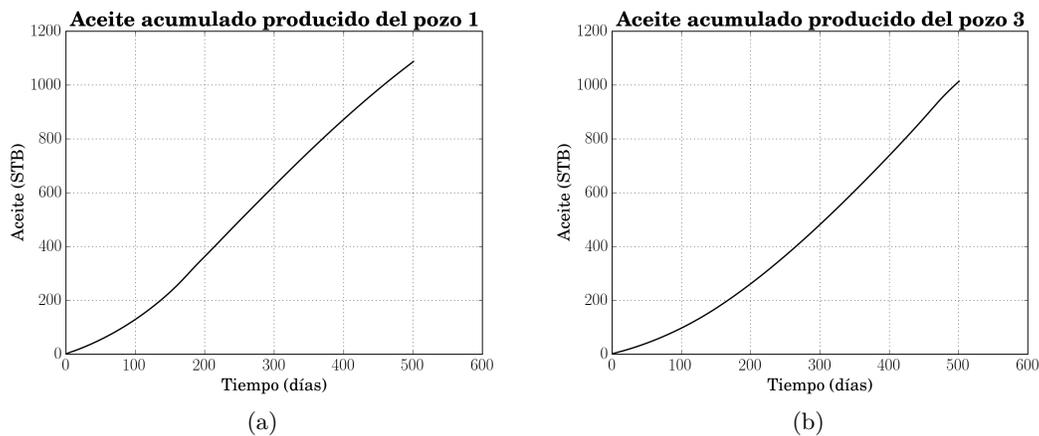


Figura 6.13: Recuperación de aceite mediante inyección de agua caliente en los pozos más cercano (1) y más lejano (3).

Los resultados de aceite producido obtenidos para el proceso de inyección de vapor se muestran en la figura 6.14. En la figura 6.14 a), se muestra el aceite producido correspondiente al pozo más cercano, en ésta se nota que el aceite producido supera los 3,500STB. Si se compararan los valores calculados por los códigos numéricos de inyección de agua y vapor para el pozo (1), es decir 1,087.3 y 3,727.8, se tiene una diferencia de producción para el pozo (1) de 2,640.5STB si se utiliza el método de inyección de vapor. La producción obtenida para el pozo (3) mediante inyección de vapor es de 1,781.5STB (véase la figura 6.14 b)), para este pozo la diferencia de producción es de 767.9STB más que en el proceso de inyección de agua caliente. Sumando la producción de los 3 pozos en ambos métodos de EOR, se obtiene una producción de aceite total de 3,188.2STB si se utiliza inyección de agua caliente, mientras que se obtienen 9,237.1STB de aceite producido si se utiliza el método de inyección de vapor. Considerando los valores de producción total de aceite obtenidos, se tiene que la producción de aceite obtenida mediante inyección de vapor es 2.89 veces mayor que si se utiliza la inyección

de agua para desplazar el aceite pesado.

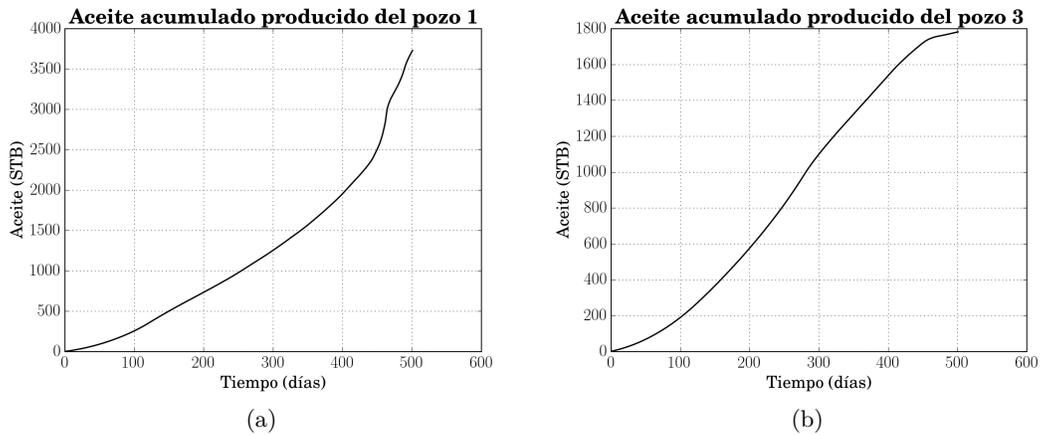


Figura 6.14: Recuperación de aceite mediante la inyección de vapor en los pozos más cercano (1) y más lejano (3).

Por otra parte, la figura 6.15, muestra las curvas de flujo fraccional obtenidas para el proceso de inyección de agua caliente. En esta figura, se aprecia que para el pozo (1) el corte de agua se obtuvo aproximadamente a los 170 días de inyección y el agua producida va en aumento hasta llegar aproximadamente al 90 % de los dos fluidos producidos, mientras que para el pozo (3), el agua comienza a producirse a los 450 días aproximadamente y ésta aumenta hasta un 42 % de los fluidos producidos en un corto tiempo. Estos resultados indican que la producción de agua supera la producción de aceite en los 3 pozos, por lo tanto, no es viable continuar con la inyección de agua caliente.

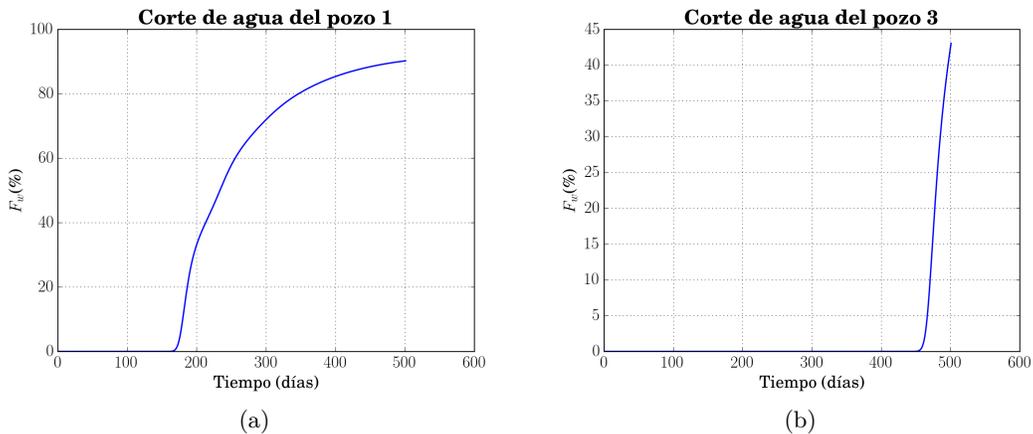


Figura 6.15: Flujo fraccional obtenido mediante inyección de agua caliente en los pozos más cercano (1) y más lejano (3).

La figura 6.16, muestra las curvas de flujo fraccional obtenidas para el proceso de inyección de vapor. La diferencia con respecto al proceso de inyección de agua caliente estriba en que el aceite es barrido por un pistón de agua seguido por un pistón de vapor. Por lo tanto existen 2 curvas de flujo fraccional, una calculada con las fases agua-aceite y otra calculada con las fases vapor-aceite. En las figuras 6.16 a) y c), las cuales corresponden al pozo (1), se nota que el agua comienza a producirse a los 120 días aproximadamente, y ésta decae casi a un valor de cero a los 460 días produciéndose vapor en su lugar. Para el pozo (3), el corte de agua ocurre a los 275 días aproximadamente (véase la figura 6.16 b)), sin embargo no se presenta producción de vapor en este pozo ya que el vapor aún no ha llegado hasta la ubicación del mismo.

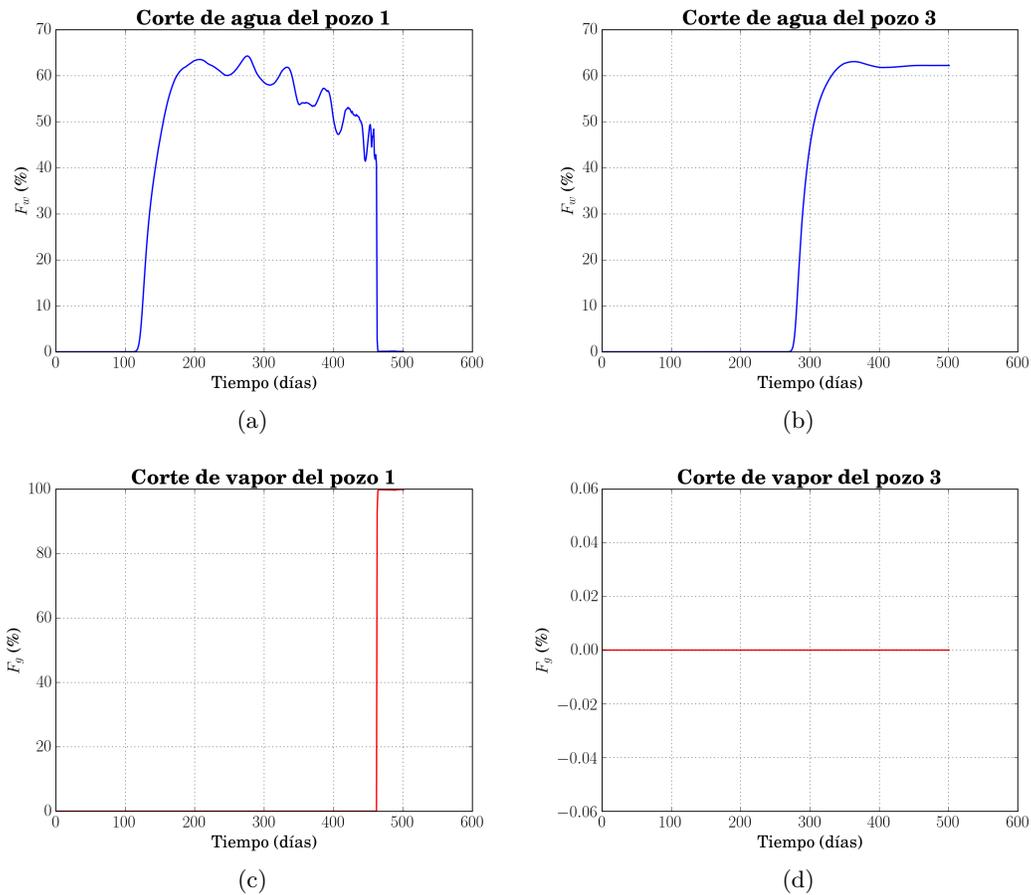


Figura 6.16: Recuperación de aceite y corte de vapor para los pozos más cercano (1) y más lejano (3), en la inyección de vapor.

Cabe aclarar que los resultados obtenidos en esta comparación sólo son aplicables al caso particular aquí presentando, por lo tanto, no deben ser extrapolados a otros casos.

6.4. Modelo numérico en dominios no ortogonales

Los aspectos matemáticos relacionados a la transformación de las ecuaciones de balance al plano computacional para simular numéricamente el proceso de inyección de vapor se trataron en el Capítulo 4 de este trabajo de tesis.

Por otra parte, el procedimiento para la discretización de las ecuaciones de balance transformadas se muestran en el apéndice A. Además, el algoritmo numérico general y el procedimiento para la construcción de los Jacobianos es idéntico al presentado en secciones previas de este capítulo, por lo que la implementación del modelo numérico se lleva a cabo de la misma manera que para el modelo en coordenadas cartesianas.

6.4.1. Resultados de la simulación numérica en dominios no ortogonales

Verificación del código numérico

Para verificar la implementación del modelo numérico se llevó a cabo la simulación del problema bidimensional simétrico de los 9 pozos, considerando los mismos parámetros físicos y numéricos dados en la sección anterior de este capítulo. Los resultados de esta simulación se muestran en la figura 6.17, la cual contiene las curvas de aceite producido.

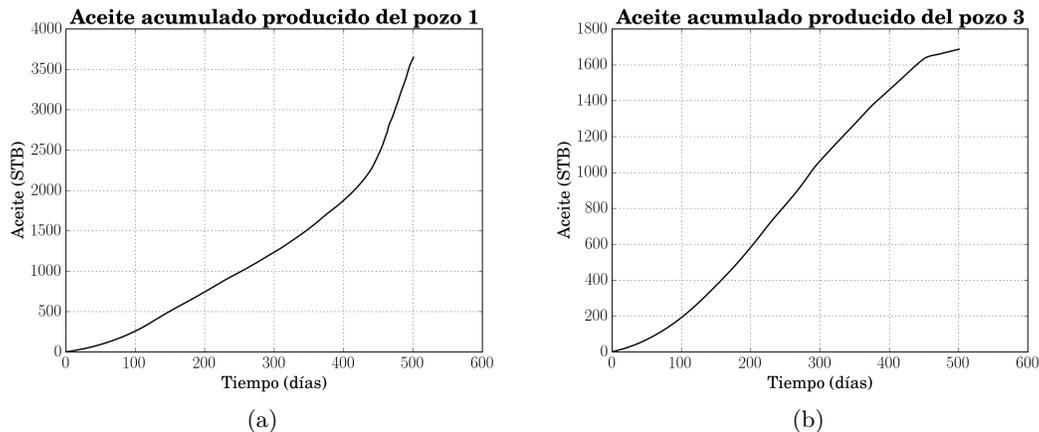


Figura 6.17: Curvas de aceite producido para los pozos más cercano (1) y más lejano (3).

Como se puede apreciar en la figura 6.17, las curvas de aceite producido presentan casi el mismo comportamiento cualitativo que para el modelo cartesiano. Sin embargo, para este caso en particular existen diferencias perceptibles en los valores de aceite producido con respecto a los resultados obtenidos utilizando el modelo cartesiano. Para el pozo más cercano se obtuvo una producción de 3,647.3STB, lo que significa una diferencia del 2.16 %, puesto que se obtuvo una producción de 3,727.8STB utilizando el código en coordenadas cartesianas. También, existe una diferencia para el pozo más lejano de 5.33 %, ya que se obtuvo una producción de 1,686.5STB, mientras que para el modelo Cartesiano se obtuvieron 1,781.5STB.

El aceite total producido tomando en cuenta los 3 pozos es de 8,981.1STB, este valor tiene una diferencia de 2.77 % comparándolo contra el resultado obtenido con el modelo numérico Cartesiano (9,237.1STB). Mediante los resultados aquí presentados se puede concluir que el modelo numérico para dominios no ortogonales está bien implementado y proporciona resultados congruentes con el comportamiento físico del problema de inyección de vapor, ya que proporciona una solución numérica cualitativa y cuantitativamente similar al modelo numérico Cartesiano para un problema ortogonal. Se optó por esta forma de comparación para el código numérico desarrollado, puesto que no hay referencias en la literatura que reporten un benchmark utilizando mallas estructuradas no ortogonales. Adicionalmente, existen diferencias reportadas de hasta el 25 % en las curvas de producción utilizando diferentes códigos numéricos para resolver el mismo problema, en el caso de la simulación del proceso de inyección de vapor (véase el cuarto proyecto comparativo de la SPE [36]).

Resultados para el dominio no ortogonal no. 1

Con el objetivo de conocer el comportamiento del código numérico desarrollado, se llevaron a cabo dos simulaciones en dominios no ortogonales considerando los mismos parámetros físicos que para la simulación cartesiana, es decir, se simula un problema con 1 pozo inyector y 3 pozos productores. Se debe señalar que ya no se tiene la simetría del problema original, puesto que la simulación numérica se lleva a cabo en dominios no ortogonales. Para el primer caso se seleccionó un dominio no ortogonal con bordes curvados, el cual se construyó sobre un cuadrado de 500x500ft. La figura 6.18, muestra los centros de los volúmenes de control para un tamaño de malla 20x20. La distribución de los pozos inyector y productores es como sigue: el pozo inyector está colocado en la esquina inferior izquierda del dominio y los pozos productores en las esquinas restantes. El tiempo de simulación total seleccionado es de 1,350 días y la frecuencia con la que se guardaron los resultados fue de 270 días.

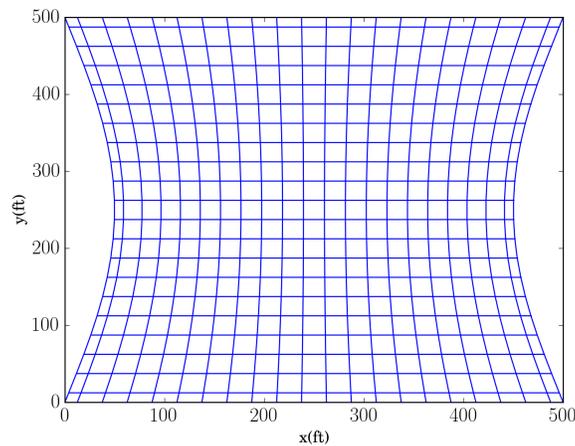


Figura 6.18: Centros de los volúmenes de control del dominio no ortogonal no. 1.

La figura 6.19, muestra los perfiles obtenidos para las variables primarias (p_o , S_w , S_g y T) después de 810 días de iniciada la inyección de vapor. En la figura 6.19 a), se distinguen dos concentraciones de agua, una en la vecindad del pozo inyector y otra por delante del frente de vapor. La concentración de agua en la vecindad del pozo inyector, se debe a que el agua condensada al inicio de la inyección no se ha evaporado por completo, aún cuando la temperatura en esa zona ha aumentado considerablemente. Observando la figura 6.19 b), se nota que el vapor ha avanzado hasta casi los 200ft del vértice inferior izquierdo del dominio, también que por delante de éste, viaja el frente de agua condensada. La figura 6.19 c), muestra el perfil correspondiente a la presión, en ésta se aprecia que la mayor presión se tiene en un radio aproximado de 180ft, tomando como referencia el punto de inyección y ésta disminuye con isólinas circulares en dirección a los pozos de producción, el cual es un comportamiento característico en este tipo de problemas. Por último, el perfil de temperatura se muestra en la figura 6.19 d), el cual corresponde con la propagación del vapor en el medio, puesto que éste transporta la mayor cantidad de energía.

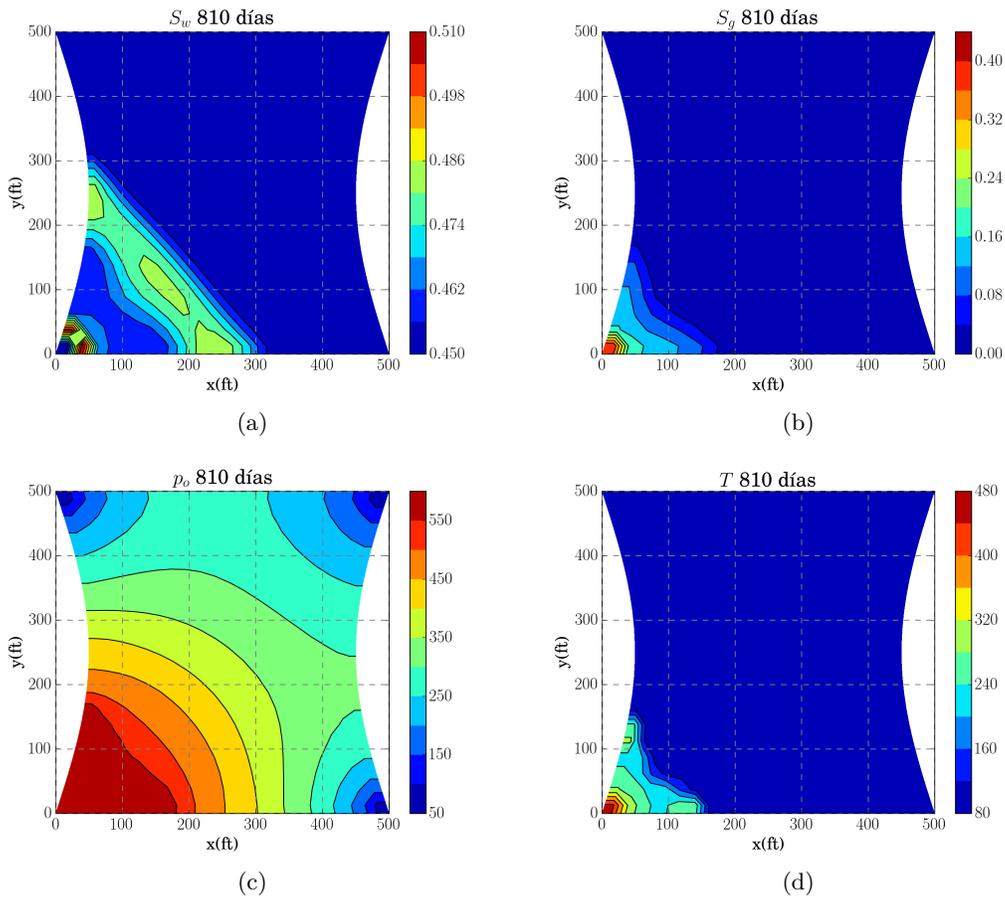


Figura 6.19: Perfiles obtenidos para las variables primarias en un tiempo de simulación de 810 días después de iniciar la inyección de vapor en el domino no ortogonal no. 1.

La figura 6.20, muestra los resultados de las variables primarias para un tiempo de simulación de 1,350 días. En la figura 6.20 a), se nota que aun se mantiene la concentración de agua en la vecindad del pozo inyector y que el frente de agua que está por delante del vapor ha avanzado hasta los pozos de producción más cercanos. Por otra parte, en la figura 6.20 b), se aprecia que el vapor se ha desplazado de manera asimétrica, pues éste ha avanzado poco más de los 300ft considerando la frontera inferior del dominio y hasta casi los 400ft, considerando como referencia la frontera curva donde se ubica el punto de inyección. Con respecto al perfil de presión mostrado en la figura 6.20 c), se observa que ésta ha incrementado en la mayoría del dominio, además que la isolínea más cercana al pozo inyector ha perdido su forma circular. El perfil de temperatura que se muestra en la figura 6.20 d), tiene correspondencia con la propagación del vapor en el medio, sin embargo se aprecia que hay una zona fría en la parte superior del pozo inyector. La isolínea no circular de presión y la zona fría pueden ser consecuencia de las oscilaciones numéricas presentes en el código, ya que éstas ocurren cuando se llega al corte de agua en la simulación.

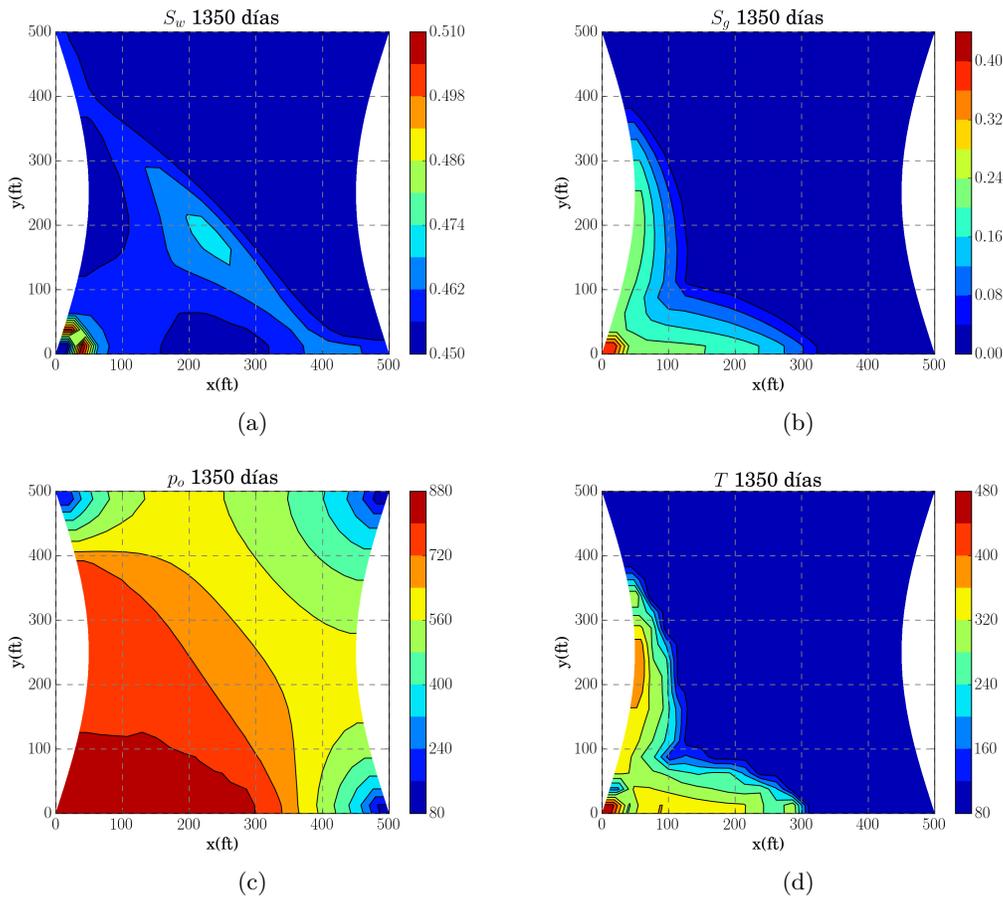


Figura 6.20: Perfiles obtenidos para las variables primarias en un tiempo de simulación de 1,350 días después de iniciar la inyección de vapor en el domino no ortogonal no. 1.

En la figura 6.21, se muestran las curvas de flujo fraccional F_w , con el objetivo de verificar que el agua ha comenzado a producirse en los pozos más cercanos. Fijando nuestra atención en la figura 6.21 a), la cual corresponde al pozo (1), se puede determinar que el corte de agua ocurrió a los 1,250 días aproximadamente y que el flujo fraccional decae al finalizar la simulación. Así mismo, observando la figura 6.21 b), se tiene que el corte de agua en el pozo (2) ocurre aproximadamente a los 1,300 días de simulación. La diferencia de tiempo en la que el agua comienza a producirse para los pozos (1) y (2) se debe naturalmente a la forma del dominio, puesto que la distancia entre los pozos inyector y productor es diferente para los dos pozos considerados.

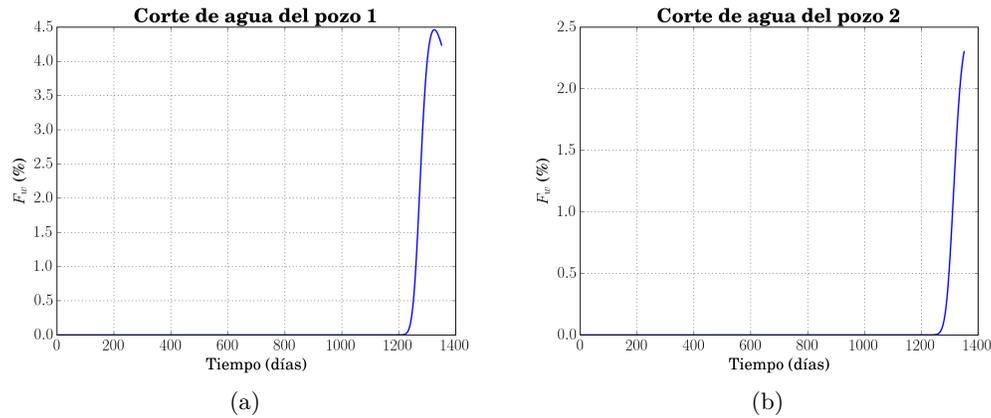


Figura 6.21: Curvas de flujo fraccional F_w para los pozos (1) y (2) en el domino no ortogonal no. 1.

Continuando con el análisis de los resultados, en la figura 6.22 se muestran las curvas correspondientes a la producción de aceite acumulado para los pozos más cercanos.

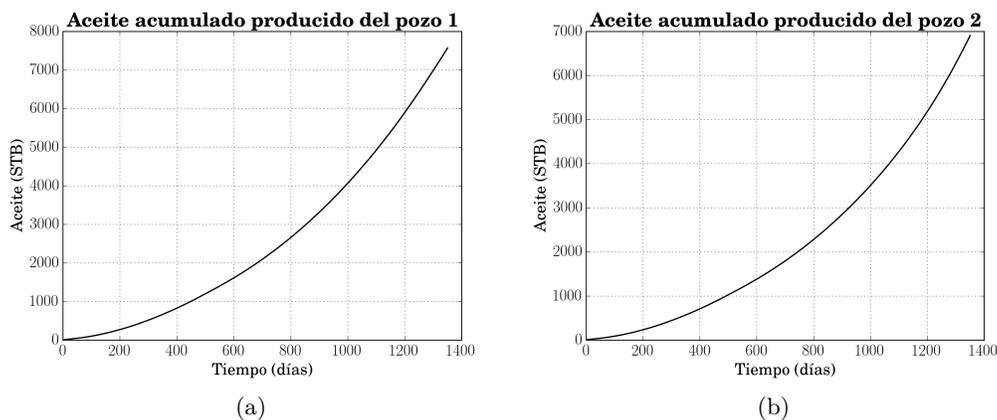


Figura 6.22: Curvas de aceite producido para los pozos (1) y (2) en el domino no ortogonal no. 1.

Observando la figura 6.22 a), se nota que se obtuvieron más de 7,500STB de aceite para el pozo (1), mientras que para el pozo (2) se obtuvieron poco menos de 7,000STB (véase la figura 6.22 b)). Los valores de aceite producido en STB calculados por el código numérico para los tres pozos de producción son: 7,567.9 para el pozo (1), 6,906.1 para el pozo (2) y 5,776.8 para el pozo (3); teniendo una producción total de 20,250.8 STB.

Resultados para el dominio no ortogonal no. 2

Para el segundo caso de aplicación se seleccionó un dominio no ortogonal con sus cuatro fronteras diferentes. La figura 6.23, muestra los centros de los volúmenes de control para un tamaño de malla 22x22. La distribución de los pozos inyector y productores se mantiene igual que en el problema anterior. El tiempo de simulación total seleccionado es de 600 días y la frecuencia con la que se guardaron los resultados fue de 100 días.

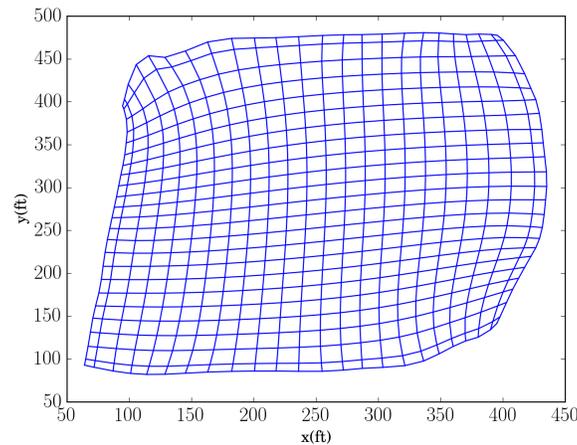


Figura 6.23: Centros de los volúmenes de control del dominio no ortogonal no. 2.

La figura 6.24, muestra los perfiles obtenidos de las variables primarias para un tiempo de simulación de 300 días. La figura 6.24 a), presenta el perfil de saturación de agua, en ésta se nota que el agua condensada se propaga desde la vecindad del pozo hacia adentro del dominio llegando hasta casi una cuarta parte del mismo. Si se observa la figura 6.24 b), la cual corresponde al perfil de saturación de vapor, se aprecia que éste sólo se ha incrementado en la vecindad del pozo inyector. Por otra parte, el perfil de la presión mostrado en la figura 6.24 c), indica que la mayor presión tiene un radio aproximado de 155ft, tomando como referencia el punto de inyección y ésta disminuye con isóneas circulares en dirección a los pozos de producción. El perfil de temperatura se muestra en la figura 6.24 d), el cual indica que sólo se han calentado los alrededores del pozo inyector.

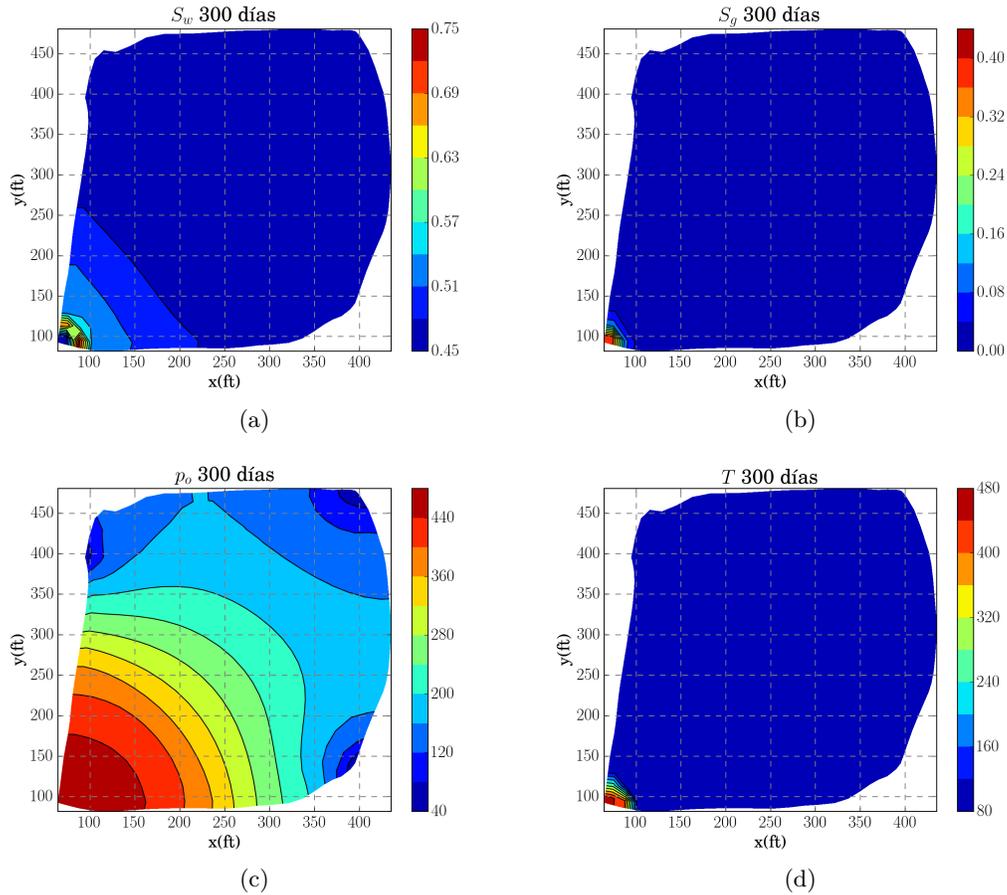


Figura 6.24: Perfiles obtenidos para las variables primarias en un tiempo de simulación de 300 días después de iniciar la inyección de vapor en el dominio no ortogonal no. 2.

Continuando con la descripción de los resultados, en la figura 6.25 se presentan los perfiles obtenidos para un tiempo de simulación de 600 días. En la figura 6.25 a), al igual que en el caso anterior, se distinguen dos concentraciones de agua, una en la vecindad del pozo inyector y otra por delante del frente de vapor, ésta última ya ha alcanzado la ubicación de los pozos de producción más cercanos. El agua condensada en la vecindad del pozo no se evapora completamente creando un banco de la misma. Por otro lado, observando la figura 6.25 b), se nota que el vapor ya se ha desplazado en el medio poroso hasta aproximadamente 300ft, considerando como referencia la ubicación del pozo inyector. En la figura 6.25 c), se presenta el perfil de presión, en ésta se aprecia que la presión ha aumentado considerablemente en más de la mitad del dominio, además que la zona de mayor presión aumentó respecto al tiempo de 300 días ya que ésta tiene un radio que se extiende desde el pozo inyector hasta aproximadamente 300ft en ambos ejes coordenados. En la figura 6.25 d), se observa que la temperatura ha aumentado en proporción en la que el vapor avanza en el medio.

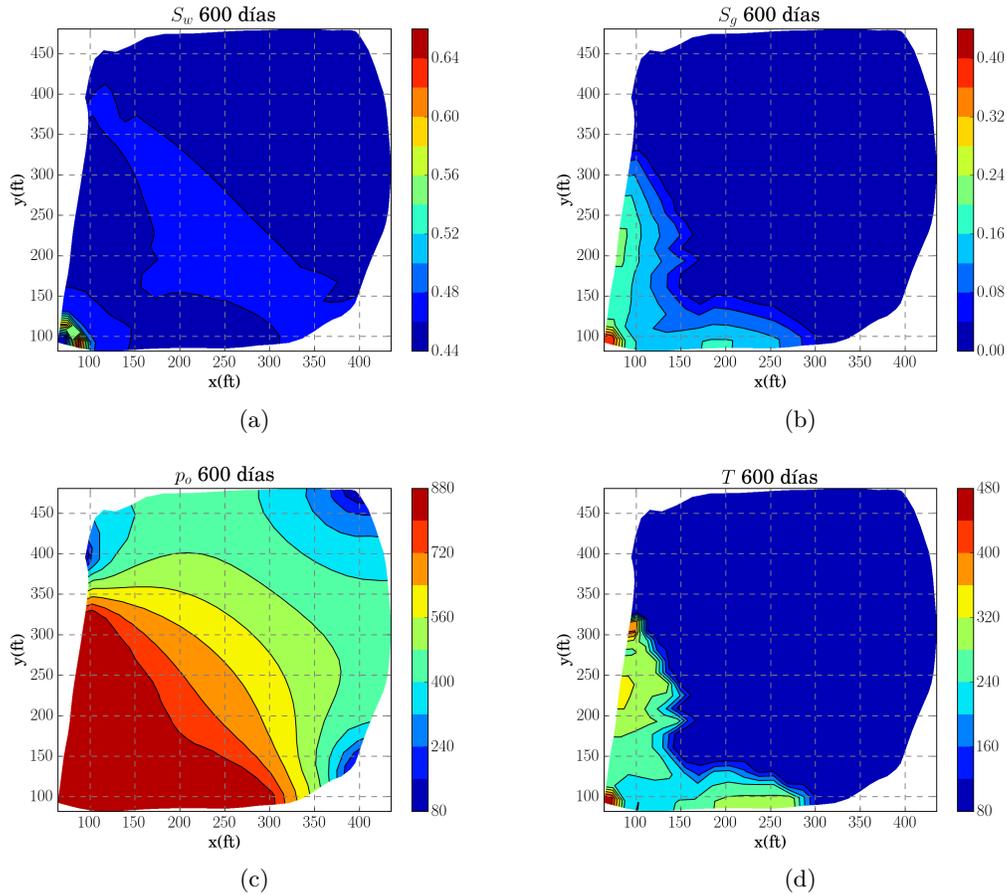


Figura 6.25: Perfiles obtenidos para las variables primarias en un tiempo de simulación de 600 días después de iniciar la inyección de vapor en el dominio no ortogonal no. 2.

En la figura 6.26, se muestran las curvas de flujo fraccional F_w . La figura 6.26 a), muestra la curva correspondiente al pozo (1), de ésta se puede determinar que el corte de agua ocurrió a los 520 días aproximadamente, además de que el flujo fraccional sube para luego decaer al finalizar la simulación. Observando la figura 6.26 b), se tiene que el corte de agua en el pozo (2) ocurre aproximadamente a los 430 días de simulación. La diferencia de tiempo en el corte de agua para los pozos (1) y (2), se debe a la forma del dominio y la distancia entre los pozos inyector y productor.

En la figura 6.27, se muestran las curvas correspondientes a la producción de aceite acumulado para los pozos más cercanos. La figura 6.27 a), muestra que se obtuvieron casi 5,000STB de aceite para el pozo (1), mientras que para el pozo (2) se obtuvieron aproximadamente de 5,200STB (véase la figura 6.27 b)). Los valores de aceite producido en STB calculados por el código para los tres pozos de producción son: 4,863.6 para el pozo (1), 5,311.4 para el pozo (2) y 2,802.3 para el pozo (3); esto da una producción total de 12,997.5STB.

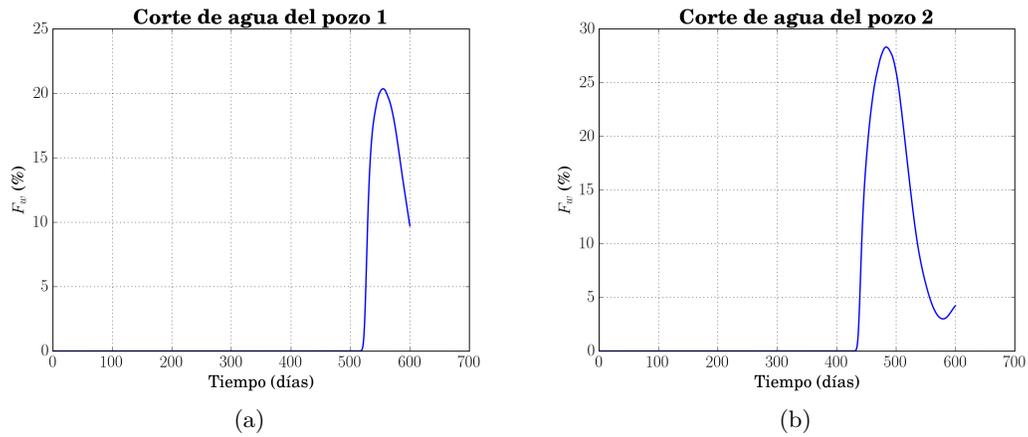


Figura 6.26: Curvas de flujo fraccional F_w para los pozos (1) y (2) en el domino no ortogonal no. 2.

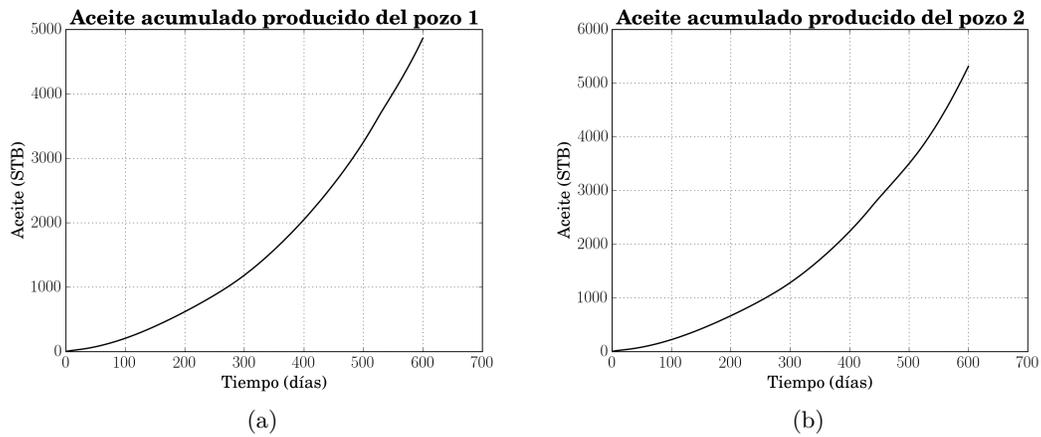


Figura 6.27: Curvas de aceite producido para los pozos (1) y (2) en el domino no ortogonal no. 2.

Con los resultados presentados para los dominios no ortogonales, se puede deducir que el código numérico desarrollado simula correctamente el comportamiento físico correspondiente a la dinámica del flujo de fluidos para el proceso de inyección de vapor en medios porosos. Sin embargo, se debe realizar un análisis numérico a profundidad con la finalidad de conocer los alcances de aplicación del código desarrollado, ya que en la literatura se han reportado casos de inestabilidad al corte de fluidos inyectados, cambio de resultados por orientación de malla para el mismo problema y resultados físicos incongruentes, por ejemplo, corte de vapor antes del corte de agua (véase referencias [6], [7] y [36]).

6.5. Experimentos numéricos de desempeño

Con objeto de analizar el desempeño de los códigos numéricos bidimensionales en coordenadas cartesianas realizados tanto para la CPU como para la GPU, se seleccionó el problema simétrico de los 9 pozos con un tiempo de simulación total de 5 días y paso de tiempo de 0.1 días. Los resultados presentados en este apartado se obtuvieron mediante la ejecución de ambos códigos numéricos en un nodo del Cluster del Instituto de Geofísica de la UNAM llamado *Olintlali*, del cual se utilizó un procesador Intel(R) Xeon(R) CPU 5650 2.67 GHz y una tarjeta GPU NVIDIA Tesla K40M(R) con 2880 cuda-cores y 12 Gigabytes de memoria.

La tabla 6.4 muestra el tiempo de cómputo obtenido en el cálculo del Jacobiano para el balance de masa, el cual incrementa considerablemente cuando el tamaño de la malla es más grande. Por ejemplo, se consumen 18.626s cuando el tamaño de malla es 1,000x1,000 en la CPU, mientras que utilizando la GPU sólo se gastan 0.031187s, eso significa 597.2x de aceleración. Este resultado muestra un considerable ahorro de tiempo tomando en cuenta que este procedimiento tiene que ser repetido cada iteración de Newton-Raphson.

Tabla 6.4: Tiempo promedio obtenido para el cálculo del Jacobiano del balance de masa.

Tamaño de malla	CPU C++ (s)	GPU CUDA (s)	Aceleración x
20x20	0.00830	0.00169	4.91x
40x40	0.02720	0.00176	15.45x
80x80	0.11298	0.00320	35.30x
180x180	0.59636	0.0152	39.23x
360x360	2.4503	0.02963	85.74x
500x500	4.5824	0.030281	151.32x
1,000x1,000	18.626	0.031187	597.23x

Si se comparan las aceleraciones obtenidas con el código numérico para flujo bifásico (véase la tabla 5.7) versus el código de inyección de vapor (tabla 6.4), las aceleraciones alcanzadas para el segundo caso son mejores. Esto se debe a que la tarjeta gráfica tesla K40 es superior en especificaciones de hardware, además de que ésta cuenta con la arquitectura *Kepler* la cual es por lo menos 3 veces más eficiente que la arquitectura *Fermi* de la tesla C2075, según especificaciones del fabricante. Como dato adicional, se debe tener en mente que la codificación de las funciones kernel para calcular el Jacobiano es similar en ambos casos.

Por otra parte, para resolver el sistema de ecuaciones lineales construido para el modelo de inyección de vapor se utilizó el método GMRES, tanto para la CPU como para la GPU. Los tiempos de cómputo se muestran en la tabla 6.5. Los resultados indican que la CPU es más rápida que la GPU cuando el sistema de ecuaciones lineales es pequeño, pero a partir de un tamaño de malla de 180x180 (97,200 incógnitas) la GPU es mucho más rápida, además de que la máxima aceleración es de 45.7x y se logra para un sistema de 3 000,000 de incógnitas.

Tabla 6.5: Tiempo de cómputo promedio obtenido para la solución del sistema de ecuaciones lineales mediante el método GMRES.

Tamaño de malla	Número de incógnitas	CPU C++ (s)	GPU CUDA (s)	Aceleración x
20x20	1,200	0.0036809	0.18021	0.0204x
40x40	48,00	0.039612	0.6765	0.0585x
80x80	19,200	0.36032	1.4049	0.2564x
180x180	97,200	6.9915	1.6055	4.35x
360x360	388,800	40.452	3.4091	11.86x
500x500	750,000	206.74	5.383	38.406x
1,000x1,000	3,000,000	803.32	17.592	45.7x

La figura 6.28, muestra el tiempo de cómputo real obtenido con la ejecución completa del código numérico tanto para la CPU como para la GPU. En la figura 6.28, se puede notar que la GPU es más lenta que la CPU cuando el problema se ejecuta con pocos nodos, mientras que la GPU es más rápida si el número de nodos aumenta. La aceleración total obtenida para el problema con un tamaño de malla de 1,000x1,000 es aproximadamente 32.6x, lo cual es un muy buen resultado. Cabe aclarar que la función para salvar datos se ejecutó 50 veces, es decir una vez cada paso de tiempo.

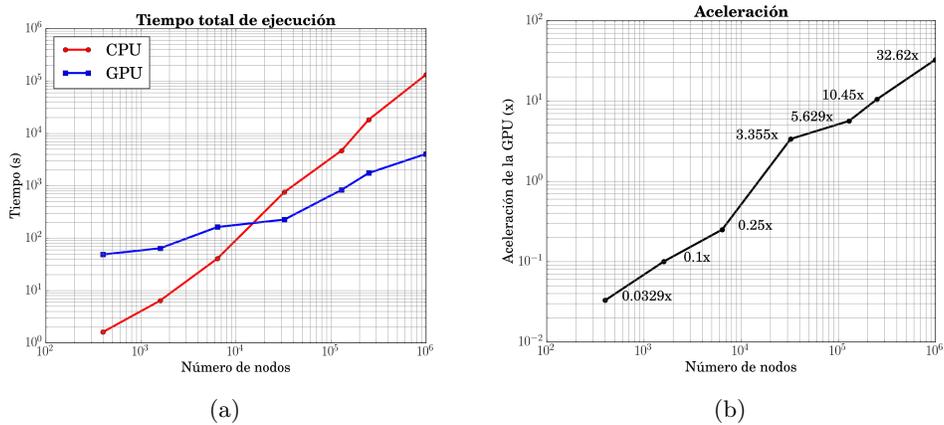


Figura 6.28: a) Tiempo de cómputo total de la CPU y la GPU, y b) aceleración lograda mediante el uso de la GPU.

Capítulo 7

Conclusiones

En este documento de investigación se presentan las metodologías numérico-computacionales para desarrollar los simuladores del flujo de fluidos bifásico y trifásico en medios porosos, tanto en códigos numéricos programados en serie utilizando la CPU como en paralelo utilizando la tecnología de las GPU's. Las implementaciones numéricas para la CPU y la GPU se desarrollaron mediante el uso de las bibliotecas de código abierto Eigen ++ y CUSP, respectivamente. Ambas implementaciones fueron validadas con problemas de referencia obteniendo resultados similares en comparación con otros autores. Con el objetivo de tener más claridad se separaron las conclusiones hechas para los modelos numérico-computacionales desarrollados.

Modelo numérico para el proceso isotérmico de inyección de agua

El modelo matemático y numérico utilizado para simular el flujo de fluidos bifásico isotérmico en medios porosos en un sistema de coordenadas cartesianas es ampliamente conocido en la literatura. Por lo que en el presente documento, enfocamos nuestros esfuerzos en la extensión de este modelo para que pudiese ser utilizado en dominios no ortogonales y en la paralelización del mismo utilizando la tecnología de las GPU's. Con relación a la aplicación del código numérico en dominios no ortogonales, éste se probó en 3 dominios diferentes con los que se pudo constatar que reproduce la física de fluidos involucrada de manera adecuada, además de mostrar una buena estabilidad numérica puesto que cuando ocurre el corte de agua el código no mostró oscilaciones para ninguno de los dominios de prueba. Sin embargo, debido a que los sistemas de ecuaciones lineales son más complejos para el modelo matemático transformado, ya que se tienen más diagonales en los bloques del Jacobiano y los términos de las derivadas cruzadas podrían mal condicionar los sistemas lineales resultantes, y por ende serían necesarios mejores solucionadores para resolverlos, se optó por implementar la estrategia de paralelización en el código numérico bidimensional Cartesiano. Esta estrategia de paralelización permite reducir el tiempo de cálculo de la matriz Jacobiana, que resulta del método de Newton-Raphson. Los resultados numéricos indican que la implementación en

la GPU es hasta 21 veces más rápida que su contraparte en la CPU para la malla más fina utilizada.

Por otra parte, la solución del sistema de ecuaciones lineales también se llevó a cabo en la GPU, obteniendo una aceleración de 2.2x para la malla más fina. En total, se obtuvo una aceleración de 3x teniendo en cuenta la construcción de la matriz Jacobiana, la solución del sistema de ecuaciones lineales y el intercambio de información entre la CPU y la GPU. Como era de esperar, esta aceleración se puede mejorar a medida que el número de incógnitas se incrementa, sin embargo, el número limitado de subprocesos y la memoria de la GPU utilizada para este problema fue un obstáculo para seguir adelante. Aunque las bibliotecas utilizadas para resolver los sistemas lineales están optimizadas, deben mejorarse con preconditionadores especiales para obtener mejores resultados en términos de tiempo de la CPU y la GPU para intentar igualar la velocidad de aceleración de 22x que se logró en nuestro mejor cálculo de Matriz Jacobiana. Por otro lado, la GPU utilizada para ejecutar este código numérico es relativamente obsoleta, de tal forma que estuvimos limitados por la cantidad de núcleos CUDA (448) y la memoria (6GB) del hardware; sin embargo, como se puede ver en los resultados, a medida que aumenta el tamaño del problema, la aceleración mejora, por lo tanto, si utilizamos, por ejemplo, una GPU Tesla K40m (2880 núcleos CUDA y 12 GB en memoria) se esperan mejores resultados. Finalmente, la estrategia de paralelización puede ser expandida para utilizar varias GPU junto con los métodos de descomposición de dominio; esto permitirá aumentar aún más el tamaño del problema (a varios millones de incógnitas) y, como consecuencia, se mejorará la velocidad. Por supuesto, esto requiere mejores solucionadores para los sistemas lineales, por ejemplo, métodos algebraicos *multigrid* o de multimalla.

Modelo numérico para el proceso térmico de inyección de vapor

El modelo matemático y numérico del flujo de fluidos trifásico térmico para simular la recuperación de aceite mediante inyección de vapor ha sido estudiado ampliamente y hasta el día de hoy sigue siendo objeto de estudio debido al crecimiento en la aplicación de los métodos de EOR a nivel internacional. En este trabajo de tesis se modificó con éxito el algoritmo propuesto por Shutler para proporcionar un mayor nivel de generalidad y aplicación. Las modificaciones hechas consisten en la utilización de una discretización completamente implícita en el tiempo para las ecuaciones de balance y el cambio de las variables primarias sustituyendo p_o , p_{cow} y p_{cog} del algoritmo original por p_o , S_w y S_g , esto para llevar a cabo el método de Newton-Rapshon. El código numérico desarrollado fue validado con los resultados experimentales para la inyección de vapor en un núcleo obtenidos por Willman *et al.* y al mismo tiempo comparado con los resultados de otros autores para este problema. Posteriormente, el código numérico se extendió para su aplicación en dominios bidimensionales donde se utilizaron los datos petrofísicos reportados por la SPE para simular el problema simétrico de los 9 pozos. Además, se construyó un código numérico para simular el proceso de inyección de agua caliente a partir de pequeños cambios hechos al código de inyección de vapor. Se lle-

vó a cabo una comparación donde se simularon la inyección de agua y la inyección de vapor para el problema antes mencionado, obteniendo como era de esperarse mejores resultados de producción de aceite en la simulación de inyección de vapor. También se amplió el modelo numérico-computacional para que pudiese ser aplicado en dominios no ortogonales, donde con base en los resultados obtenidos en los dominios no ortogonales de prueba, se concluye que el código numérico desarrollado proporciona resultados congruentes con la física del problema de inyección de vapor.

Debido a la complejidad inherente que se tiene en el modelo de inyección de vapor, se decidió paralelizar el código numérico bidimensional en coordenadas cartesianas utilizando la misma metodología que se utilizó para el modelo bifásico. En el cálculo del Jacobiano los resultados numéricos indican que la implementación en la GPU es hasta 597 veces más rápida que la CPU para la malla más fina utilizada. Así mismo, se obtuvo una aceleración de hasta 32.6 veces en el tiempo de ejecución total del código, por lo que se concluye que el uso de las GPU's es sin duda una opción viable para acelerar la simulación numérica del flujo de fluidos en yacimientos petroleros. También, se puede concluir que las especificaciones del hardware juegan un rol muy importante en la ejecución de un código numérico desarrollado para utilizar las GPU's, puesto que los resultados de aceleración obtenidos para el código de inyección de vapor son mucho mejores que los obtenidos para el código de inyección de agua.

Por otra parte, a diferencia del modelo bien conocido de flujo bifásico, para el código numérico de inyección de vapor desarrollado para dominios no ortogonales se debe realizar un análisis a profundidad sobre aspectos numéricos tales como: estabilidad del código al corte de fluidos de inyección, cálculo y direccionamiento del término condensación- evaporación Mg , relación de aspecto de malla y orden de cálculo para las derivadas del Jacobiano. Todo esto con la finalidad de tener un código mucho más robusto, puesto que en la literatura aún se reportan incluso para modelos cartesianos casos de inestabilidad numérica y resultados incongruentes con la física de fluidos en cuestión.

Por último, el contenido de este trabajo puede ser utilizado en el desarrollo de futuros simuladores multifásicos para su aplicación en dominios no ortogonales, puesto que se detalla la metodología de transformación, linealización y solución de las ecuaciones de balance de masa y energía. Adicionalmente, se presenta una metodología sencilla y efectiva para su paralelización utilizando las GPU's.

Apéndice A

Discretización de las ecuaciones transformadas

A.1. Inyección de agua

A.1.1. Discretización de la ecuación de saturación en el plano computacional

En esta sección se discretiza paso a paso la ecuación de saturación del modelo bidimensional en el plano transformado para simular el proceso de inyección de agua.

Reescribiendo la ecuación (5.61) en dos dimensiones y despreciando el efecto de la gravedad,

$$\frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) = \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} \right] + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_o}{\partial \xi} + A_{22} \frac{\partial p_o}{\partial \eta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} \right] + \frac{S}{J},$$

donde los tensores A_{ij} , y B_{ij} son:

$$\begin{aligned} A_{11} &= \frac{1}{J} (a_{11}^w \xi_x^2 + a_{22}^w \xi_y^2), & B_{11} &= \frac{1}{J} (b_{11}^w \xi_x^2 + b_{22}^w \xi_y^2), \\ A_{12} &= \frac{1}{J} (a_{11}^w \xi_x \eta_x + a_{22}^w \xi_y \eta_y), & B_{12} &= \frac{1}{J} (b_{11}^w \xi_x \eta_x + b_{22}^w \xi_y \eta_y), \\ A_{21} &= \frac{1}{J} (a_{11}^w \xi_x \eta_x + a_{22}^w \xi_y \eta_y), & B_{21} &= \frac{1}{J} (b_{11}^w \xi_x \eta_x + b_{22}^w \xi_y \eta_y), \\ A_{22} &= \frac{1}{J} (a_{11}^w \eta_x^2 + a_{22}^w \eta_y^2), & B_{22} &= \frac{1}{J} (b_{11}^w \eta_x^2 + b_{22}^w \eta_y^2), \end{aligned} \tag{A.1}$$

además los coeficientes a_{ii} y b_{ii} , están dados por:

$$\begin{aligned} a_{11}^w &= k_{11} \lambda_w, & b_{11}^w &= k_{11} \lambda_w \frac{dp_c}{dS_w}, \\ a_{22}^w &= k_{22} \lambda_w, & b_{22}^w &= k_{22} \lambda_w \frac{dp_c}{dS_w}. \end{aligned} \tag{A.2}$$

Por otra parte, el Jacobiano y las métricas, se definen de la siguiente manera:

$$J = (x_\xi y_\eta - x_\eta y_\xi), \quad (\text{A.3})$$

$$\xi_x = Jy_\eta, \quad (\text{A.4})$$

$$\xi_y = -Jx_\eta, \quad (\text{A.5})$$

$$\eta_x = -Jy_\xi, \quad (\text{A.6})$$

$$\eta_y = Jx_\xi. \quad (\text{A.7})$$

La figura A.1 muestra el volumen elemental en el plano transformado (computacional) donde se resolverán las ecuaciones gobernantes.

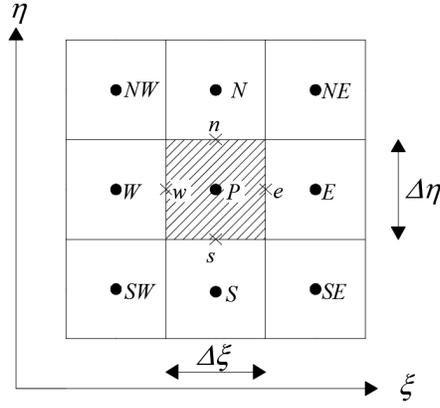


Figura A.1: Volumen de control elemental en el plano transformado. (Ilustración adaptada de [46].)

Integrando la ecuación (A.1) en el tiempo y sobre el volumen elemental de la figura A.1, se tiene:

$$\int_n^{n+1} \int_{dA} \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) d\xi d\eta d\tau = \int_n^{n+1} \int_{dA} \left\{ \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} \right] + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_o}{\partial \xi} + D_{22} \frac{\partial p_o}{\partial \eta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} \right] + \frac{S}{J} \right\} d\xi d\eta d\tau. \quad (\text{A.8})$$

Integrando el primer término de la ecuación (A.8), se obtiene:

$$\int_n^{n+1} \int_{\Delta A} \left(\frac{\partial}{\partial \tau} \frac{Q}{J} \right) d\xi d\eta d\tau = \left[\left(\frac{\phi S_w}{J} \right)^{n+1} - \left(\frac{\phi S_w}{J} \right)^n \right] \Delta\xi \Delta\eta. \quad (\text{A.9})$$

Para aproximar los términos restantes de la ecuación (A.8) se utiliza un esquema implícito en el tiempo y omitiendo el superíndice $n + 1$, se llega a:

$$\begin{aligned}
 & \int_n^{n+1} \int_{dA} \left\{ \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} \right] \right. \\
 & \quad \left. + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_o}{\partial \xi} + A_{22} \frac{\partial p_o}{\partial \eta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} \right] + \frac{S}{J} \right\} d\xi d\eta d\tau = \\
 & \quad \left\{ \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} \right]_e \Delta \eta - \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} \right]_w \Delta \eta - \right. \\
 & \quad \left[B_{11} \frac{\partial S_w}{\partial \xi} + B_{12} \frac{\partial S_w}{\partial \eta} \right]_e \Delta \eta + \left[B_{11} \frac{\partial S_w}{\partial \xi} + B_{12} \frac{\partial S_w}{\partial \eta} \right]_w \Delta \eta + \\
 & \quad \left[A_{21} \frac{\partial p_o}{\partial \xi} + A_{22} \frac{\partial p_o}{\partial \eta} \right]_n \Delta \xi - \left[A_{21} \frac{\partial p_o}{\partial \xi} + A_{22} \frac{\partial p_o}{\partial \eta} \right]_s \Delta \xi - \\
 & \quad \left. \left[B_{21} \frac{\partial S_w}{\partial \xi} + B_{22} \frac{\partial S_w}{\partial \eta} \right]_n \Delta \xi + \left[B_{21} \frac{\partial S_w}{\partial \xi} + B_{22} \frac{\partial S_w}{\partial \eta} \right]_s \Delta \xi \right\} \Delta \tau + \frac{S}{J} \Big|_P \Delta \xi \Delta \eta \Delta \tau.
 \end{aligned} \tag{A.10}$$

Para la discretización de las derivadas de la ecuación (A.10) se utilizan diferencias centradas, las cuales están dadas como sigue:

$$\begin{aligned}
 \frac{\partial p_o}{\partial \xi} \Big|_e &= \frac{p_{o,E} - p_{o,P}}{\Delta \xi}, \\
 \frac{\partial p_o}{\partial \xi} \Big|_w &= \frac{p_{o,P} - p_{o,W}}{\Delta \xi}, \\
 \frac{\partial p_o}{\partial \eta} \Big|_n &= \frac{p_{o,N} - p_{o,P}}{\Delta \eta}, \\
 \frac{\partial p_o}{\partial \eta} \Big|_s &= \frac{p_{o,P} - p_{o,S}}{\Delta \eta},
 \end{aligned} \tag{A.11}$$

y para las derivadas cruzadas [45]:

$$\begin{aligned}
 \frac{\partial p_o}{\partial \eta} \Big|_e &= \frac{(p_{o,N} + p_{o,NE}) - (p_{o,S} + p_{o,SE})}{4\Delta \eta}, \\
 \frac{\partial p_o}{\partial \eta} \Big|_w &= \frac{(p_{o,N} + p_{o,NW}) - (p_{o,S} + p_{o,SW})}{4\Delta \eta}, \\
 \frac{\partial p_o}{\partial \xi} \Big|_n &= \frac{(p_{o,E} + p_{o,NE}) - (p_{o,W} + p_{o,NW})}{4\Delta \xi}, \\
 \frac{\partial p_o}{\partial \xi} \Big|_s &= \frac{(p_{o,E} + p_{o,SE}) - (p_{o,W} + p_{o,SW})}{4\Delta \xi}.
 \end{aligned} \tag{A.12}$$

Para obtener las derivadas parciales de la saturación de agua (S_w) se sustituye S_w en lugar de p_o en las ecuaciones (A.11) y (A.12). Sustituyendo las ecuaciones (A.11) y (A.12) en la ecuación (A.10) se obtiene:

$$\begin{aligned}
 \int_n^{n+1} \int_{dA} \left\{ \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} \right] \right. \\
 \left. + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_o}{\partial \xi} + A_{22} \frac{\partial p_o}{\partial \eta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} \right] + \frac{S}{J} \right\} d\xi d\eta d\tau = \\
 \left\{ A_{11} \left(\frac{p_{o,E} - p_{o,P}}{\Delta \xi} \right) + A_{12} \left[\frac{(p_{o,N} + p_{o,NE}) - (p_{o,S} + p_{o,SE})}{4\Delta \eta} \right] \right\}_e \Delta \eta \Delta \tau - \\
 \left\{ A_{11} \left(\frac{p_{o,P} - p_{o,W}}{\Delta \xi} \right) + A_{12} \left[\frac{(p_{o,N} + p_{o,NW}) - (p_{o,S} + p_{o,SW})}{4\Delta \eta} \right] \right\}_w \Delta \eta \Delta \tau - \\
 \left\{ B_{11} \left(\frac{S_{w,E} - S_{w,P}}{\Delta \xi} \right) + B_{12} \left[\frac{(S_{w,N} + S_{w,NE}) - (S_{w,S} + S_{w,SE})}{4\Delta \eta} \right] \right\}_e \Delta \eta \Delta \tau + \\
 \left\{ B_{11} \left(\frac{S_{w,P} - S_{w,W}}{\Delta \xi} \right) + B_{12} \left[\frac{(S_{w,N} + S_{w,NW}) - (S_{w,S} + S_{w,SW})}{4\Delta \eta} \right] \right\}_w \Delta \eta \Delta \tau + \\
 \left\{ A_{22} \left(\frac{p_{o,N} - p_{o,P}}{\Delta \eta} \right) + A_{21} \left[\frac{(p_{o,E} + p_{o,NE}) - (p_{o,W} + p_{o,NW})}{4\Delta \xi} \right] \right\}_n \Delta \xi \Delta \tau - \\
 \left\{ A_{22} \left(\frac{p_{o,P} - p_{o,S}}{\Delta \eta} \right) + A_{21} \left[\frac{(p_{o,E} + p_{o,SE}) - (p_{o,W} + p_{o,SW})}{4\Delta \xi} \right] \right\}_s \Delta \xi \Delta \tau - \\
 \left\{ B_{22} \left(\frac{S_{w,N} - S_{w,P}}{\Delta \eta} \right) + B_{21} \left[\frac{(S_{w,E} + S_{w,NE}) - (S_{w,W} + S_{w,NW})}{4\Delta \xi} \right] \right\}_n \Delta \xi \Delta \tau + \\
 \left\{ B_{22} \left(\frac{S_{w,P} - S_{w,S}}{\Delta \eta} \right) + B_{21} \left[\frac{(S_{w,E} + S_{w,SE}) - (S_{w,W} + S_{w,SW})}{4\Delta \xi} \right] \right\}_s \Delta \xi \Delta \tau + \frac{S}{J} \Delta \eta \Delta \xi \Delta \tau \Big|_P.
 \end{aligned} \tag{A.13}$$

Rearreglando términos y considerando $\Delta \xi = \Delta \eta = 1$ se obtiene la discretización bidimensional en el plano computacional de la ecuación de saturación (S_w), escribiendo de forma compacta la expresión inmediatamente anterior, se llega a:

$$\begin{aligned}
 & [(\phi S_w - \phi S_w)^n]_P 1 / (J \Delta \tau) = \\
 & - L_P p_{o,P} + L_E p_{o,E} + L_W p_{o,W} + L_N p_{o,N} + L_S p_{o,S} + L_{NE} p_{o,NE} + L_{SE} p_{o,SE} + L_{NW} p_{o,NW} + L_{SW} p_{o,SW} \\
 & + M_P S_{w,P} - M_E S_{w,E} - M_W S_{w,W} - M_N S_{w,N} - M_S S_{w,S} - M_{NE} S_{w,NE} - M_{SE} S_{w,SE} - M_{NW} S_{w,NW} \\
 & - M_{SW} S_{w,SW} + \frac{S}{J} \Big|_P,
 \end{aligned} \tag{A.14}$$

donde los coeficientes L_{XY} y M_{XY} son:

$$\begin{aligned}
 L_P &= A_{11}|_e + A_{11}|_w + A_{22}|_n + A_{22}|_s, & M_P &= B_{11}|_e + B_{11}|_w + B_{22}|_n + B_{22}|_s, \\
 L_E &= A_{11}|_e + \frac{A_{21}}{4}|_n - \frac{A_{21}}{4}|_s, & M_E &= B_{11}|_e + \frac{B_{21}}{4}|_n - \frac{B_{21}}{4}|_s, \\
 L_W &= A_{11}|_w + \frac{A_{21}}{4}|_s - \frac{A_{21}}{4}|_n, & M_W &= B_{11}|_w + \frac{B_{21}}{4}|_s - \frac{B_{21}}{4}|_n, \\
 L_N &= A_{22}|_n + \frac{A_{12}}{4}|_e - \frac{A_{12}}{4}|_w, & M_N &= B_{22}|_n + \frac{B_{12}}{4}|_e - \frac{B_{12}}{4}|_w, \\
 L_S &= A_{22}|_s + \frac{A_{12}}{4}|_w - \frac{A_{12}}{4}|_e, & M_S &= B_{22}|_s + \frac{B_{12}}{4}|_w - \frac{B_{12}}{4}|_e, \\
 L_{NE} &= \frac{A_{12}}{4}|_e + \frac{A_{21}}{4}|_n, & M_{NE} &= \frac{B_{12}}{4}|_e + \frac{B_{21}}{4}|_n, \\
 L_{SE} &= -\frac{A_{12}}{4}|_e - \frac{A_{21}}{4}|_s, & M_{SE} &= -\frac{B_{12}}{4}|_e - \frac{B_{21}}{4}|_s, \\
 L_{NW} &= -\frac{A_{12}}{4}|_w - \frac{A_{21}}{4}|_n, & M_{NW} &= -\frac{B_{12}}{4}|_w - \frac{B_{21}}{4}|_n, \\
 L_{SW} &= \frac{A_{12}}{4}|_w + \frac{A_{21}}{4}|_s, & M_{SW} &= \frac{B_{12}}{4}|_w + \frac{B_{21}}{4}|_s.
 \end{aligned} \tag{A.15}$$

A.1.2. Discretización de la ecuación de presión en el plano computacional

Reescribiendo la ecuación de presión (5.71) en dos dimensiones y despreciando el efecto de la gravedad, se llega a:

$$\frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_o}{\partial \xi} + A_{12} \frac{\partial p_o}{\partial \eta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} \right] + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_o}{\partial \xi} + A_{22} \frac{\partial p_o}{\partial \eta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} \right] = -\frac{S}{J},$$

donde los coeficientes A_{ij} , y B_{ij} son:

$$\begin{aligned}
 A_{11} &= \frac{1}{J} (a_{11}\xi_x^2 + a_{22}\xi_y^2), & B_{11} &= \frac{1}{J} (b_{11}\xi_x^2 + b_{22}\xi_y^2), \\
 A_{12} &= \frac{1}{J} (a_{11}\xi_x\eta_x + a_{22}\xi_y\eta_y), & B_{12} &= \frac{1}{J} (b_{11}\xi_x\eta_x + b_{22}\xi_y\eta_y), \\
 A_{21} &= \frac{1}{J} (a_{11}\xi_x\eta_x + a_{22}\xi_y\eta_y), & B_{21} &= \frac{1}{J} (b_{11}\xi_x\eta_x + b_{22}\xi_y\eta_y), \\
 A_{22} &= \frac{1}{J} (a_{11}\eta_x^2 + a_{22}\eta_y^2), & B_{22} &= \frac{1}{J} (b_{11}\eta_x^2 + b_{22}\eta_y^2).
 \end{aligned} \tag{A.16}$$

aquí las métricas de la transformación fueron previamente definidas por las ecuaciones (A.3)-(A.7) y los coeficientes a_{ii} y b_{ii} , están dados en las siguientes ecuaciones:

$$\begin{aligned}
 a_{11} &= k_{11}\lambda, & b_{11} &= k_{11}\lambda_w \frac{dp_c}{dS_w}, \\
 a_{22} &= k_{22}\lambda, & b_{22} &= k_{22}\lambda_w \frac{dp_c}{dS_w}.
 \end{aligned} \tag{A.17}$$

Siguiendo el mismo procedimiento descrito para la ecuación de saturación, se llega a:

$$\begin{aligned}
 &-L_{PPo,P} + L_{EPo,E} + L_{WPo,W} + L_{NPo,N} + L_{SPo,S} + L_{NEPo,NE} + L_{SEPo,SE} + L_{NWPo,NW} + L_{SWPo,SW} \\
 &+ M_P S_{w,P} - M_E S_{w,E} - M_W S_{w,W} - M_N S_{w,N} - M_S S_{w,S} - M_{NE} S_{w,NE} - M_{SE} S_{w,SE} - M_{NW} S_{w,NW} \\
 &- M_{SW} S_{w,SW} = -\frac{S}{J} \Big|_P,
 \end{aligned} \tag{A.18}$$

donde los coeficientes L_{XY} y M_{XY} tienen la misma forma que aquellos dados por las ecuaciones (A.15).

Las ecuaciones (A.14) y (A.18) son no lineales y están acopladas, por lo que se utiliza el método de Newton-Rapshon para su linealización y solución.

A.2. Inyección de vapor

A.2.1. Discretización de la ecuación general de balance de masa en el plano computacional

En esta sección se discretiza la ecuación general del balance de masa del modelo bidimensional en el plano transformado para resolver numéricamente el problema de inyección de vapor.

Reescribiendo la ecuación (4.42) en dos dimensiones, se obtiene:

$$\frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) = \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_\alpha}{\partial \xi} + A_{12} \frac{\partial p_\alpha}{\partial \eta} - B_{11} \frac{\partial z}{\partial \xi} - B_{12} \frac{\partial z}{\partial \eta} \right] + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_\alpha}{\partial \xi} + A_{22} \frac{\partial p_\alpha}{\partial \eta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} \right] + \frac{S}{J}, \quad (\text{A.19})$$

donde los tensores A_{ij} , y B_{ij} , están dados por:

$$\begin{aligned} A_{11} &= \frac{1}{J} (a_{11}^\alpha \xi_x^2 + a_{22}^\alpha \xi_y^2), & B_{11} &= \frac{1}{J} (b_{11}^\alpha \xi_x^2 + b_{22}^\alpha \xi_y^2), \\ A_{12} &= \frac{1}{J} (a_{11}^\alpha \xi_x \eta_x + a_{22}^\alpha \xi_y \eta_y), & B_{12} &= \frac{1}{J} (b_{11}^\alpha \xi_x \eta_x + b_{22}^\alpha \xi_y \eta_y), \\ A_{21} &= \frac{1}{J} (a_{11}^\alpha \xi_x \eta_x + a_{22}^\alpha \xi_y \eta_y), & B_{21} &= \frac{1}{J} (b_{11}^\alpha \xi_x \eta_x + b_{22}^\alpha \xi_y \eta_y), \\ A_{22} &= \frac{1}{J} (a_{11}^\alpha \eta_x^2 + a_{22}^\alpha \eta_y^2), & B_{22} &= \frac{1}{J} (b_{11}^\alpha \eta_x^2 + b_{22}^\alpha \eta_y^2), \end{aligned} \quad (\text{A.20})$$

y los coeficientes a_{ii} y b_{ii} , se definen como sigue:

$$\begin{aligned} a_{11}^\alpha &= \rho_\alpha k_{11} \frac{k_{r\alpha}}{\mu_\alpha}, & b_{11}^\alpha &= \rho_\alpha k_{11} \frac{k_{r\alpha}}{\mu_\alpha} \rho_\alpha \varphi, \\ a_{22}^\alpha &= \rho_\alpha k_{22} \frac{k_{r\alpha}}{\mu_\alpha}, & b_{22}^\alpha &= \rho_\alpha k_{22} \frac{k_{r\alpha}}{\mu_\alpha} \rho_\alpha \varphi. \end{aligned} \quad (\text{A.21})$$

Integrando la ecuación (A.19) en el tiempo y sobre el volumen elemental de la figura A.1,

$$\begin{aligned} \int_n^{n+1} \int_{dA} \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) d\xi d\eta d\tau &= \int_n^{n+1} \int_{dA} \left\{ \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_\alpha}{\partial \xi} + A_{12} \frac{\partial p_\alpha}{\partial \eta} - B_{11} \frac{\partial z}{\partial \xi} - B_{12} \frac{\partial z}{\partial \eta} \right] \right. \\ &\quad \left. + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_\alpha}{\partial \xi} + A_{22} \frac{\partial p_\alpha}{\partial \eta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} \right] + \frac{S}{J} \right\} d\xi d\eta d\tau. \end{aligned} \quad (\text{A.22})$$

Integrando el primer término de la ecuación (A.22), se obtiene:

$$\int_n^{n+1} \int_{\Delta A} \left(\frac{\partial Q}{\partial \tau} \frac{Q}{J} \right) d\xi d\eta d\tau = \left[\left(\frac{\phi \rho_\alpha S_\alpha}{J} \right)^{n+1} - \left(\frac{\phi \rho_\alpha S_\alpha}{J} \right)^n \right] \Delta \xi \Delta \eta. \quad (\text{A.23})$$

Utilizando un esquema implícito en el tiempo, omitiendo el superíndice $n+1$ y evaluando con derivadas centradas las derivadas que aparecen dentro de la ecuación (A.22), se llega a:

$$\begin{aligned} \int_n^{n+1} \int_{\Delta A} \left\{ \frac{\partial}{\partial \xi} \left[A_{11} \frac{\partial p_\alpha}{\partial \xi} + A_{12} \frac{\partial p_\alpha}{\partial \eta} - B_{11} \frac{\partial S_w}{\partial \xi} - B_{12} \frac{\partial S_w}{\partial \eta} \right] \right. \\ \left. + \frac{\partial}{\partial \eta} \left[A_{21} \frac{\partial p_\alpha}{\partial \xi} + A_{22} \frac{\partial p_\alpha}{\partial \eta} - B_{21} \frac{\partial S_w}{\partial \xi} - B_{22} \frac{\partial S_w}{\partial \eta} \right] + \frac{S}{J} \right\} d\xi d\eta d\tau = \\ \left\{ A_{11} \left(\frac{p_{\alpha,E} - p_{\alpha,P}}{\Delta \xi} \right) + A_{12} \left[\frac{(p_{\alpha,N} + p_{\alpha,NE}) - (p_{\alpha,S} + p_{\alpha,SE})}{4\Delta \eta} \right] \right\}_e \Delta \eta \Delta \tau - \\ \left\{ A_{11} \left(\frac{p_{\alpha,P} - p_{\alpha,W}}{\Delta \xi} \right) + A_{12} \left[\frac{(p_{\alpha,N} + p_{\alpha,NW}) - (p_{\alpha,S} + p_{\alpha,SW})}{4\Delta \eta} \right] \right\}_w \Delta \eta \Delta \tau - \\ \left\{ B_{11} \left(\frac{z_E - z_P}{\Delta \xi} \right) + B_{12} \left[\frac{(z_N + z_{NE}) - (z_S + z_{SE})}{4\Delta \eta} \right] \right\}_e \Delta \eta \Delta \tau + \\ \left\{ B_{11} \left(\frac{z_P - z_W}{\Delta \xi} \right) + B_{12} \left[\frac{(z_N + z_{NW}) - (z_S + z_{SW})}{4\Delta \eta} \right] \right\}_w \Delta \eta \Delta \tau + \\ \left\{ A_{22} \left(\frac{p_{\alpha,N} - p_{\alpha,P}}{\Delta \eta} \right) + A_{21} \left[\frac{(p_{\alpha,E} + p_{\alpha,NE}) - (p_{\alpha,W} + p_{\alpha,NW})}{4\Delta \xi} \right] \right\}_n \Delta \xi \Delta \tau - \\ \left\{ A_{22} \left(\frac{p_{\alpha,P} - p_{\alpha,S}}{\Delta \eta} \right) + A_{21} \left[\frac{(p_{\alpha,E} + p_{\alpha,SE}) - (p_{\alpha,W} + p_{\alpha,SW})}{4\Delta \xi} \right] \right\}_s \Delta \xi \Delta \tau - \\ \left\{ B_{22} \left(\frac{z_N - z_P}{\Delta \eta} \right) + B_{21} \left[\frac{(z_E + z_{NE}) - (z_W + z_{NW})}{4\Delta \xi} \right] \right\}_n \Delta \xi \Delta \tau + \\ \left\{ B_{22} \left(\frac{z_P - z_S}{\Delta \eta} \right) + B_{21} \left[\frac{(z_E + z_{SE}) - (z_W + z_{SW})}{4\Delta \xi} \right] \right\}_s \Delta \xi \Delta \tau + \frac{S}{J} \Delta \eta \Delta \xi \Delta \tau \Big|_P. \end{aligned} \quad (\text{A.24})$$

Rearreglando términos y considerando $\Delta \xi = \Delta \eta = 1$ se obtiene la discretización bidimensional en el plano computacional de la ecuación de balance de masa de la fase α , escribiendo de forma compacta la ecuación (A.24), se obtiene:

$$\begin{aligned} & [(\phi \rho_\alpha S_\alpha - \phi \rho_\alpha S_\alpha)^n]_P 1/(J \Delta \tau) = \\ & - L_P p_{\alpha,P} + L_E p_{\alpha,E} + L_W p_{\alpha,W} + L_N p_{\alpha,N} + L_S p_{\alpha,S} + L_{NE} p_{\alpha,NE} + L_{SE} p_{\alpha,SE} + L_{NW} p_{\alpha,NW} + L_{SW} p_{\alpha,SW} \\ & + M_P z_P - M_E z_E - M_W z_W - M_N z_N - M_S z_S - M_{NE} z_{NE} - M_{SE} z_{SE} - M_{NW} z_{NW} \\ & - M_{SW} z_{SW} + \frac{S}{J} \Big|_P, \end{aligned} \quad (\text{A.25})$$

donde los coeficientes L_{XY} y M_{XY} son:

$$\begin{aligned}
 L_P &= A_{11}|_e + A_{11}|_w + A_{22}|_n + A_{22}|_s, & M_P &= B_{11}|_e + B_{11}|_w + B_{22}|_n + B_{22}|_s, \\
 L_E &= A_{11}|_e + \frac{A_{21}}{4}|_n - \frac{A_{21}}{4}|_s, & M_E &= B_{11}|_e + \frac{B_{21}}{4}|_n - \frac{B_{21}}{4}|_s, \\
 L_W &= A_{11}|_w + \frac{A_{21}}{4}|_s - \frac{A_{21}}{4}|_n, & M_W &= B_{11}|_w + \frac{B_{21}}{4}|_s - \frac{B_{21}}{4}|_n, \\
 L_N &= A_{22}|_n + \frac{A_{12}}{4}|_e - \frac{A_{12}}{4}|_w, & M_N &= B_{22}|_n + \frac{B_{12}}{4}|_e - \frac{B_{12}}{4}|_w, \\
 L_S &= A_{22}|_s + \frac{A_{12}}{4}|_w - \frac{A_{12}}{4}|_e, & M_S &= B_{22}|_s + \frac{B_{12}}{4}|_w - \frac{B_{12}}{4}|_e, \\
 L_{NE} &= \frac{A_{12}}{4}|_e + \frac{A_{21}}{4}|_n, & M_{NE} &= \frac{B_{12}}{4}|_e + \frac{B_{21}}{4}|_n, \\
 L_{SE} &= -\frac{A_{12}}{4}|_e - \frac{A_{21}}{4}|_s, & M_{SE} &= -\frac{B_{12}}{4}|_e - \frac{B_{21}}{4}|_s, \\
 L_{NW} &= -\frac{A_{12}}{4}|_w - \frac{A_{21}}{4}|_n, & M_{NW} &= -\frac{B_{12}}{4}|_w - \frac{B_{21}}{4}|_n, \\
 L_{SW} &= \frac{A_{12}}{4}|_w + \frac{A_{21}}{4}|_s, & M_{SW} &= \frac{B_{12}}{4}|_w + \frac{B_{21}}{4}|_s.
 \end{aligned} \tag{A.26}$$

A.2.2. Discretización de la ecuación de balance de energía en el plano computacional

La ecuación de balance de energía bidimensional en el plano transformado es:

$$\begin{aligned}
 \frac{\partial}{\partial \tau} \left(\frac{Q}{J} \right) &= \frac{\partial}{\partial \xi} \left[\sum_{\alpha} \left(A_{11} \frac{\partial p_{\alpha}}{\partial \xi} + A_{12} \frac{\partial p_{\alpha}}{\partial \eta} - B_{11} \frac{\partial z}{\partial \xi} - B_{12} \frac{\partial z}{\partial \eta} \right) + C_{11} \frac{\partial T}{\partial \xi} + C_{12} \frac{\partial T}{\partial \eta} \right] \\
 &+ \frac{\partial}{\partial \eta} \left[\sum_{\alpha} \left(A_{21} \frac{\partial p_{\alpha}}{\partial \xi} + A_{22} \frac{\partial p_{\alpha}}{\partial \eta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} \right) + C_{21} \frac{\partial T}{\partial \xi} + C_{22} \frac{\partial T}{\partial \eta} \right] + \frac{S}{J},
 \end{aligned} \tag{A.27}$$

donde los tensores A_{ij} , B_{ij} y C_{ij} , están dados por:

$$\begin{aligned}
 A_{11} &= \frac{1}{J} (a_{11}^{\alpha} \xi_x^2 + a_{22}^{\alpha} \xi_y^2), & B_{11} &= \frac{1}{J} (b_{11}^{\alpha} \xi_x^2 + b_{22}^{\alpha} \xi_y^2), & C_{11} &= \frac{1}{J} (c_{11} \xi_x^2 + c_{22} \xi_y^2), \\
 A_{12} &= \frac{1}{J} (a_{11}^{\alpha} \xi_x \eta_x + a_{22}^{\alpha} \xi_y \eta_y), & B_{12} &= \frac{1}{J} (b_{11}^{\alpha} \xi_x \eta_x + b_{22}^{\alpha} \xi_y \eta_y), & C_{12} &= \frac{1}{J} (c_{11} \xi_x \eta_x + c_{22} \xi_y \eta_y), \\
 A_{21} &= \frac{1}{J} (a_{11}^{\alpha} \xi_x \eta_x + a_{22}^{\alpha} \xi_y \eta_y), & B_{21} &= \frac{1}{J} (b_{11}^{\alpha} \xi_x \eta_x + b_{22}^{\alpha} \xi_y \eta_y), & C_{21} &= \frac{1}{J} (c_{11} \xi_x \eta_x + c_{22} \xi_y \eta_y), \\
 A_{22} &= \frac{1}{J} (a_{11}^{\alpha} \eta_x^2 + a_{22}^{\alpha} \eta_y^2), & B_{22} &= \frac{1}{J} (b_{11}^{\alpha} \eta_x^2 + b_{22}^{\alpha} \eta_y^2), & C_{22} &= \frac{1}{J} (c_{11} \eta_x^2 + c_{22} \eta_y^2),
 \end{aligned} \tag{A.28}$$

a su vez los coeficientes a_{ii}^{α} , b_{ii}^{α} y c_{ii} , están definidos por:

$$\begin{aligned}
 a_{11}^{\alpha} &= h_{\alpha} \rho_{\alpha} k_{11} \frac{k_{r\alpha}}{\mu_{\alpha}}, & b_{11}^{\alpha} &= h_{\alpha} \rho_{\alpha} k_{11} \frac{k_{r\alpha}}{\mu_{\alpha}} \rho_{\alpha} \wp, & c_{11} &= k_{T11}, \\
 a_{22}^{\alpha} &= h_{\alpha} \rho_{\alpha} k_{22} \frac{k_{r\alpha}}{\mu_{\alpha}}, & b_{22}^{\alpha} &= h_{\alpha} \rho_{\alpha} k_{22} \frac{k_{r\alpha}}{\mu_{\alpha}} \rho_{\alpha} \wp, & c_{22} &= k_{T22},
 \end{aligned} \tag{A.29}$$

donde el Jacobiano y las métricas son los mismos que en los casos anteriores.

Integrando la ecuación (A.27) en el tiempo utilizando un esquema completamente implícito y sobre el volumen elemental de la figura A.1, se llega a:

$$\begin{aligned}
 & \left\{ \left[\frac{Q}{J} \right]^{n+1} - \left[\frac{Q}{J} \right]^n \right\} \Delta\xi \Delta\eta = \\
 & \left[\sum_{\alpha} \left(A_{11} \frac{\partial p_{\alpha}}{\partial \xi} + A_{12} \frac{\partial p_{\alpha}}{\partial \eta} - B_{11} \frac{\partial z}{\partial \xi} - B_{12} \frac{\partial z}{\partial \eta} \right) + C_{11} \frac{\partial T}{\partial \xi} + C_{12} \frac{\partial T}{\partial \eta} \right]_e^{n+1} \Delta\eta \Delta\tau - \\
 & \left[\sum_{\alpha} \left(A_{11} \frac{\partial p_{\alpha}}{\partial \xi} + A_{12} \frac{\partial p_{\alpha}}{\partial \eta} - B_{11} \frac{\partial z}{\partial \xi} + B_{12} \frac{\partial z}{\partial \eta} \right) + C_{11} \frac{\partial T}{\partial \xi} + C_{12} \frac{\partial T}{\partial \eta} \right]_w^{n+1} \Delta\eta \Delta\tau + \\
 & \left[\sum_{\alpha} \left(A_{21} \frac{\partial p_{\alpha}}{\partial \xi} + A_{22} \frac{\partial p_{\alpha}}{\partial \eta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} \right) + C_{21} \frac{\partial T}{\partial \xi} + C_{22} \frac{\partial T}{\partial \eta} \right]_n^{n+1} \Delta\xi \Delta\tau - \\
 & \left[\sum_{\alpha} \left(A_{21} \frac{\partial p_{\alpha}}{\partial \xi} + A_{22} \frac{\partial p_{\alpha}}{\partial \eta} - B_{21} \frac{\partial z}{\partial \xi} - B_{22} \frac{\partial z}{\partial \eta} \right) + C_{21} \frac{\partial T}{\partial \xi} + C_{22} \frac{\partial T}{\partial \eta} \right]_s^{n+1} \Delta\xi \Delta\tau + \frac{S}{J} \Delta\xi \Delta\eta \Delta\tau. \quad (\text{A.30})
 \end{aligned}$$

Para la evaluación de las derivadas de la ecuación (A.30), se utilizan diferencias centradas como sigue:

$$\begin{aligned}
 & \left\{ \left[\frac{Q}{J} \right]^{n+1} - \left[\frac{Q}{J} \right]^n \right\} \Delta\xi \Delta\eta = \\
 & \left\{ \sum_{\alpha} \left[A_{11}|_e \left(\frac{p_{\alpha,E} - p_{\alpha,P}}{\Delta\xi} \right) + A_{12}|_e \left(\frac{(p_{\alpha,N} + p_{\alpha,NE}) - (p_{\alpha,S} + p_{\alpha,SE})}{4\Delta\eta} \right) \right] - \right. \\
 & \quad \sum_{\alpha} \left[B_{11}|_e \left(\frac{z_E - z_P}{\Delta\xi} \right) + B_{12}|_e \left(\frac{(z_N + z_{NE}) - (z_S + z_{SE})}{4\Delta\eta} \right) \right] + \\
 & \quad \left. C_{11}|_e \left(\frac{T_E - T_P}{\Delta\xi} \right) + C_{12}|_e \left[\frac{(T_N + T_{NE}) - (T_S + T_{SE})}{4\Delta\eta} \right] \right\}^{n+1} \Delta\eta \Delta\tau - \\
 & \left\{ \sum_{\alpha} \left[A_{11}|_w \left(\frac{p_{\alpha,P} - p_{\alpha,W}}{\Delta\xi} \right) + A_{12}|_w \left(\frac{(p_{\alpha,N} + p_{\alpha,NW}) - (p_{\alpha,S} + p_{\alpha,SW})}{4\Delta\eta} \right) \right] - \right. \\
 & \quad \sum_{\alpha} \left[B_{11}|_w \left(\frac{z_P - z_W}{\Delta\xi} \right) + B_{12}|_w \left(\frac{(z_N + z_{NW}) - (z_S + z_{SW})}{4\Delta\eta} \right) \right] + \\
 & \quad \left. C_{11}|_w \left(\frac{T_P - T_W}{\Delta\xi} \right) + C_{12}|_w \left[\frac{(T_N + T_{NW}) - (T_S + T_{SW})}{4\Delta\eta} \right] \right\}^{n+1} \Delta\eta \Delta\tau + \\
 & \left\{ \sum_{\alpha} \left[A_{21}|_n \left(\frac{(p_{\alpha,E} + p_{\alpha,NE}) - (p_{\alpha,W} + p_{\alpha,NW})}{4\Delta\xi} \right) + A_{22}|_n \left(\frac{p_{\alpha,N} - p_{\alpha,P}}{\Delta\eta} \right) \right] - \right. \\
 & \quad \sum_{\alpha} \left[B_{21}|_n \left(\frac{(z_E + z_{NE}) - (z_W + z_{NW})}{4\Delta\xi} \right) + B_{22}|_n \left(\frac{z_N - z_P}{\Delta\eta} \right) \right] + \\
 & \quad \left. C_{21}|_n \left[\frac{(T_E + T_{NE}) - (T_W + T_{NW})}{4\Delta\xi} + C_{22}|_n \left(\frac{T_N - T_P}{\Delta\eta} \right) \right] \right\}^{n+1} \Delta\xi \Delta\tau - \\
 & \left\{ \sum_{\alpha} \left[A_{21}|_s \left(\frac{(p_{\alpha,E} + p_{\alpha,SE}) - (p_{\alpha,W} + p_{\alpha,SW})}{4\Delta\xi} \right) + A_{22}|_s \left(\frac{p_{\alpha,P} - p_{\alpha,S}}{\Delta\eta} \right) \right] - \right. \\
 & \quad \sum_{\alpha} \left[B_{21}|_s \left(\frac{(z_E + z_{SE}) - (z_W + z_{SW})}{4\Delta\xi} \right) + B_{22}|_s \left(\frac{z_P - z_S}{\Delta\eta} \right) \right] + \\
 & \quad \left. C_{21}|_s \left[\frac{(T_E + T_{SE}) - (T_W + T_{SW})}{4\Delta\xi} + C_{22}|_s \left(\frac{T_P - T_S}{\Delta\eta} \right) \right] \right\}^{n+1} \Delta\xi \Delta\tau + \frac{S}{J} \Big|_P^{n+1} \Delta\xi \Delta\eta \Delta\tau. \quad (\text{A.31})
 \end{aligned}$$

En la ecuación (A.31) se tienen que evaluar las temperaturas en la frontera del volumen de control para poder conocer las variables físicas que dependen de la misma, tales como la densidad, viscosidad y la entalpía. En este trabajo se utiliza una interpolación lineal para calcular las temperaturas de las caras.

Rearreglando los términos de la ecuación (A.31) y considerando $\Delta\xi = \Delta\eta = 1$ se obtiene la discretización bidimensional en el plano computacional de la ecuación de balance de energía:

$$\begin{aligned}
 [Q - Q^n]_P / (J\Delta\tau) = & \\
 - \sum_{\alpha} (L_{\alpha,PP\alpha,P}) + \sum_{\alpha} (L_{\alpha,EP\alpha,E}) + \sum_{\alpha} (L_{\alpha,WP\alpha,W}) + \sum_{\alpha} (L_{\alpha,NP\alpha,N}) + \sum_{\alpha} (L_{\alpha,SP\alpha,S}) & \\
 + \sum_{\alpha} (L_{\alpha,NEP\alpha,NE}) + \sum_{\alpha} (L_{\alpha,SEP\alpha,SE}) + \sum_{\alpha} (L_{\alpha,NWP\alpha,NW}) + \sum_{\alpha} (L_{\alpha,SWP\alpha,SW}) & \\
 + \sum_{\alpha} (M_{\alpha,PzP}) - \sum_{\alpha} (M_{\alpha,EzE}) - \sum_{\alpha} (M_{\alpha,WzW}) - \sum_{\alpha} (M_{\alpha,NzN}) - \sum_{\alpha} (M_{\alpha,SzS}) & \\
 - \sum_{\alpha} (M_{\alpha,NEzNE}) - \sum_{\alpha} (M_{\alpha,SEzSE}) - \sum_{\alpha} (M_{\alpha,NWzNW}) - \sum_{\alpha} (M_{\alpha,SWzSW}) & \\
 - N_P T_P + N_E T_E + N_W T_W + N_N T_N - N_S T_S - N_{NE} T_{NE} - N_{SE} T_{SE} - N_{NW} T_{NW} - N_{SW} T_{SW} + \frac{S}{J} \Big|_P, & \\
 \end{aligned} \tag{A.32}$$

donde los coeficientes $L_{\alpha,XY}$, $M_{\alpha,XY}$ y N_{XY} son:

$$\begin{aligned}
 L_{\alpha,P} = A_{11}|e + A_{11}|w + A_{22}|n + A_{22}|s, \quad M_{\alpha,P} = B_{11}|e + B_{11}|w + B_{22}|n + B_{22}|s, \quad N_P = C_{11}|e + C_{11}|w + C_{22}|n + C_{22}|s, \\
 L_{\alpha,E} = A_{11}|e + \frac{A_{21}}{4}|n - \frac{A_{21}}{4}|s, \quad M_{\alpha,E} = B_{11}|e + \frac{B_{21}}{4}|n - \frac{B_{21}}{4}|s, \quad N_E = C_{11}|e + \frac{C_{21}}{4}|n - \frac{C_{21}}{4}|s, \\
 L_{\alpha,W} = A_{11}|w + \frac{A_{21}}{4}|s - \frac{A_{21}}{4}|n, \quad M_{\alpha,W} = B_{11}|w + \frac{B_{21}}{4}|s - \frac{B_{21}}{4}|n, \quad N_W = C_{11}|w + \frac{C_{21}}{4}|s - \frac{C_{21}}{4}|n, \\
 L_{\alpha,N} = A_{22}|n + \frac{A_{12}}{4}|e - \frac{A_{12}}{4}|w, \quad M_{\alpha,N} = B_{22}|n + \frac{B_{12}}{4}|e - \frac{B_{12}}{4}|w, \quad N_N = C_{22}|n + \frac{C_{12}}{4}|e - \frac{C_{12}}{4}|w, \\
 L_{\alpha,S} = A_{22}|s + \frac{A_{12}}{4}|w - \frac{A_{12}}{4}|e, \quad M_{\alpha,S} = B_{22}|s + \frac{B_{12}}{4}|w - \frac{B_{12}}{4}|e, \quad N_S = C_{22}|s + \frac{C_{12}}{4}|w - \frac{C_{12}}{4}|e, \\
 L_{\alpha,NE} = \frac{A_{12}}{4}|e + \frac{A_{21}}{4}|n, \quad M_{\alpha,NE} = \frac{B_{12}}{4}|e + \frac{B_{21}}{4}|n, \quad N_{NE} = \frac{C_{12}}{4}|e + \frac{C_{21}}{4}|n, \\
 L_{\alpha,SE} = -\frac{A_{12}}{4}|e - \frac{A_{21}}{4}|s, \quad M_{\alpha,SE} = -\frac{B_{12}}{4}|e - \frac{B_{21}}{4}|s, \quad N_{SE} = -\frac{C_{12}}{4}|e - \frac{C_{21}}{4}|s, \\
 L_{\alpha,NW} = -\frac{A_{12}}{4}|w - \frac{A_{21}}{4}|n, \quad M_{\alpha,NW} = -\frac{B_{12}}{4}|w - \frac{B_{21}}{4}|n, \quad N_{NW} = -\frac{C_{12}}{4}|w - \frac{C_{21}}{4}|n, \\
 L_{\alpha,SW} = \frac{A_{12}}{4}|w + \frac{A_{21}}{4}|s, \quad M_{\alpha,SW} = \frac{B_{12}}{4}|w + \frac{B_{21}}{4}|s, \quad N_{SW} = \frac{C_{12}}{4}|w + \frac{C_{21}}{4}|s. \\
 \end{aligned} \tag{A.33}$$

Las ecuaciones discretas (A.25) y (A.32) se resuelven mediante el método de Newton-Raphson para simular numéricamente el problema de inyección de vapor en dominios estructurados no ortogonales.

Apéndice B

Ecuaciones constitutivas para el modelo de inyección de vapor

B.1. Propiedades de flujo

La determinación de las permeabilidades relativas para un sistema trifásico aun no se ha establecido con claridad y es difícil experimentalmente hablando. Comenzando por Leverett y Lewis, la mayoría de las mediciones para permeabilidades relativas de un sistema trifásico han sido experimentales [44]. Estas mediciones indican que las permeabilidades relativas de las fases mojante y no-mojante en un sistema trifásico son funciones de sus respectivas saturaciones como en un sistema bifásico. La permeabilidad relativa para la fase intermedia es una función de las dos saturaciones independientes,

$$k_{ro_{ow}} = k_{ro}(S_w, S_g). \quad (\text{B.1})$$

La forma funcional de la ecuación B.1 raramente se conoce. En la práctica la estimación de las permeabilidades relativas de un sistema trifásico está basada en dos grupos de datos de dos sistemas bifásicos, la permeabilidad relativa de un sistema intermedio mojante agua-aceite, se define como:

$$k_{ro_{ow}} = k_{ro_{ow}}(S_w), \quad (\text{B.2})$$

y la permeabilidad relativa de un sistema intermedio no mojante gas-aceite:

$$k_{ro_{og}} = k_{ro_{og}}(S_g). \quad (\text{B.3})$$

El procedimiento más simple para determinar la permeabilidad relativa k_{ro} de un sistema trifásico es:

$$k_{ro} = k_{roow}k_{roog}. \quad (\text{B.4})$$

En este trabajo se utiliza el esquema de Stone's II para determinar la permeabilidad relativa del aceite k_{ro} de un sistema trifásico [44]:

$$k_{ro} = k_{rc} [(k_{roow}/k_{rc} + k_{rw})(k_{roog}/k_{rc} + k_{rg}) - (k_{rw} + k_{rg})]. \quad (\text{B.5})$$

Por otra parte, la presión capilar agua-aceite se considera únicamente función de la saturación de agua S_w :

$$p_{c_{ow}} = p_{c_{ow}}(S_w), \quad (\text{B.6})$$

donde la dependencia funcional esta determinada por el sistema agua-aceite. La presión capilar gas-aceite se considera función de la saturación del gas $S_{g_{og}}$ en un sistema gas-aceite:

$$p_{c_{og}} = p_{c_{og}}(S_{g_{og}}). \quad (\text{B.7})$$

Además de las permeabilidades relativas y presiones capilares, se tienen que definir la forma y dependencia de las otras propiedades físicas de los fluidos tale como: la densidad, viscosidad y entalpía.

B.2. Propiedades del aceite

Para la densidad del aceite se tiene la siguiente relación dependiente de la temperatura:

$$\rho_o(T) = \rho_{o-Ref} - C_1(T - 60) + C_2(T - 60)^2, \quad (\text{B.8})$$

donde C_1 y C_2 , tiene la siguiente forma:

$$C_1 = 0.0133 + 152.4(\rho_{o-Ref})^{-2.45},$$

$$C_2 = 8.1 \times 10^{-06} - 0.0622 \times 10^{(0.0794\rho_{o-Ref})}.$$

Posteriormente, se aplica una corrección debido a la presión:

$$\rho_o(T, p) = \rho_o(T)[1 + C_o(p - p_{int})], \quad (\text{B.9})$$

donde C_o , está dado por:

$$\begin{aligned}
C_o &= \left(\frac{R_1}{R_2} - 1.0 \right) / \Delta p, \\
R_1 &= \rho_{o-Ref} + D_1 P_1 - D_2 P_1^2, \\
R_2 &= \rho_{o-Ref} + D_1 P - D_2 P^2, \\
D_1 &= 0.167 + 16.181 X 10^{-(0.0425 \rho_{o-Ref})}, \\
D_2 &= 0.00299 + 2.63 X 10^{-(0.0603 \rho_{o-Ref})}, \\
P_1 &= (p + \Delta p) / 1000, \\
P &= p / 1000,
\end{aligned}$$

además la Δp se ha seleccionado igual 100psi para este trabajo, la presión p en psi, la compresibilidad C_{o1} en psi^{-1} y la densidad ρ_o esta dada en lbm/ft^3 [10]. Por otra parte, la viscosidad del aceite esta dada por:

$$\mu_o(T) = A e^{B/T}. \quad (\text{B.10})$$

Las constantes A y B dependen del tipo de aceite que se considere. En la ecuación μ_o esta dada en cP y T es la temperatura absoluta en R o K . Para los problemas abordados en este documento, los datos de viscosidad también se proporcionan en tablas, por lo que se interpolan a partir de los datos proporcionados.

El calor específico c_o del aceite esta dado por la siguiente relación [53]:

$$c_o(T) = \frac{0.388 + 0.00045T}{\sqrt{\gamma_o}}, \quad (\text{B.11})$$

donde c_o tiene unidades de $\text{Btu}/\text{lb}^\circ\text{F}$, la temperatura se proporciona en $^\circ\text{F}$ y la gravedad específica depende del $^\circ\text{API}$ del hidrocarburo en cuestión de la siguiente manera [53]:

$$\gamma_o = \frac{141.5}{^\circ\text{API} + 131.5}. \quad (\text{B.12})$$

B.3. Propiedades del agua

La dependencia del agua con la temperatura $\rho_w(T)$, se describe mediante la siguiente correlación:

$$\begin{aligned}
\rho_w(T) &= 236.372 - 1.29187T + 0.00378125T^2 - 5.40258X10^{-06}T^3 \\
&\quad + 3.74277X10^{-09}T^4 - 1.01916X10^{-12}T^5.
\end{aligned} \quad (\text{B.13})$$

Posteriormente, se aplica una corrección por presión como sigue:

$$\rho_w(T, p) = \rho_w(T)[1 + C_w(p - p_{int})], \quad (\text{B.14})$$

donde p esta en psi, C_w en psi^{-1} y ρ_w en lbm/ft^3 . La viscosidad del agua μ_w cambia con la temperatura de la siguiente manera:

$$\mu_w(T) = \left(\frac{1176 - T}{25.6T - 89} \right), \quad (\text{B.15})$$

aquí T es en $^{\circ}\text{F}$ y μ_w esta cP. Para la entalpía del agua, se tiene la siguiente correlación:

$$\begin{aligned} H_w(T) = & 10174.2 - 87.4729T + 0.301147T^2 - 5.38409X10^{-04}T^3 \\ & + 5.33392X10^{-07}T^4 - 2.77814X10^{-10}T^5 + 5.95201X10^{-14}T^6. \end{aligned} \quad (\text{B.16})$$

donde T esta en $^{\circ}\text{R}$ y H_w tiene unidades Btu/lbm.

B.4. Propiedades del vapor

Para la densidad del vapor se utiliza la ecuación de los gases ideales:

$$\rho_s(T, p) = \frac{pM}{ZRT}, \quad (\text{B.17})$$

donde M es el peso molecular, p la presión en psi, T es la temperatura en $^{\circ}\text{R}$, R es la constante universal de los gases $R = 10.731 \text{ psi ft}^3/(\text{lbmole } ^{\circ}\text{R})$ y Z el factor de compresibilidad del vapor:

$$Z = 1.012 - 4.461X10^{-04}T + 2.98X10^{-06}T^2 - 1.663X10^{-08}T^3, \quad T = ^{\circ}\text{C}. \quad (\text{B.18})$$

Para calcular la viscosidad del vapor, se utiliza la siguiente expresión:

$$\mu_s(T) = (0.2T + 81.97)(10^{-04}), \quad (\text{B.19})$$

en la expresión B.19 la temperatura se proporciona en $^{\circ}\text{F}$ y la viscosidad es en cP. Por último, la entalpía del vapor es:

$$\begin{aligned} H_s(T) = & 9469.85 + 87.2545T - 0.299668T^2 + 5.4361X10^{-04}T^3 \\ & - 5.4648X10^{-07}T^4 + 2.8875X10^{-10}T^5 + 6.2806X10^{-14}T^6, \end{aligned} \quad (\text{B.20})$$

donde T esta en $^{\circ}\text{R}$ y H_s tiene unidades Btu/lbm.

B.5. Modelado de pozos

Para incorporar los pozos de inyección y de producción de una manera apropiada se consideró 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) [44]. Por lo tanto, para un pozo vertical ya sea de inyección o producción correspondiente a la fase α se tiene:

$$q_\alpha = \frac{2\pi\rho_\alpha k_{r\alpha} \Delta z \sqrt{k_{11}k_{22}}}{\mu_\alpha (\ln(r_e/r_{well}) + s)} (p_{bh} - p_\alpha - \rho_\alpha \wp (z_{bh} - z)), \quad (\text{B.21})$$

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 (B.21) para un medio anisotropico se define como [44]:

$$r_e = \frac{0.14 \left((k_{22}/k_{11})^{1/2} \Delta x^2 + (k_{11}/k_{22})^{1/2} \Delta y^2 \right)^{1/2}}{0.5 \left((k_{22}/k_{11})^{1/4} + (k_{11}/k_{22})^{1/4} \right)}, \quad (\text{B.22})$$

donde Δx y Δy se refieren al tamaño de celda en los ejes x y y que contiene el pozo vertical. El índice del pozo, se expresa como sigue:

$$WI = \frac{2\pi \Delta z \sqrt{k_{11}k_{22}}}{\ln(r_e/r_{well}) + s}. \quad (\text{B.23})$$

Los efectos en la cercanía del pozo que pudieran afectar la productividad o inyectividad pueden ser incorporados usando el factor superficial s . Estos efectos se desprecian en los problemas abordados en los capítulos anteriores, por lo que para todos los problemas de este trabajo de tesis se considera $s = 0$.

Para el caso de pozos perforados horizontalmente se tienen que llevar algunas modificaciones a la ecuación (B.21) para considerar la orientación correcta en los ejes coordenados. El índice de un pozo horizontal paralelo a la dirección del eje x es:

$$WI = \frac{2\pi \Delta x \sqrt{k_{22}k_{33}}}{\ln(r_e/r_{well}) + s} \quad (\text{B.24})$$

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

$$WI = \frac{2\pi \Delta y \sqrt{k_{11}k_{33}}}{\ln(r_e/r_{well}) + s} \quad (\text{B.25})$$

El acople de los pozos a las ecuaciones de balance 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 $p_{bh} = cte$, el cambio en el método de Newton-Raphson es $\delta p_{bh} = 0$, por lo que sólo se tienen que calcular las derivadas de la ecuación (B.21) con respecto a las variables primarias seleccionadas. Para afondar en más casos particulares el lector puede consultar la referencia [44].

B.6. Término de condensación- evaporación M_g

Existen varias restricciones para calcular el término de condensación evaporación M_g . La primera restricción es cuando la temperatura está por debajo del punto de saturación de vapor y no existe vapor fluyendo en dicha región, el término M_g tendrá un valor de cero ($M_g = 0$).

Otra restricción a tomar en cuenta, es cuando el vapor exista en un volumen discreto y el volumen adyacente se encuentre abajo de la temperatura de saturación de vapor. El vapor fluye al elemento adyacente, de acuerdo al término convectivo de la ecuación de balance de masa, pero en forma de agua condensada. Entonces, todo el vapor que fluya hacia un volumen frío es inmediatamente condensado, por ejemplo, cuando el vapor está fluyendo de izquierda a derecha, se tiene la siguiente expresión:

$$M_{g,P} = -T_{g,w}(p_{g,P} - p_{g,W}) \quad \text{para} \quad p_{g,W} > p_{g,P} \quad \& \quad T_W \geq T_{vapor} \quad \& \quad T_P < T_{vapor}. \quad (\text{B.26})$$

De la misma manera que para la ecuación (B.26), se pueden obtener las componentes de M_g tomando en cuenta las 6 caras de un volumen de control tridimensional.

La clave para obtener un código numérico estable cuando se resuelve el problema de inyección de vapor secuencialmente recae en la tercera restricción, la cual trata con la formulación matemática del término de condensación en aquellas regiones donde el vapor existe. Para estas zonas, el término M_g tiene la siguiente forma [1]:

$$M_g = \frac{-1}{h_g - h_w} \left\{ \nabla \cdot k_T \nabla T - \sum_{\alpha} (v_{\alpha} \rho_{\alpha} h_{\alpha}) + \sum_{\alpha} q_{\alpha} - \left[\phi \sum_{\alpha} \left(S_{\alpha} \rho_{\alpha} \frac{\partial h_{\alpha}}{\partial T} \right) + (1 - \phi)(\rho c)_R \right] \frac{\partial T}{\partial p} (p_g - p_g)^{old} \right\}. \quad (\text{B.27})$$

Siguiendo la metodología presentada en capítulos anteriores el término M_g se discretiza y se linealiza para acoplarlo a las ecuaciones de balance de masa y energía para simular numéricamente el proceso de inyección de vapor. Cabe señalar que para utilizar el término M_g en el plano transformado, éste se debe de transformar con base en la metodología presentada en el Capítulo 4.

Apéndice C

Programación utilizando CUDA

La programación en CUDA se lleva a cabo definiendo funciones llamadas *kernel*, las cuales ejecutan las operaciones en paralelo mediante la utilización de hilos de cálculo. Para la ejecución de una función kernel, es necesario definir una malla de bloques (*grid of blocks*) y los hilos por bloque (*threads per block*). La cantidad máxima de bloques y los hilos por bloque están definidos estrictamente por el hardware que se utilice. En la figura C.1, se muestra el código para la implementación de la suma de matrices en CUDA y su ejecución desde el programa principal. En esta figura la palabra clave `__global__` define la función kernel, las variables `i` y `j` definen los índices de los threads para llevar a cabo la suma de matrices. La línea `MatAdd<<<numBlocks,threadsPerBlock>>>(A, B, C)` dentro del programa `int main()` ejecuta la función para la suma de matrices en paralelo.

```
// Definición de la función kernel
__global__ void MatAdd(float A[N][N], float B[N][N], float C[N][N])
{
    int i = blockIdx.x * blockDim.x + threadIdx.x;
    int j = blockIdx.y * blockDim.y + threadIdx.y;
    if (i < N && j < N) C[i][j] = A[i][j] + B[i][j];
}

int main()
{...
// Ejecución de la función kernel
    dim3 threadsPerBlock(16, 16);
    dim3 numBlocks(N / threadsPerBlock.x, N / threadsPerBlock.y);
    MatAdd<<<numBlocks,threadsPerBlock>>>(A, B, C);
... }
```

Figura C.1: Función kernel para la suma de matrices en CUDA

Para mejor entendimiento de la función presentada, es necesario conocer como se asignan los índices `i` y `j` que definen los threads o hilos de ejecución en paralelo. La figura C.2, muestra la distribución de los threads dentro de una malla de bloques.

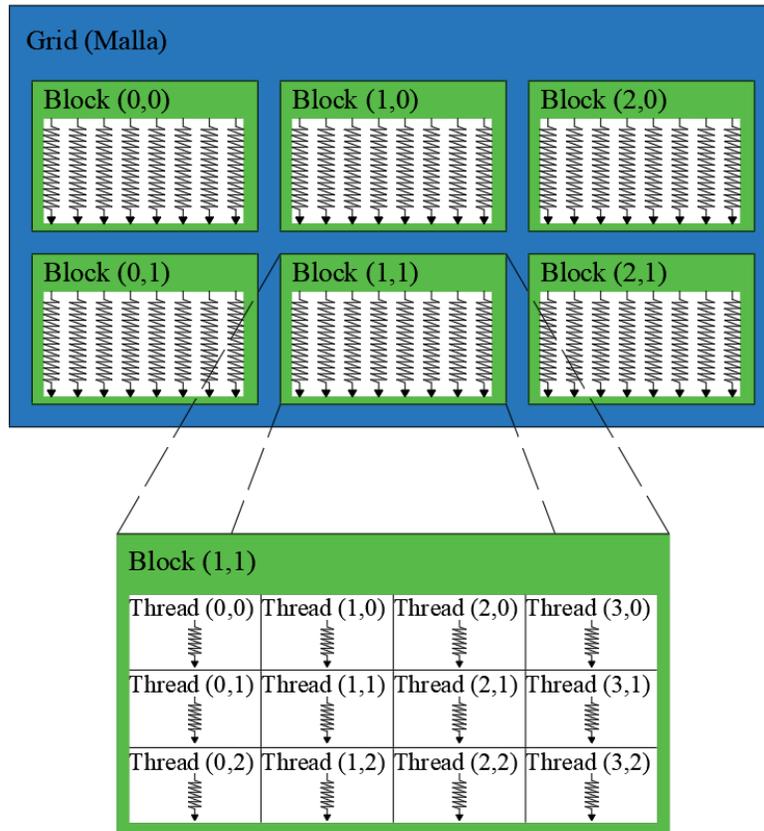


Figura C.2: Hilos de cálculo (threads) en un kernel.
(Ilustración adaptada del sitio web: <https://www.developer.nvidia.com/cuda-toolkit/>.)

Con base en lo mostrado en la figura C.2, se deduce que cada suma valor a valor en la operación $C[i][j]=A[i][j]+B[i][j]$ se realiza por un thread en forma independiente, es decir, la suma se realiza en paralelo, mientras que en un código en serie se realiza mediante un ciclo para recorrer así uno a uno los valores de las matrices.

Otra característica del lenguaje CUDA es que permite escribir códigos híbridos en serie-paralelo. Esta característica es de suma importancia puesto, que la mayoría de los códigos numéricos cuentan con partes no paralelizables, es decir, no todas las operaciones son independientes entre sí. La figura C.3 muestra la estructura de un código serie-paralelo utilizando CUDA. Esta característica se utilizó para la implementación de los códigos numéricos reportados este documento.

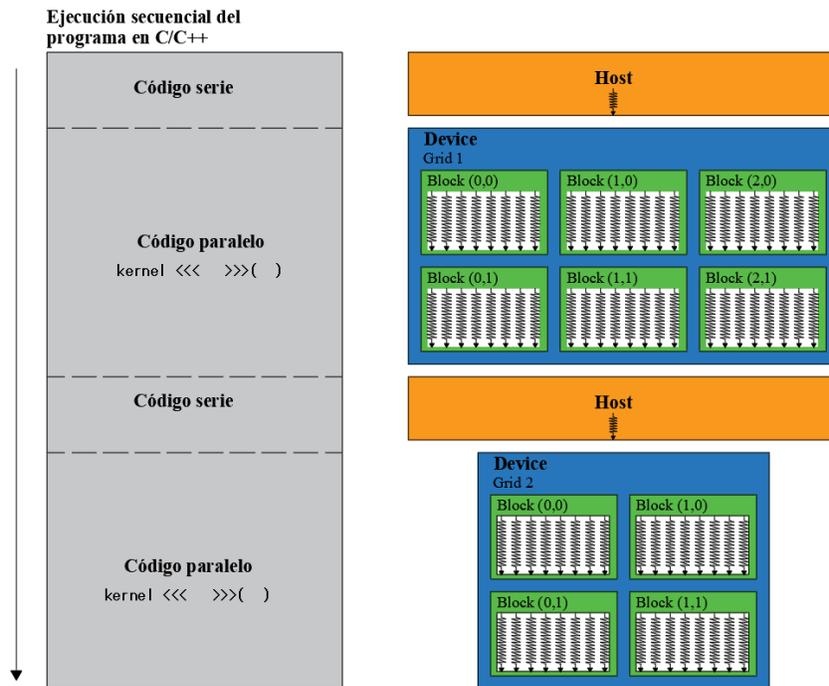


Figura C.3: Estructura de un código serie-paralelo en CUDA
 (Ilustración adaptada del sitio web: [https://www.developer.nvidia.com/cuda-toolkit.](https://www.developer.nvidia.com/cuda-toolkit))

Bibliografía

- [1] N. Shutler *et al.*, “Numerical, three-phase simulation of the linear steamflood process,” *Society of Petroleum Engineers Journal*, vol. 9, no. 02, pp. 232–246, 1969.
- [2] L. W. Lake, *Enhanced Oil Recovery*. New Jersey, United States: Prentice Hall, 1 ed., 1989.
- [3] E. R. Germán, “El futuro de la producción de aceite en México: Recuperación avanzada y mejorada,” tech. rep., Comisión Nacional de Hidrocarburos, Distrito Federal, México, 2012.
- [4] M. Latil, *Enhanced oil recovery*. France, Paris: Editions TECHNIP, 1 ed., 1980.
- [5] N. Shutler *et al.*, “Numerical three-phase model of the two-dimensional steamflood process,” *Society of Petroleum Engineers Journal*, vol. 10, no. 04, pp. 405–417, 1970.
- [6] A. Abdalla, K. Coats, *et al.*, “A three-phase, experimental and numerical simulation study of the steam flood process,” in *Fall Meeting of the Society of Petroleum Engineers of AIME*, Society of Petroleum Engineers, 1971.
- [7] K. Coats, W. George, C. Chu, B. Marcum, *et al.*, “Three-dimensional simulation of steamflooding,” *Society of Petroleum Engineers Journal*, vol. 14, no. 06, pp. 573–592, 1974.
- [8] J. W. Grabowski, P. K. Vinsome, R. C. Lin, G. A. Behie, and B. Rubin, “A fully implicit general purpose finite-difference thermal model for in situ combustion and steam,” *Society of Petroleum Engineers Journal*, no. 8396, 1979.
- [9] B. Rubin, W. L. Buchanan, *et al.*, “A general purpose thermal model,” *Society of Petroleum Engineers Journal*, vol. 25, no. 02, pp. 202–214, 1985.
- [10] P. Sarathi, “Thermal numerical simulator for laboratory evaluation of steamflood oil recovery,” tech. rep., National Inst. for Petroleum and Energy Research, Bartlesville, OK (USA), 1991.
- [11] C. Jabbour, M. Quintard, H. Bertin, and M. Robin, “Oil recovery by steam injection: three-phase flow effects,” *Journal of Petroleum Science and Engineering*, vol. 16, no. 1, pp. 109–130, 1996.

-
- [12] T. Babadagli, "Temperature effect on heavy-oil recovery by imbibition in fractured reservoirs," *Journal of Petroleum Science and Engineering*, vol. 14, no. 3, pp. 197–208, 1996.
- [13] K. Sasaki, S. Akibayashi, N. Yazawa, Q. Doan, S. F. Ali, *et al.*, "Numerical and experimental modelling of the steam assisted gravity drainage (sagd) process," *Journal of Canadian Petroleum Technology*, vol. 40, no. 1, pp. 44–50, 2001.
- [14] T. Sheorey and K. Muralidhar, "Isothermal and non-isothermal oil–water flow and viscous fingering in a porous medium," *International Journal of Thermal Sciences*, vol. 42, no. 7, pp. 665–676, 2003.
- [15] I. Y. Akkutlu and Y. C. Yortsos, "The dynamics of in-situ combustion fronts in porous media," *Combustion and Flame*, vol. 134, no. 3, pp. 229–247, 2003.
- [16] I. Kocabas, "Thermal transients during nonisothermal fluid injection into oil reservoirs," *Journal of Petroleum Science and Engineering*, vol. 42, no. 2, pp. 133–144, 2004.
- [17] M. R. Kristensen, M. G. Gerritsen, P. G. Thomsen, M. L. Michelsen, and E. H. Stenby, "An equation-of-state compositional in-situ combustion model: a study of phase behavior sensitivity," *Transport in Porous Media*, vol. 76, no. 2, pp. 219–246, 2009.
- [18] S. Yin, M. B. Dusseault, L. Rothenburg, *et al.*, "Fully coupled numerical modelling of ground surface uplift in steam injection," *Journal of Canadian Petroleum Technology*, vol. 49, no. 1, p. 16, 2010.
- [19] E. Niz-Velásquez, R. Moore, K. C. van Fraassen, S. A. Mehta, M. G. Ursenbach, *et al.*, "Experimental and numerical modeling of three-phase flow under high-pressure air injection," *SPE reservoir evaluation & engineering*, vol. 13, no. 5, p. 782, 2010.
- [20] A. Azad and R. J. Chalaturnyk, "An improved sagd analytical simulator: Circular steam chamber geometry," *Journal of Petroleum Science and Engineering*, vol. 82, pp. 27–37, 2012.
- [21] B. Hascakir, T. Babadagli, and S. Akin, "Experimental and numerical modeling of heavy-oil recovery by electrical heating," International Thermal Operations and Heavy Oil Symposium, 20-23 October, Calgary, Alberta, Canada, Society of Petroleum Engineers, 2008.
- [22] S. K. Kouhanestani, R. B. Boozarjomehry, and M. Pishvaie, "Fully implicit compositional thermal simulator using rigorous multiphase calculations," *Scientia Iranica*, vol. 18, no. 3, pp. 509–517, 2011.
- [23] J. Killough, R. Bhogeswara, *et al.*, "Simulation of compositional reservoir phenomena on a distributed-memory parallel computer," *Journal of Petroleum Technology*, vol. 43, no. 11, pp. 1–368, 1991.

-
- [24] G. S. Shiralkar, R. Stephenson, W. Joubert, O. Lubeck, B. van Bloemen Waanders, *et al.*, “Falcon: A production quality distributed memory reservoir simulator,” *SPE Reservoir Evaluation & Engineering*, vol. 1, no. 05, pp. 400–407, 1998.
- [25] Y. Ma and Z. Chen, “Parallel computation for reservoir thermal simulation of multicomponent and multiphase fluid flow,” *Journal of Computational Physics*, vol. 201, no. 1, pp. 224–237, 2004.
- [26] A. H. Dogru, L. S. Fung, U. Middy, T. M. Al-Shaalan, J. A. Pita, K. HemanthKumar, H. Su, J. C. Tan, H. Hoy, W. Dreiman, *et al.*, “A next-generation parallel reservoir simulator for giant reservoirs,” in *SPE/EAGE Reservoir Characterization & Simulation Conference*, 2009.
- [27] K. Wang, H. Liu, and Z. Chen, “A scalable parallel black oil simulator on distributed memory parallel computers,” *Journal of Computational Physics*, vol. 301, pp. 19–34, 2015.
- [28] J. Sanders and E. Kandrot, *CUDA by example: an introduction to general-purpose GPU programming*. Addison-Wesley Professional, 2010.
- [29] R. Li and Y. Saad, “Gpu-accelerated preconditioned iterative linear solvers,” *The Journal of Supercomputing*, vol. 63, no. 2, pp. 443–466, 2013.
- [30] H. Liu, S. Yu, Z. J. Chen, B. Hsieh, L. Shao, *et al.*, “Parallel preconditioners for reservoir simulation on gpu,” in *SPE Latin America and Caribbean Petroleum Engineering Conference*, Society of Petroleum Engineers, 2012.
- [31] H. Liu, S. Yu, Z. Chen, *et al.*, “Development of algebraic multigrid solvers using gpus,” in *SPE Reservoir Simulation Symposium*, Society of Petroleum Engineers, 2013.
- [32] A. Anciaux-Sedrakian, J. Eaton, J. Gratien, T. Guignon, P. Havé, C. Preux, O. Ricois, *et al.*, “Will gpgpus be finally a credible solution for industrial reservoir simulators?,” in *SPE Reservoir Simulation Symposium*, Society of Petroleum Engineers, 2015.
- [33] K. Mukundakrishnan, K. Esler, D. Dembeck, V. Natoli, J. Shumway, Y. Zhang, J. Gilman, H. Meng, *et al.*, “Accelerating tight reservoir workflows with gpus,” in *SPE Reservoir Simulation Symposium*, Society of Petroleum Engineers, 2015.
- [34] S. Mozaffari, M. Nikookar, M. R. Ehsani, L. Sahranavard, E. Roayaie, and A. H. Mohammadi, “Numerical modeling of steam injection in heavy oil reservoirs,” *Fuel*, vol. 112, pp. 185–192, 2013.
- [35] B. Willman, V. Valleroy, G. Runberg, A. Cornelius, L. Powers, *et al.*, “Laboratory studies of oil recovery by steam injection,” *Journal of Petroleum Technology*, vol. 13, no. 07, pp. 681–690, 1961.

-
- [36] K. Aziz, A. Ramesh, P. Woo, *et al.*, “Fourth spe comparative solution project: comparison of steam injection simulators,” *Journal of petroleum technology*, vol. 39, no. 12, pp. 1–576, 1987.
- [37] M. Chan, G. Sarioglu, *et al.*, “Numerical modelling of cyclically steamed and fractured oil-sand reservoirs,” in *International Meeting on Petroleum Engineering*, Society of Petroleum Engineers, 1992.
- [38] T. Nasr, H. Golbeck, S. Lorimer, *et al.*, “Analysis of the steam assisted gravity drainage (sagd) process using experimental/numerical tools,” in *International conference on horizontal well technology*, Society of Petroleum Engineers, 1996.
- [39] M. Sumnu, W. Brigham, K. Aziz, L. Castanier, *et al.*, “An experimental and numerical study on steam injection in fractured systems,” in *SPE/DOE Improved Oil Recovery Symposium*, Society of Petroleum Engineers, 1996.
- [40] M. Yang, T. G. Harding, Z. Chen, K. Yu, H. Liu, B. Yang, R. He, *et al.*, “Numerical modelling of hybrid steam and in-situ combustion performance for oil sands,” in *SPE Reservoir Simulation Conference*, Society of Petroleum Engineers, 2017.
- [41] I. Herrera and G. F. Pinder, *Mathematical modeling in science and engineering: An axiomatic approach*. New Jersey, United States: John Wiley & Sons, 1 ed., 2012.
- [42] I. Herrera and G. S. Herrera, “Unified formulation of enhanced oil-recovery methods,” *Geofísica internacional*, vol. 50, no. 1, pp. 85–98, 2011.
- [43] Z. Chen, G. Huan, and Y. Ma, *Computational methods for multiphase flows in porous media*. SIAM, 2006.
- [44] Z. Chen, *Reservoir simulation: mathematical techniques in oil recovery*. Philadelphia, United States: SIAM, 1 ed., 2007.
- [45] N. Ozisik, *Finite difference methods in heat transfer*. Washington D.C.: CRC press, 1 ed., 1994.
- [46] C. R. Maliska, *Transfêrencia de Calor e Mecânica de Fluidos Computacional*. Rio de Janeiro, Brazil: Livros Técnicos e Científicos, 1 ed., 1995.
- [47] V. L. Teja-Juárez and M. Luis, “A graphic processing unit (gpu) based implementation of an incompressible two-phase flow model in porous media,” *Geofísica internacional*, vol. 57, no. 3, pp. 205–222, 2018.
- [48] Jacob, B and Guennebaud G, and others, *EIGEN Eigen C++, template library for linear algebra, Version 3.3.1*, 2016.
- [49] Maia, F and Dalton, S and others, *CUSP Generic Parallel Algorithms for Sparse Matrix and Graph Computations, Version 0.5.1*, 2016.

-
- [50] J. D. Hunter, “Matplotlib: A 2d graphics environment,” *Computing in science & engineering*, vol. 9, no. 3, pp. 90–95, 2007.
- [51] M. A. Diaz Viera, D. López Falcon, A. Moctezuma Berthier, and A. Ortiz Tapia, “Comsol implementation of a multiphase fluid flow model in porous media,” COMSOL Conference, Boston U.S.A., 2008.
- [52] L. Nghiem, D. A. Collins, R. Sharma, *et al.*, “Seventh spe comparative solution project: Modelling of horizontal wells in reservoir simulation,” in *SPE Symposium on Reservoir Simulation*, Society of Petroleum Engineers, 1991.
- [53] S. M. F. Ali, *Practical heavy oil recovery*. SM Farouq Ali, 1997.